IFIP AICT 387

Arthur Tatnall
(Ed.)

Reflections on the History of Computing

Preserving Memories and Sharing Stories

SURVEY

Springer

IFIP Advances in Information and Communication Technology 387

IFIP – The International Federation for Information Processing

IFIP was founded in 1960 under the auspices of UNESCO, following the First World Computer Congress held in Paris the previous year. An umbrella organization for societies working in information processing, IFIP's aim is two-fold: to support information processing within its member countries and to encourage technology transfer to developing nations. As its mission statement clearly states,

> *IFIP's mission is to be the leading, truly international, apolitical organization which encourages and assists in the development, exploitation and application of information technology for the benefit of all people.*

IFIP is a non-profitmaking organization, run almost solely by 2500 volunteers. It operates through a number of technical committees, which organize events and publications. IFIP's events range from an international congress to local seminars, but the most important are:

- The IFIP World Computer Congress, held every second year;
- Open conferences;
- Working conferences.

The flagship event is the IFIP World Computer Congress, at which both invited and contributed papers are presented. Contributed papers are rigorously refereed and the rejection rate is high.

As with the Congress, participation in the open conferences is open to all and papers may be invited or submitted. Again, submitted papers are stringently refereed.

The working conferences are structured differently. They are usually run by a working group and attendance is small and by invitation only. Their purpose is to create an atmosphere conducive to innovation and development. Refereeing is also rigorous and papers are subjected to extensive group discussion.

Publications arising from IFIP events vary. The papers presented at the IFIP World Computer Congress and at open conferences are published as conference proceedings, while the results of the working conferences are often published as collections of selected and edited papers.

Any national society whose primary activity is about information processing may apply to become a full member of IFIP, although full membership is restricted to one society per country. Full members are entitled to vote at the annual General Assembly, National societies preferring a less committed involvement may apply for associate or corresponding membership. Associate members enjoy the same benefits as full members, but without voting rights. Corresponding members are not represented in IFIP bodies. Affiliated membership is open to non-national societies, and individual and honorary membership schemes are also offered.

Arthur Tatnall (Ed.)

Reflections on the History of Computing

Preserving Memories and Sharing Stories

Springer

Volume Editor

Arthur Tatnall
Victoria University
School of Management and Information Systems
City Flinders Campus, PO Box 14428
Melbourne, VIC 8001, Australia
E-mail: arthur.tatnall@vu.edu.au

ISSN 1868-4238 e-ISSN 1868-422X
ISBN 978-3-642-33898-4 e-ISBN 978-3-642-33899-1
DOI 10.1007/978-3-642-33899-1
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012948628

CR Subject Classification (1998): K.2, K.3.1-2, A.1, B.1.0, C.5.0, D.1.0, D.2.0, D.3.0, D.4.0, H.2.0

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Editorial

Introduction to the *Survey* Subline of the IFIP AICT Series

This volume constitutes the first issue in the *Survey* subline of the IFIP Advances in Information and Communication Technology (AICT) series. Each AICT Survey is intended to provide a compendium of related articles on the development of some aspect of information and communication technology.

It is appropriate that this initial AICT *Survey* issue should address various aspects of computing history. IFIP's Working Group 9.7 of IFIP Technical Committee 9 on ICT and Society has a long and successful history of promoting the documentation of various aspects of ICT history throughout the world. This volume not only provides articles that describe historical events and developments in ICT, but also makes the case for studying our history.

Our intention is that other working groups or technical committees within IFIP will produce additional *Survey* volumes for the IFIP AICT series. We are pleased to inaugurate this additional way that IFIP can contribute to the dissemination of information about various aspects of ICT.

A. Joe Turner
IFIP AICT Editor-in-Chief

Preface

There is a tendency among many people, some of whom are historians, to consider that history is about things that happened a long time ago: that it is about wars, kings, politics, nations and such things. This book is about the history of computing, which is a comparatively recent phenomenon, as computers as we now know them came into being less than 70 years ago. We should then question why a study of the history of computing is important. The study of history is often justified by use of the famous quote from George Santayana who wrote that: "*Those who cannot remember the past are condemned to repeat it*". Are there particular computing mistakes made in the past that we should try to avoid in the future? The only way we will know the answer to this question is to study our history.

Another reason for studying history, of computing or anything else, is that it gives us an opportunity to hear some fascinating stories that, as well as being of interest, may just be relevant to our future lives. In the final chapter of this book John Impagliazzo writes of how he taught the history of computing to his computer science students, and how it can: "*produce interesting excursions on technical subjects*" and so complement their studies in computing.

This book has 22 chapters written by people who find the history of computing to be a compelling topic. Many of the authors were actually involved in the events they describe, while the others write of things they have studied carefully and find particularly interesting. Most of the authors are members of IFIP (International Federation for Information Processing) Working Group 9.7 – History of Computing. All the articles were peer reviewed before final acceptance.

Most of the articles in this book cover the period from the 1940s to the 1990s, but one goes back to look at Italian computing machines from the first century (clearly not computers as we now know them) to the 20^{th} century. A number of articles concern early electronic digital computer hardware of the late 1940s and early 1950s involving machines such as CSIRAC, LEO and the Ferranti Sirius. At the other end of the spectrum a couple of articles discuss the use of microcomputers in schools and homes in the 1980s.

The authors of many of the articles were present at the time and took part in the events they describe. Their articles represent reflections on their experiences of this time. Are they then necessarily an accurate representation of exactly what happened then? Are they precisely correct in regard to dates and times? Does this matter? Few historians have exactly the same view of past events, but by being written by those who, in many cases, took part in these events, the articles in this book are very different to those written by academic historians. The title of the book was chosen for this purpose. The book does not try to cover the whole range of computing history, just those parts the authors were involved in or find particularly interesting. It represents their reflections on this history.

The articles cover a wide range of computing-related topics. One article, *Early Italian Computing Machines and their Inventors* by Silvio Hénin, looks at some of the pre-history of digital computing. Several articles, namely *Australia's WREDAC: It Was Rocket Science* by John Deane, *Remembering LEO* by Frank Land and *Information Technology in Italy: The Origins and the Early Years (1954–1965)* by Corrado Bonfanti, describe specific early computer systems; their construction, their use and their users. Software, programming and operating systems are discussed in the following papers: *A Possible First Use of CAM/CAD by* Norman Sanders, *Micro Programming* by Herman Spanjersberg, *History of Data Centre Development* by Rihards Balodis and Inara Opmane, *The History of Computer Language Selection* by Kevin Parker and Bill Davey and *The Impact of the Y2K Event on the Popularity of the Pick Database Environment* by Stasys Lukaitis.

Another group of articles looks specifically at the people involved in the theory, design and use of these computers: *Roberto Busa (1913–2011), Pioneer of Computers for the Humanities* by Corrado Bonfanti, *From the History of Russian Computer Science* by Yakov Fet and *Hungarian Scientists in Information Technology* by Győző Kovács. Two articles provide personal reflections on their authors' own computing careers: *Experiences and Reflections* by 'Bud' Lawson and *Looking Back* by Martha Crosby.

Computer education is important and aspects of university computing curricula are a topic of interest with articles including: *Information Systems Degrees in Australia: The Genesis* by Audra Lukaitis, Stasys Lukaitis and Bill Davey, *Evolution of Computer Science Education in the Purview of Free Education* by Nandasara S.T. and *Evolution of Computer Education in Spain: From Early Times to the Implementation of the Bologna Agreement* by Ramon Puigjaner and, as mentioned earlier, John Impagliazzo's article on *My Fascination with Computing History*. School and home computing in the 1980s is also a topic of interest with two articles: *Computing for the Masses? Constructing a British Culture of Computing in the Home* by Tilly Blythe and *Reflections on the History of Computer Education in Schools in Victoria* by Arthur Tatnall and Bill Davey covering some of this topic.

A couple of articles deal with conservation of computing technology and *Institutional Nostalgia – Museum Victoria's Cabinet of Computing Curiosities* by David Demant and Arthur Tatnall and *The Changing Face of the History of Computing: The Role of Emulation in Protecting Our Digital Heritage* by David Anderson, Janet Delve and Vaughan Powell discuss this topic.

The articles in this book cover the history of computing only up to the 1990s, but what significant events occurred after that time? The answer is that there were many, but that this is for future historians to investigate. I hope that you find the articles in the book both interesting and informative. These reflections really do represent some important preserved memories and are stories that ought to be shared.

July 2012 Arthur Tatnall

Table of Contents

Australia's WREDAC – It *Was* Rocket Science

John Deane

Australia

Abstract. From 1950 Australia's *Long Range Weapons Establishment* took steps into an almost unknown future. Building on a tiny bit of experience John Ovenstone addressed a deepening problem with calculations and defined an *automatic computing machine*. Elliott Brothers used their electronics expertise and bent their efforts from developing their first commercial computer to fill Ovenstone's order. As LRWE became the *Weapons Research Establishment*, the *ELLIOTT 403* digital automatic computer became *WREDAC*, and Australia's second computer - just[1]. WRE's computer was special, it took input from locally built analogue to digital conversion of missile range data, processed this with locally written software, and produced performance reports, off-line, on Australia's first line printer and the world's first digital plotters. While this machine was a number cruncher, Ovenstone saw that it could be used for business applications - he programmed demonstration examples and told everyone who would listen that this was the way ahead. Like the other first generation computers *WREDAC* soon had competition from fast, reliable transistorised machines. Unlike the others *WREDAC* did not have a university environment to support it and its life was relatively short - but productive and inspiring.

Keywords: Weapons research, long range weapons, Woomera (Australia), ELLIOTT 403.

1 Vengeance Weapons and Woomera

Late in 1944, in the fifth year of World War 2, the German army started firing V-2 rockets at Paris, London and many Allied cities. It was understood that they were ballistic missiles carrying around 1 tonne of high explosives, but there was no known defence and the Allies couldn't duplicate the technology.

At the end of the war, and with the added knowledge of the US atomic bomb, the UK *Ministry of Supply* created a *Guided Weapons Directorate* to develop rocket based armaments. One of their first problems was to create a test site at least 800 km long. Europe was too densely populated and firing over water would limit the recovery of missiles. The best choice seemed to be either somewhere in Canada or Australia[2].

---

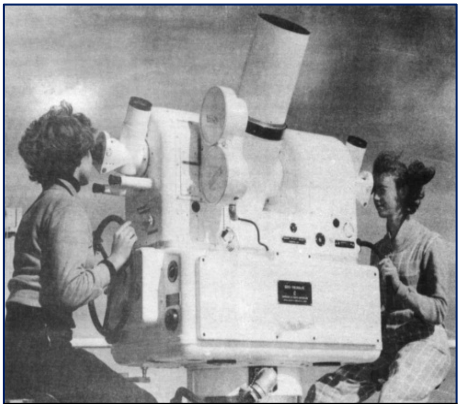
[1] Trevor Pearcey's CSIRAC first ran in November 1949, WREDAC in June 1956 (though it probably ran in the UK about August 1955), SILLIAC in July 1956 and UTECOM in September 1956 (and this English Electric DEUCE probably ran in the UK about April 1956).

[2] Peter Morton, 'Fire across the Desert' (Australian Government Publishing Service, 1989). Morton's large and impressive book has a detailed account of many matters associated with Woomera, and especially demonstrates the complexity of management as well as technical aspects.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 1–21, 2012.

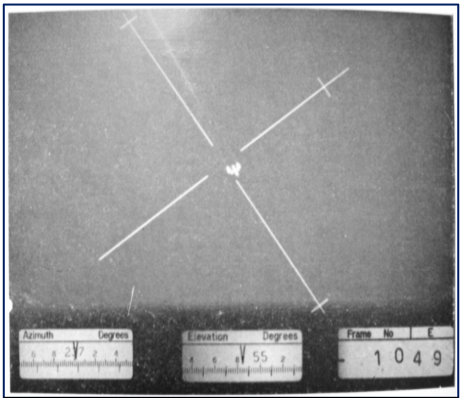
An Australian Army officer, William Coulson, stationed in London heard of this proposal and drew up a map with a suggested test range running from Mt Eba in South Australia, north-west for 1,800 km to the coast of Western Australia near Broome. This became the core of a proposal to the Australian Prime Minister, J.B. Chifley, in September 1945. This promised access to weapons information, technical development, and money – and the Australian Government rapidly agreed in principle[3]. A UK team surveyed the range in early 1946 and generated a substantial report. This resulted in the creation of the *Anglo-Australian Joint Project* at year's end, then with the arrival of the core UK scientific team, the *Long Range Weapons Establishment* (LRWE) on 1st April 1947[4].

The LRWE was housed in what had recently been a large munitions factory at Salisbury 20 km north of Adelaide, while the newly christened *Woomera*[5] rocket range started a further 500 km north-west[6].



Kinetheodolite tracking a missile on the Woomera Range (from "The Weapons Research Establishment and Rocket Range" pamphlet WRE 1956)

Photo: Courtesy, Defence Science & Technology Organisation



One frame of kinetheodolite film (from WRE Annual Report 1967-1968)

Photo: Courtesy, Defence Science & Technology Organisation

Their job was to evaluate rockets principally developed by the UK military contractors and provide detailed technical reports. Considerable equipment was installed on the range to do this. Initially:

---

[3] Peter Morton 1989 *ibid*. Morton has multiple references to William Coulson.

[4] Peter Morton 1989 *ibid*. Lt General John Evetts led the initial survey as well as the scientific team. His report which defined what the LRWE would do was known as *Evetts' Bible*.

[5] A 'woomera' is an Australian Indigenous invention to aid in launching a spear. The thrower uses a long grooved stick to increase the leverage of the throw and propel the spear with extra force.

[6] 'The Weapons Research Establishment and Rocket Range' (Commonwealth Department of Supply, 1956).

- a wide angle camera which took multiple exposures on a glass photographic plate – this was only useful at night,
- a film movie camera,
- a pair of kinetheodolites used to track the rocket manually - the telescope image and telescope bearing and elevation angles were recorded on movie film,
- radar provided bearing and range which were plotted to show the current position of the rocket and also recorded by movie camera,
- 12 geophones arrayed round the target position plus a multi-channel sound recorder,
- a precision timing source[7].

These were duplicated and extended throughout the life of the Range. Many rockets also transmitted internal state telemetry by radio which was displayed on a CRT screen and recorded by another movie camera.

Three "Computers" (from the film "WRE Maths Services", NAA D4994/CA7504/146 1953)

Photo: Courtesy, Defence Science & Technology Organisation

What was done with the recordings from these instruments varied according to the rocket developer's requirements, but there was a frequent request for the track in three dimensions, speed through the flight and rotation of the rocket body. After developing the photographic films, analysis started with labour intensive tabulation of data on a frame-by-frame basis. For the kinetheodolite record, the two angle values plus any offset of the rocket image from the centre of the frame were read on a film viewer and listed[8].

The initial planning for the Woomera range included 'Computers' to process this data. In 1946 this meant young women operating electro-mechanical desk calculators. The original estimate was that 6 *Computers* could analyse one rocket trial in a week.

2 Computers to a Computer

As trials got under way the team of *Computers* could transcribe 10,000 data points from film in 50 hours, calculate the 3-dimensional track in another 50 hours, and plot the results in 20 hours. In a particular trial in 1950 there was a discrepancy between the

---

[7] Peter Morton 1989 *ibid*. The kinetheodolites were by Askania, radar was an anti-aircraft AA No.3 Mk.7 unit, position plotting tables were by EMI, the geophone recorder was by Miller, and the timing source was built by AWA.

[8] Peter Morton 1989 *ibid*. This lists the many instruments used and how they changed over time plus many of the problems with using the recordings.

hand calculated track and that recorded by radar – 5 months later the analysis was 'almost complete'![9]

The LRWE managers were aware of overseas events and sent four representatives to the 'Conference on High Speed Automatic Computing Machines' held at the University of Cambridge in mid-1949. In their report they wrote:

The following remarks are made bearing in mind the possible requirement of a general purpose computer for Australian use.

1. A number of high speed digital computers are at present under development in England. Those actually operating are EDSAC ... [at] Cambridge; the Manchester University's computer[10]; and ACE ... at the National Physical Laboratory. All these are general purpose machines, and can be used to solve any problem capable of numerical solution to a high order of accuracy.
2. All the English machines have fundamental differences in conception and design. ... A thorough investigation of these and many other alternatives would be necessary before a choice of design could be made.
3. From informal discussions held during the conference, the capital outlay for such general purpose computers appears to be about £10,000 sterling. This compares favourably with the estimated cost of the Royal Aircraft Establishment's electronic differential analyser, £20,000, and with the annual hire charge for a complete punched-card installation – about £5,000. ...
4. An analysis of the time factor for the calculation of a certain problem ... has shown that the use of an electronic machine can reduce the computation time by hand methods by a factor of 500.
5. A detailed examination of EDSAC circuitry shows that all radio components are standard, and not subject to extreme tolerances. Many complete units, (e.g. teleprinters), would be available as war surplus. ...

   In conclusion it is felt that a high speed digital computer could be constructed in Australia without importing materials or labour for the project, but that much investigation would need to precede its design details. Whatever its form, LRWBE[11] should have sufficient suitable problems arising from missile and aircraft developments to keep such a machine economically occupied[12].

They were also aware of Trevor Pearcey's progress towards building an electronic computer for CSIRO and established good relations. Pearcey sent his circuit diagrams and LRWE's *Maths Services Section* assembled a proposal to construct the 'LRWE

---

[9] NAA Series D174/84 Item SA5145 Part 1 'Computation of trials results 15 May 1950'

[10] i.e. the 'Small Scale Experimental Machine' or *BABY*

[11] Long Range Weapons Base Establishment

[12] NAA Series D174/84 Item SA5145 Part 1 'Computation of trials results 27 July 1949'. The LRWE attendees were G.E.Barlow, B.S.Deegan, W.C.J.White and R.H.Whitten.

Electronic Digital Automatic Computer' – *LEDAC*. This "*should use standard components wherever possible, and should be built with ease of servicing in mind. The design should not concentrate on speed; reliability was more important*"[13].

Over 1950 they obtained punched card equipment as an interim measure to help the data processing task. At the end of the year a small group started work on *LEDAC's* mercury filled delay-line memory[14].

Nearly a year later the LRWE Board considered the electronic computer question, cancelled the *LEDAC* project and directed that a UK Ferranti *MARK 1* was to be purchased. In January 1952 Ferranti's price quadrupled to £195,000 and this was cancelled too.

Maths Services took a step back and realised that the punched card equipment was not delaying the overall processing as the initial data tabulation was so slow. They surveyed similar US establishments then bought new tape recorders, and movie film viewers with built in card punches[15].

However, they knew that the rate of rocket trials was going to increase sharply. Planning for 1955 and 1956 suggested that 400,000 position points, 6 million telemetry samples and 600 km of film would need to be processed, and around 200 lady Computers would be needed. They revisited the electronic computer question.

Back in 1950 a young Queensland mathematician, John Allen-Ovenstone[16], had joined Maths Services, then headed off to Cambridge to do a doctorate. He had used Pearcey's CSIRO computer then Cambridge's first computer *EDSAC*. When he returned to LRWE in 1953 he developed a broad specification for a computer and the various converters that would be required[17]. This document described input converters for rocket telemetry, the new doppler radar, and tracking radar. The converters were to re-code the data and write magnetic tapes suitable for the computer. Kinetheodolite film viewers were to be modified to punch paper tape rather than cards. There should also be output converters to accept a magnetic tape from the computer to print tabulated data and draw graphs. Clearly the computer itself had to handle a great deal of input-output, but Allen-Ovenstone's document did not specify great calculating power. All told these were very bold requirements!

The specification document was sent to British computer manufacturers, and Allen-Ovenstone with experienced engineer George Barlow travelled to the UK in early 1953. They visited English Electric who had just completed a prototype of their first commercial machine, the *DEUCE*. However, this didn't yet have software or any performance statistics. It also couldn't be delivered before mid-1955! The other

---

13 Peter Morton 1989 *ibid*. LRWE's Frank O'Grady proposed the contact with Trevor Pearcey at CSIRO.

14 Peter Goddard private communication Dec 2007.

15 Peter Morton 1989 *ibid*. W.A.S.Butement, R.W.Boswell & R.P.Bonnell travelled through the USA. They recommended Ampex tape recorders and Benson-Lehner *Boscar* ballistic film analyser and recorders.

16 He later dropped the hyphen.

17 Peter Morton 1989 *ibid*. The document can be identified as 'J. Allen-Ovenstone, Notes on data processing at LRWE, 9 Oct 1953' (21pp) but this hasn't been located in Australian Defence archives, nor in UK computing archives.

available computer was Ferranti's *MARK 1*. They found this a 'neat' machine, but it was not constructed of plug-in units which they considered essential for servicing. In addition they suspected that its electrostatic memory would be sensitive to radio transmissions present at Salisbury[18].

A UK computer company they apparently didn't visit was Elliott Brothers. Elliotts had championed plug-in electronics for their WW2 Naval radar and continued this in their first significant computer, the *401*. At the same time as the UK visit, the University of Sydney requested information about this machine – and their London contact arranged for this, and sent a second copy to LRWE[19].

With the number of rocket trials rapidly increasing, and still no computer, Barlow and Allen-Ovenstone returned to the UK in mid-1954, and the head of Maths Services, Major Jacoby, travelled to New York for a demonstration of IBM's first large electronic computer, the *701* 'Defence Computer'. Jacoby was very impressed[20].

Barlow and Allen-Ovenstone returned to Ferranti for a preview of their new machine, the *PEGASUS*. They liked the design but thought it was more complicated than LRWE needed, and Ferranti was not interested in building the simpler machine Allen-Ovenstone wanted[21]. Then they followed up the *Elliott 401* brochure and received a much more encouraging response. Historian Simon Lavington wrote in *Moving Targets*:

> *Laurence Clarke believes that the basic design of ... [LRWE's computer] was carried out in the space of about a week after the initial visit of Ovenstone and Barlow ... Laurence was in North Wales on holiday visiting family and was called back ... because of the importance of the project*[22].

At this time Elliott Bros. were working on their first commercial computer, their *402*. In addition they had a series of small projects which might be related to the *402* and they administratively grouped these as project *403*. The preliminary work for LRWE was based on the same plug-in logic modules as the *402* and identified under the *403* odd jobs classification[23].

Back at LRWE the discussions must have been interesting. Jacoby recommended the *IBM 701* though it used similar electro-static memory to the rejected *Ferranti*

---

[18] Peter Morton 1989 *ibid*.

[19] National Archives of the History of Computing at John Rylands University Library, The University of Manchester: National Research Development Council records: NAHC/NRDC/C15/8 "Computer 401 Elliott Bros, Publicity". Correspondence 1953-55. 3 letters on 13 May, 1953, NRDC to Prof D.M.Myers, University of Sydney; NRDC to W.A.S.Butement, Dept. of Supply, Melbourne; and NRDC to W.R.Blunden, Military Board, Australia House, London.

[20] Peter Morton 1989 *ibid*. Jacoby was also involved in the *LEDAC* project.

[21] Bernard B Swann "The Ferranti Computer Department" (1974, unpublished manuscript in the John Bennett collection of the Australian Computer Museum Society)

[22] Simon Lavington "Moving Targets" (Springer 2011). Laurence Clarke became Technical Director of Elliotts.

[23] NAHC/NRDC/C13/3 'Electronic Computers, Elliott Bros Ltd, Development, File No 3'. Correspondence 1952-53. NRDC Internal Memo 27 August 1953. Has considerable detail about Project 403 but it is not what became the Elliott 403 / WREDAC.

MARK 1, while Barlow and Allen-Ovenstone had found the Elliotts people very responsive. The final choice for Elliotts may have owed more to their lower price than the technical arguments.

> *On 29 September 1954, the first manufacturing progress meeting for ... [LRWE's computer] was held at Elliott's Borehamwood Laboratories, 'amid a flurry of telexes querying and replying to minor details'. A crucial matter was the interface between the Woomera instrumentation's data-converters ... [and the computer]*[24].

But progress wasn't exactly rapid. An internal memo the following February, five months later, included:

> *... while I was at Borehamwood, I was shown the state of progress on the 402 and 403 machines. ... Adherence to this modular principle has enabled Elliott Brothers to design rapidly and easily the 403 machine which is the special machine to be delivered to the Long Range Weapons Establishment in Australia.*
> *The 403 will comprise sixteen cabinets ...*
> *Assembly of the 403 has not yet commenced ...* [25]

While 'Elliott 403' remained the official name, it was often referred to within Elliotts as *COBBER*. Then in January 1955 LRWE was amalgamated with two other research laboratories that had been created on the Salisbury site, and the new organisation was christened the *Weapons Research Establishment* - WRE - and from then on the computer was *WREDAC*[26]. The machine Elliotts was building was: [27]

Processor	Technology	Bit serial operation using vacuum-tube logic, 34 bit words, and 333 KHz clock	Constructed from 640 standard Elliott plug-in units incorporating about 1,600 tubes
	Instructions	2 per word, single address, with separate formats for arithmetic and input/output	48 instructions including **+ - × ÷** transfers, input/ output, indexing etc.
	Memory	512 words of magneto-strictive nickel delay-line fast access working store	(= 2 KBytes)
	Speed	average 600 microseconds per instruction	(= 1.6 KHz)

---

24 Simon Lavington 2011 *ibid*. These quotations refer to the computer as 'WREDAC' which it wasn't then called.

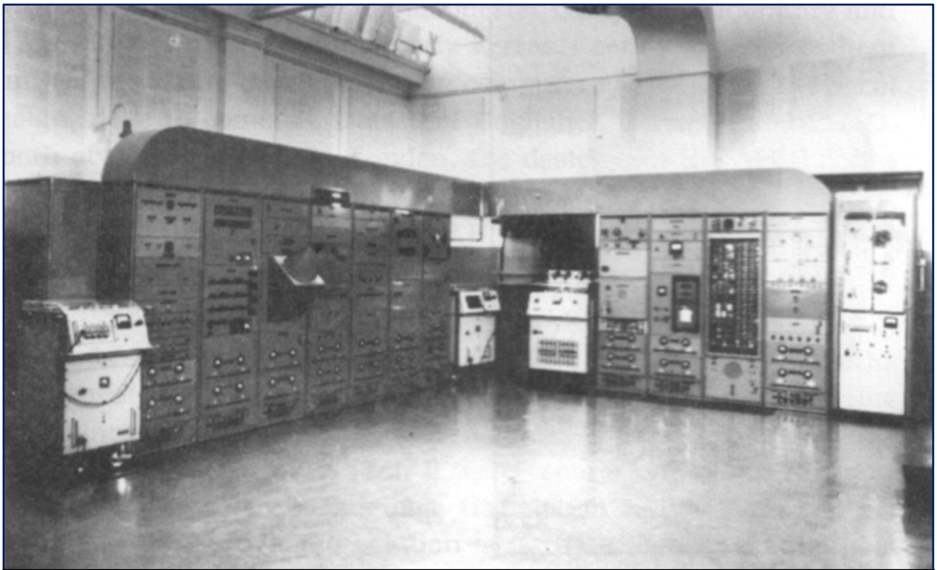
25 NAHC/NRDC/C13/8 'Computers, Elliott Bros, Manufacturing Programme Contract No 3'. Correspondence 1953-65. NRDC Internal Memo 15th February, 1955.

26 'Weapons Research Establishment Digital Automatic Computer' of course.

27 Digital Computer Newsletter Vol.7 No.4 Oct 1955 (Office of Naval Research, U.S. Navy). A. St.Johnston 'A Series of Computers using plug-in units' Proc. IEE Vol.103 Part B Supplement 2 Convention on Digital-Computer Techniques pp186-7, April 1956.

Program medium	Paper tape	5 channel teletype reader and punch		Instructions were punched in a symbolic form
I/O		Disk	46 cm diameter disk rotating at 2300 rpm with 80 read/write heads providing 16 K words of backing store	(= 68 KBytes)
		Magnetic Tape	2 units, 1/4 inch tape ancillary store written at 100 bits per inch & similar to EDSAC's	(a 2400 foot tape could hold about 250 KBytes)

This was just the processor, 16 cabinets nearly 2m tall and formed in a big 'U' shape about 6m wide with 3m arms. Plus power supplies and air-conditioning plant.



The *Elliott 403*'s Input Converters: Rocket Telemetry Converter (left) and Doppler Radar Converter (right) – from T. Pearcey 'A History of Australian Computing' – Chisholm Institute of Technology 1988)
Photo: Courtesy, Defence Science & Technology Organisation

Elliotts also developed an 'Output Converter' for WRE. This separate system had a third magnetic tape unit to read processor output, 7 large cabinets of vacuum-tube logic and 5 output devices. The largest was a line printer by Bull of Paris which could print 2 lines of characters per second. Then, remarkably, there were 4 graph plotters. These were *Mufax* facsimile machines, modified by Elliotts, for which the converter synthesised data streams at 4 lines of dots per second. Printer and plotters used treated paper which had to be developed by the *Dyeline* process[28 29].

[28] The result is not permanent and there seem to be no extant examples of either output system.
[29] 'Data Processing System - The Australian Weapons Research Establishment - Salisbury, S. Australia', Communications of the ACM. Vol.1 No.7 July 1958.

The third major part of the overall system was a pair of Input Converters developed by George Barlow, Leo Cohen and Fred Thonemann at WRE. Rocket telemetry tapes recorded at the range were read, processed by another 7 cabinets of logic and written to a magnetic tape compatible with *WREDAC*. The Doppler Radar Converter accepted the second type of range recording and via a further 4 cabinets processed this into a form *WREDAC* could use[30].

To keep track of Elliotts' work WRE engineer Jack Bowie stayed in the UK for some months up to mid-1955. Simon Lavington continued Laurence Clarke's recollections:

> *Bowie was always threatening us with a 'crutching knife' if the project ran late. This is a horrendous tool used by the sheep shearers!* [31]

The *403* was due to be completed in June 1955, and it arrived by sea in Adelaide in September 1955 accompanied by three Elliotts technicians and a large kit of spares. But not the Output Converter, printer or plotters. The basic installation was completed during November, but there were problems. One type of tube kept failing in the memory electronics, and the cooling system did not really cope as temperatures passed 35ºC. [32]

The Output sub-system was shipped in May 1956 and by mid-year *WREDAC* passed its acceptance tests and was handed over to Allen-Ovenstone and his team.

The *Elliott 403*'s Output Converter in the UK in 1955: 4 Mufax plotters, Bull Line printer, Output processor, Magnetic Tape input (from the Elliott's archive at BAE Systems and Telent PLC UK)
Photo: Courtesy, Defence Science & Technology Organisation

---

30 R.W.M. Boswell 'Guided Flight Trials' (J. Royal Aeronautical Society, Vol.62 June 1958).

31 Simon Lavington 2011 *ibid*.

32 Simon Lavington 2011 *ibid*. The tubes were '12AT7 twin triodes' operating too close to their design limits. This was never fixed but WRE engineers eventually screened new valves for the best performers.

3 Using WREDAC

Meanwhile, Allen-Ovenstone had written 'An Introduction to Programming for Automatic Digital Computers' which included detailed explanations of programming mathematical functions, and everything one needed to program 'the WRE DAC'.[33] He also wrote the basic system software which was a very compact symbolic program loader on the same principles as EDSAC's. [34]

It is likely he wrote the initial trials data processing programs, and by September an enthusiastic report was presented at a symposium 'Automation and Australia' at the NSW University of Technology[35] in Sydney. WRE's Principal Scientific Officer, Fred Thoneman discussed UK business computing then gave a very clear overview of WRE's activities:

> *... much experimentation, engineering work and engineering testing involves the making of a very large number of measurements and organising them into coherent and intelligible forms. ... It's the kind of job that data-processing automata are excellently fitted to do, if, and this is an important point, if there is enough data to justify the use of these automata. They cost a lot of money, at least they do at present. ...*
>
> *[We] have been almost obliged to take advantage of high-speed automatic processing. A large part of our work involved with guided weapons is what might be called the mass production of measurements, and the introduction of data-processing automata in this field was motivated by the desire to get our flight-trial results quickly into a clearly surveyable form. The sheer bulk of raw data generated at high rates, and very high rates these are, by elaborately instrumented guided weapons, requires that the system be fully automatic. If you leave a man, no matter how quick he is, actively engaged in the routine daily processing work, he must necessarily slow the entire system down to human rates, and the factor is somewhere about the order of 10 to 100,000 to 1. ...*
>
> *We will imagine that the observing instrument is in a missile. ... This is coupled in the missile to an electrical oscillator whose frequency it controls. It is necessary, of course, because the missile is inaccessible, to telemeter, by radio-link, the information of the out-put of that detecting device to ... a ground receiver. ... The telemetry system is time-shared, that is, the output from many observing instruments are sent sequentially along the same transmission channel, and ... we have to deal with many thousands of points or data a second. ...*
>
> *In the automatic system, the received data is recorded directly on magnetic tape ... The data is played back at a slower speed into a fully automatic measuring device which measures the data as it comes along, sample by sample, and codes those measurements in binary-digital*

---

[33] J. Allen-Ovenstone 'An Introduction to Programming for Automatic Digital Computers' WRE TM 50, 1955.

[34] Peter Main, private communication Dec 2007.

[35] Later the University of NSW.

> *code. ... the machine will work for hours, even days, at a time, and if it does break down the break-downs can be fairly easily repaired. These measurements, which are now recorded on magnetic tape in binary-digital code, are played into a high-speed electronic digital computer of the type of which this University has recently become the proud possessor*[36]*. That is where the calculations are done on this data, and the output of the computer is then recorded again on a third magnetic tape, and also, as willed, can be tabulated very rapidly using a fast printer ... or, by using another type of decoder, caused to plot graphs at very high speed on anything up to 12 simultaneous channels. ... The pay-off of this system can be expressed very simply and briefly. To process completely 10,000 data by relatively up-to-date manual methods required a total of 20 working hours. The automatic system does the entire job in a total time of fifteen minutes. Both systems are equally accurate.* [37]

At the end of 1956 Allen-Ovenstone's Quarterly Report included:

> *For the first time since the range commenced operation there was no backlog of trial calculation over the Christmas period and, despite the shortage of skilled programming and maintenance staff, a reasonable service to the establishment was maintained.* [38]

This was very positive, but there were reliability problems. They were keeping track of performance and the first period reported was from acceptance in July 1956 to March 1957: [39]

Average Total Time Available/Week	40 hrs	*- ie 1 shift operation*
Average Scheduled Maintenance/Week	10 hrs	*- 2 hrs at the start of each day*
Average Unscheduled Maintenance/Week	8 hrs	*- ie faults*
Average Effective Time/Week	20 hrs	
Average Standby Time/Week	2 hrs	*- time with no jobs to run*
Efficiency Factor = $\frac{\text{Effective Time}}{\text{Total Time}}$	50%	

That 'Effective Time' looks like a concern but the required work was being done.

One might think that the Maths Services Group had enough to do, but no. They could have been inspired by the 'Automation and Australia' symposium, or by the Cambridge *EDSAC* team's support of the Lyons Electric Office (*LEO*) business computing initiatives, but Allen-Ovenstone became the driving force behind planning

36 'UTECOM', an English Electric *DEUCE*.

37 F. Thonemann 'Automatic High-speed Data Processing' in 'Automation and Australia - proceedings of Symposium at NSW University of Technology 11 & 12th September 1956' (NSW University of Technology)

38 J. Allen-Ovenstone 'WREDAC quarterly report 1.10.56 to 20.12.56' SA5398/11/2, reproduced in Peter Morton 1989 *ibid.*

39 'Data Processing System - The Australian Weapons Research Establishment - Salisbury, S. Australia', Communications of the ACM. Vol.1 No.7 July 1958.

an international computing conference for mid-1957. A direct cause was given in the conference proceedings – WRE had been providing advice on computers "*on a consulting basis, with their application to office procedures for banks, insurance companies and general administration of large undertakings*". WRE gave three objectives for the conference:

(a) to tell the many interested people and organisations of our experiences and to show them our system and equipment and demonstrate our operations;
(b) to get intelligent and constructive criticism of our methods from those versed in the art with a view to improving our techniques and operations;
(c) to provide sound advice from our own staff, and the many renowned visiting delegates and lecturers, to those business and technical delegates interested in the use of such machines in their own spheres of activity. [40]

For the first of these objectives John Ovenstone[41] wrote two demonstration programs to run on WREDAC:

Stock and Inventory Control – to model a business producing 500 items manufactured from 1,000 catalogued components and to demonstrate tracking of stock levels, production changes and costs.

Hourly Payroll Calculations – which modelled a business with complex employee records, daily employee details updates, and weekly hours worked, then it produced pay lists and total wage details.

For each of these he considered many practical issues, such as the time required to prepare the basic records and the regular update procedures. He created a suite of sample records and documented the programs with flowcharts. Ovenstone also prepared a paper on computing for business accounting.[42]

The Conference was enormous, and it was split into three sections: *Programming and Mathematics*, *Engineering*, and *Business Applications*. There were talks by visitors from:

UK: Ferranti Ltd., EMI Electronics Ltd., National Cash Register Co., English Electric Co., British Tabulating Machines Co., University of London, Cambridge University, University of Manchester, University of Leeds, Royal Aircraft Establishment, Royal Radar Establishment, National Physical Laboratory,

USA: IBM Pty. Ltd., Burroughs Ltd., University of Michigan, RCA Patrick Air Force Base,

---

[40] H.J. Brown, 'Introduction' to 'Data Processing and Automatic Computing Machines, Proceedings of Conference held at Weapons Research Establishment, Salisbury, S.A. 3rd - 8th June 1957' (Commonwealth of Australia Department of Supply 1958).

[41] About this time he dropped the hyphenated surname.

[42] J.A. Ovenstone, 'Business and Accountancy Data Processing' and 'Demonstration Problems on the WREDAC System' in 'Data Processing and Automatic Computing Machines, Proceedings of Conference' 1957 *ibid.*

> and locally: University of Adelaide, University of Sydney, NSW University of Technology, University of Melbourne, University of Tasmania, CSIRO, Snowy Mountains Hydroelectricity Authority, Kalamazoo Ltd., and WRE.

There were 8 talks involving WREDAC, 14 related to the other Australian computers (CSIRAC, UTECOM and SILLIAC), 8 on UK computers and another 8 'advertising' commercially available equipment. Plus 8 talks on the older analogue computing techniques and a collection more on various technical issues. Overall 68 papers were presented over the 6 days of the conference along with discussions and demonstrations.

Initial papers repeated the usefulness of WREDAC and its associated systems and added:

> Other applications of the WREDAC computer have been exploited outside the field of guided missiles. For example, it has already been applied to a variety of calculations arising in aerodynamics and aircraft structures, and now the possibility of its use for business calculations is under consideration. An officer of the Public Service Board has spent some time here studying the WREDAC in order to report to the Board on the possible applications of high speed digital computers in Government Departments. [43]

Clearly the conference was a great success, and with hindsight the proceedings provide a remarkable description of the start of British and Australian digital computing, along with papers that had surprising influences subsequently[44].

George Barlow, who was deeply involved in developing the WREDAC system, was fascinated by one presentation in particular. He wrote later:

> At the time we knew we were ahead of the British ... To my delight, I found that while RCA, who operated the [missile test] range at Cape Canaveral, had both Telemetry and Doppler Converters, neither worked. ... Bill Boswell was so pleased that he dispatched Bill Watson and myself to a conference in New York held by the Instrument Society of America, which had a session scheduled on Automatic Processing of Trials Data ... We were just in time to hear the chairman say, 'Well, we all agree that range data is a fruitful ground for automatic processing, but has anybody actually done it?' Bill nudged me, and I managed to stammer out that we had automatically processed telemetry data - in Australia. There was a stunned silence. [45]

Whether John Ovenstone was inspired by the conference, or frustrated by his programming experience, he had instigated a series of improvements to WREDAC by the end of 1957. The first was a much enlarged operator's console. The original was an array of switches and two modest display tubes built into a processor cabinet which had to be used while standing. The replacement was an imposing desk with

---

[43] B.G. Gates, 'Opening Address' in 'Data Processing and Automatic Computing Machines, Proceedings of Conference' 1957 *ibid.*

[44] eg. Charles Hamblin's description of *Reverse Polish Notation* which led to stack based computing and influenced the design of hand held calculators.

[45] G.E. Barlow quoted in Peter Morton 1989 *ibid.*

switches, dials and two television sized screens. This would certainly have provided a much more comfortable program debugging environment! [46] In addition he specified additions to the processor's logic. Three new instructions would be provided to modify the operation of existing instructions:

'Use B-lines' allowed for multiple groups of address modification, or indexing hardware, i.e. more complex handling of tabulated data,

'Use Logic' would support new hardware for floating-point arithmetic, and for business oriented character operations, and

'Use Units' would allow more magnetic tape drives to be installed, along with multiple paper tape readers and punches, without changing existing input/output instructions. [47]

Ovenstone's vision and ability to get things done did not go unnoticed. In 1958 he was promoted out of WRE to First Assistant Secretary for the Department of Defence where he worked at converting many administrative tasks to computer operation - but that's another story. [48]

As experience was gained WREDAC was used for many tasks beyond missile trials data analysis. Reported work included:

- CSIRO's *Division of Mathematical Statistics* had regular access to carry out a large-scale climate change study, up to the end of 1959. [49]
- WRE's *Systems Assessment* and *Aerodynamics* divisions had missile flight data reprocessed and converted to analogue form for further analysis on their simulator.[50]
- Another CSIRO Division, *Soils*, made notable use of WREDAC to process x-ray scattering data for the first stage of determining the atomic structure of mica. [51]
- In 1960 WRE's new *Theoretical Supersonics Group* programmed a simulation of missile wing behaviour. [52]
- University of Adelaide MSc student John Sanderson wrote a high-level language interpreter for WREDAC. [53]

In mid-1958 the Maths Services folk summarised WREDAC's performance:

> *... about 18 months and 30 man-years of effort were required before the automatic system completely replaced the corresponding manual processes. The total elapsed time between a trial and the presentation*

---

46 The new console position is shown in WRE drawing MSK56 of 22 October 1957. Details of the consoles may be obtained from undated photos in Peter Morton 1989 *ibid* and J.M. Bennett et al 'Computing in Australia' Hale & Iremonger 1994.

47 'Data Processing System' CACM. Vol.1 No.7 July 1958 *ibid*.

48 Peter Morton 1989 *ibid*.

49 Research Review of the CSIRO 1960. Accessed through Google books February 2012.

50 White, Overheu & Wheadon "A report on data processing at WRE", WRE TRD2, 1959 (reported in Peter Morton 1989 *ibid*).

51 E.W. Radoslovich (CSIRO Div Soils) "The Structure of Muscovite" Acta Crystallography 13/919 1960 (reported in Peter Morton 1989 *ibid*).

52 Y.Y. Chan "Theoretical pressure distributions on a series of supersonic wing-body combinations" WRE TM HSA 85-90 1960.

53 J.G. Sanderson, "Automatic Programming for Digital Computers" University of Adelaide MSc Thesis, 1957.

WREDAC's new console about 1958
(from T. Pearcey "A History of Australian Computing" - Chisholm Institute of Technology 1988)
Photo: Courtesy, Defence Science & Technology Organisation

of final results has been reduced to one-tenth of that required by the manual system, the cost of processing trials information has been reduced by a factor of at least 20. The peak load capacity of the automatic system is about 50 times that of the manual system, and the complete cost of the automatic system has been recovered in the first two years of operation. [54]

The report continued with figures showing increasing use and improving usefulness:

	July 1956 - March 1957	March 1957 - Dec. 1957	Dec. 1957 - March 1958
Average Total Time Available/Week	40 h	57 h	70 h
Average Scheduled Maintenance/Week	10 h	10 h	10 h
Average Unscheduled Maintenance/Week	8 h	6 h	8 h
Average Effective Time/Week	20 h	40 h	50 h
Average Standby Time/Week	2 h	1 h	2 h
Efficiency Factor = $\frac{\text{Effective Time}}{\text{Total Time}}$	50%	70%	72%

By late 1960 this last efficiency factor was ‘above 80%’. [55]

4 WREDAC’s End

WREDAC was performing well but new requirements were on the horizon. The UK was developing an intermediate range ballistic missile, the *Blue Streak*, which would be capable of carrying a nuclear weapon. It would be tested at *Woomera* and a great deal of new range equipment would be needed, possibly including as many as 50 high performance tracking cameras. A 1959 report stated:

new electronic digital computing facilities capable of coping with 100 times the 1958 computational load must be acquired urgently. [56]

This report went on to carefully justify this speed factor and then to give some benchmarks for available transistorised, i.e. reliable, computers:

Operating speed relative to WREDAC

Machine / Calculation	WREDAC	IBM 7090	TRANSAC S-2000	EMIDEC 2400
Accepted Measure	1	68	27	21
Polynomial	1	74	60	20
Square Root	1	63	32	34
Table look-up	1	145	67	34
Scaling of number	1	76	64	21
Runge-Kutta	1	52	35	16

---

54 “Data Processing System” CACM. Vol.1 No.7 July 1958 *ibid.*

55 Peter Morton 1989 *ibid.* Trials superintendent J. Clegg reported WREDAC’s reliability above 80%.

56 White, Overheu & Wheadon “A report on data processing at WRE”, WRE TRD2, 1959 (quoted in WRE TM TRD 35).

Clearly the above figures indicate ... that there is only one available machine able to cope with the expected computational load.

The IBM 7090 promised a great deal of available software, and the new standardised FORTRAN language. In March 1960 the WRE Board of Management voted to hire the 7090 at US$330,000 a year.

Five weeks later the *Blue Streak* was cancelled.

However, missile trials continued. The IBM 7090 was delivered at the end of 1960 and handed over in February 1961. The two computers were run in parallel while the required suite of new programs was developed. Telemetry processing and missile range data conversion were updated and the 7090 took over trials analysis, as indicated in the following table. [57]

	Trajectory points calculated per day	Cost of each calculated point
1955-56 (pre WREDAC)	800	A$4.00
1956-57 (WREDAC)	1000	50c
1963-64 (IBM 7090)	3000	10c

WREDAC continued to be used too, with new programs ranging from analysis of wind tunnel tests of missiles, to statistics and mathematical work.[58] However, after only 7 years' use WREDAC still needed constant maintenance and looked very old next to the sleek transistorised IBM 7090.

At the end of 1962 WRE Principal Research Scientist George Barlow recommended that WREDAC should be disposed of, and that it might be offered free to the South Australian Institute of Technology. Peter Morton wrote:

Its ultimate fate is hazy. A few parts did go back to Britain as spares, but most of it probably ended as scrap. [59]

The old machine wasn't missed, as Peter Morton continued:

The excellent reliability and much greater speed of the new IBM 7090 helped the growth of programming skills within WRE. ... The new computer played an important role in WRE's support of the American space programs, and it was vital to the success of the ambitious ELDO [European Launcher Development Organisation] firings of the mid-1960s. [60]

---

[57] Peter Morton 1989 *ibid*.

[58] Knight, D. C. "Wredac programmes for the reduction of wind tunnel force-measurement data", Jul 1960 WRE-TRD-TM-53. Nicholls, L. A.; Phillips, A. D. "Some facilities on WREDAC for the analysis of stationary time series", Aug 1960 WRE-TRD-TM-52. Overheu, D. L. "Some programs for the conjugate gradient methods for the solution of linear simultaneous equations", Jul 1961 WRE-TRD-TM-57A.

[59] George Barlow biography from http://www.eoas.info/biogs/P003358b.htm (March 2012) and Peter Morton 1989 *ibid*.

[60] Peter Morton 1989 *ibid*.

Appendix 1 - Some of the People Associated with WREDAC

John Allen-Ovenstone	Born 1925 in Sydney. During WW2 John Allen-Ovenstone served with the AIF in S.E. Asia and then joined the NSW Police force. In 1948 he enrolled at the University of Sydney and obtained a BSc with 1st class honours in mathematics and physics. He also used the CSIRO's first computer (CSIRAC) being developed on the University site. Following graduation he joined LRWE and was sent to Cambridge to complete his maths studies. There he used their first computer, EDSAC, which had just been completed. He was awarded a PhD in 1953 and returned to WRE where he specified, supported and championed their first computer (WREDAC). He rose to the rank of Principal Scientific Officer responsible for missile test data processing. In 1958 he transferred to the Department of Defence where he introduced data processing and a vision of integrated, communicating systems. From 1964 he was foundation Professor of Computing Science at the University of Adelaide. In 1971 he became Managing Director of business consultants Inbucon (Australia). Following recovery from severe injuries in a car crash he had a series of government jobs up to retirement in 1983. He died in 1984. [61]
George Edgerton Barlow	Born 1924 in Melbourne. At Melbourne University he received a MSc in Physics in 1947 and joined LRWE. He received training in the UK then from 1949 worked solidly on computing related electronics. Initially this was analog processing, then LEDAC and WREDAC. He designed the important missile telemetry converter with Leo Cohen and Fred Thonemann in 1953. In 1968 he transfered to DSTO Canberra and from 1980 until his retirement in 1987 he was Deputy Chief Defence Scientist. He died in 2005. [62]
Robert William McGregor (Bill) Boswell	Born 1911 in Melbourne. For his 1935 MSc at Melbourne University he used radio direction finding to trace thunderstorms across southern Australia. He continued this with CSIRO, then joined the PMG which sent him to the UK for further study. During WW2 he joined the Navy and worked on radar. In 1946 he joined the Department of Civil Aviation as senior engineer. Then, in 1948 he became Principal Scientific Officer at LRWE and headed the electronics group. The next year he was put in charge of the test range. He joined the CSIRO Committee on Mathematical Instruments (ie computers), toured US and UK missile test ranges and in 1956 received an OBE. From 1958 he was Director of WRE and known as 'Mr Rocket Range'. In 1965 he left WRE to become Secretary to the Department of National Development and Chairman of the Snowy Mountains Council. Then in 1969 he was Deputy High Commissioner in London and returned to be Chairman of the Australian Atomic Energy Commission. He died in 1976. [63]

61 Who's Who in Australia 1965 (Colorgravure Publications), Encyclopedia of Australian Science (http://www.eoas.info/biogs/P003832b.htm), "Outstanding scientist dies" The Australian 24 Jul 1984.

62 Encyclopedia of Australian Science (http://www.eoas.info/biogs/P003358b.htm), "Developing the science to defend the nation" Sydney Morning Herald 25 Nov 2005.

63 Australian Dictionary of Biography (http://adb.anu.edu.au/biography/boswell-robert-william-mcgregor-9547), Encyclopedia of Australian Science (http://www.eoas.info/biogs/P001605b.htm)

Appendix 2 - Summary of WREDAC's Instructions

This is a summary of 'An Introduction to Programming for Automatic Digital Computors' written about 'the WRE DAC' by John Allen-Ovenstone in 1955 as Technical Memorandum WRE TM50. This document is 73 pages long, so detail and extensive examples have been skipped.

Programming Notes:

- Instructions occupied half a memory word (17 bits), and the low order half ('even') was executed before the high ('odd') instruction.
- There were two instruction formats, 'arithmetic' (identified by its low order bit = 0) and 'transfer' (low order bit = 1).
- Run time addresses could be modified by adding the contents of a B-line register (or index) specified as α 0 i.e. no modification, β B-line 1, γ B-line 2, δ B-line 3. B-lines 1 to 3 occupied memory locations 1 to 3.
- Program tapes were punched in a 5 bit teletype code:

Code	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Letter	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	α	β	γ	δ	φ
Number			1		2	3			4	5		6					7	8		9	/			0							

The program loader interpreted operation code field as letters and addresses as decimal numbers.

- Numbers were stored with values -2 to +2 in 34 bits as sign, unit, 32 bit fraction.

Arithmetic Instructions:

Operation	Address	B-line	Type
5 bits	9 bits	2 bits	'0'

Here *b* represents the B-line which may be α 0, β B-line 1, γ B-line 2, δ B-line 3.
A accumulator, R multiply register.
(addr) implies the contents of location addr.
addr may be modified, see under cues below.

As punched	Op	Description	As punched	Op	Description
CH *b*	0	Check, stop if the console 'Stop' key is depressed	CA *addr b*	16	Clear and add A = (addr)
GE addr b	1	Go to even addr	CS addr b	17	Clear and subtract A = - (addr)
PO addr b	2	If A positive go to addr odd	SW addr b	18	Swap A and (addr)
PE addr b	3	If A positive go to addr even	CL addr b	19	Clear to A, (addr) = A then clear A
NO addr b	4	If A negative go to addr odd	AD addr b	20	Add A = A + (addr)
NE addr b	5	If A negative go to addr even	SA addr b	21	Subtract A = A - (addr)
JO addr b	6	If B-line 1 ≠ 0 increment its address bits and jump odd	CT addr b	22	Collate A = A AND (addr)
JE addr b	7	ditto even	AS addr b	23	Add subtract A = (addr) - A
LD addr b	8	Long division A = A + R / (addr)	SB addr b	24	Set B-line, if addr>0 (1) = addr else (1)=512 - |addr|
SM addr b	9	Set multiplier R = (addr)	CO addr b	25	Clear to odd, (addr)odd = high A, clear A
MA b	10	Multiply and add A = A + A × R	CE addr b	26	Clear to even, (addr)even = high A, clear A
MS b	11	Multiply and subtract A = A - A × R	ST addr b	27	Store (addr) = A
SL addr b	12	Shift left (A) by addr places	AO addr b	28	Add odd, high A + (addr)odd

SR addr b	13	Shift right	SO addr b	29	Subtract odd, high A - (addr)odd
NM b	14	Normalise, shift A till |A| between 1 and 2, store count in B-line 1	AE addr b	30	Add even, high A + (addr)even
MD b	15	Modulus, A = |A|	SE addr b	31	Subtract even, high A - (addr)even

Transfer Instructions:

Tape unit	Operation	Memory block	Disc track	Type
1 bit	4 bits	3 bits	8 bits	'1'

Here there is no B-line modification and all transfer orders end with α. Block transfers are for 64 words.

As punched	Op	Description	As punched	Op	Description
	0	spare	R T d α	8	Read magnetic tape, one word from drive d to A
	1	spare	W T d α	9	Write magnetic tape, from A to drive d
I L α	2	Input paper tape to 5 low bits of A	B b T d α	10	Block to tape transfer, from main store block b to drive d
I M α	3	Input paper tape to most significant 5 bits of A	T d B b α	11	Tape to block transfer, from drive d to main store block b
O L α	4	Output from least significant 5 bits of A to paper tape	T d H α	12	Tape drive d halt
O M α	5	Output from most significant 5 bits of A to paper tape	T d G α	13	Tape drive d go
B b D t α	6	Block to disc transfer, from store block b to disc track t	T d F α	14	Tape drive d forward
D t B b α	7	Disc to block transfer, from disc track t to store block b	T d R α	15	Tape drive d reverse

Program Loader Cues:

As punched	Description
QA *m* α	Set next order address to *m* and store *m* in 7
QC α	Obey the following command as soon as it is assembled (especially 'QC α GE *startaddress* α' at the end)
QP *m* α	Place next order in word *m* as a parameter and continue storing whole words
QS α	Place next order in the address from the last QA
QN *m* α	Read next 7 characters as a binary fraction and store in *m*, repeat till 7th char is odd

The program loader (or initial orders) occupied 64 words and set a number of memory locations. It could modify an instruction's address to support standard subroutine loading by defining locations identified by A, D, G, H, K, M, N, S or U to be set to a load address. Then, instructions with this letter appended to their address field would be modified at load time, e.g.

QA 300 α sets the load address to 300 and places 300 in load variable 'A'
CA 5 A α is loaded into 300 with its address modified to 305.

Memory set by the loader:

1 to 3 (i.e. Bline 1 to 3) = 0	$6 = 1\times2^{-3}+1\times2^{-16}$
$4 = 1\times2^{-32}$ round off	Load variables: 7 A, 8 D, 9 G, 10 H, 11 K, 12 M, 13 N, 14 S, 15 U
$5 = 1\times2^{-29}$ unit address	

During 1958 WREDAC's processor was modified along the lines recommended by John Ovenstone. The new head of Mathematical Services, Don Overheu, then wrote 'An Introductory Coding Manual for the WRE Automatic Computer' (WRE TRD39). The following notes are based on the differences this shows from the preceding section.

Revised Arithmetic Instructions:

CH *addr b*	0	If *addr* ≠ 0 and A=0 goto addr odd	- Replacing the conditional halt.
LD *addr b*	8	Long division A = A + (addr) / R	- Inverting the sense of the division.
UL *spec* α	23	Use Logic as spec given by Mode, ABC:	- Replacing AS (Add subtract).
		Mode (3 bits) = 1 for Use B-line logic (the only Use-logic implemented) A (2 bits) specify which 2 B-lines are to be added before use as modifiers: 0 use indicated B-line, 1: 1+2, 2: 2+3, 3: 3+1 B (2 bits) specify which of the 4 sets of B-lines will be used: 0, 1, 2 or 3 C = specify the B-line for SB, JO, JE and NM, 0 none, 1, 2 or 3	

Also addresses may be written X / Y where X is the block (0 to 7) and Y the offset (0 to 63).

All Transfer Instructions Were Revised:

Operation	Block address	Sector address	Type
5 bits	3 bits	8 bits	'1'

Here there is no B-line modification and all transfer orders end with φ. Block transfers are for 64 words.

As punched	Op	Description	As punched	Op	Description
	0	spare	WT 0 φ	16	Write magnetic tape channel 0 from A
	1	spare	WT 10 φ	17	Write magnetic tape channel 1 from A
IL φ	2	Input paper tape and add to 5 low bits of upper A	BT 0 *b* φ	18	Block to tape transfer, channel 0 from main store block *b*
IM φ	3	Input paper tape and add to most significant 5 bits of A	BT 10 *b* φ	19	Block to tape transfer, channel 1 from main store block *b*
OL φ	4	Output least significant 5 bits of upper A to paper tape	RT 0 φ	20	Read magnetic tape, one word from channel 0 & add to A
OM φ	5	Output most significant 5 bits of A to paper tape	RT 10 φ	21	Read magnetic tape, one word from channel 1 & add to A
BD *b* / *s* φ	6	Block to disc transfer, from store block *b* to disc sector *s*	TB 0 *b* φ	22	Tape to block transfer, from channel 0 to store block *b*
DB *b* / *s* φ	7	Disc to block transfer, from disc sector *s* to store block *b*	TB 10 *b* φ	23	Tape to block transfer, from channel 1 to store block *b*
	8	spare	TF 0 φ	24	Tape channel 0 forward
	9	spare	TF 10 φ	25	Tape channel 1 forward
	10	spare	TR 0 φ	26	Tape channel 0 reverse
	11	spare	TR 10 φ	27	Tape channel 1 reverse
FG0 φ	12	Find gap channel 0 ie blank tape	TH 0 φ	28	Tape channel 0 halt
FG1 φ	13	Find gap channel 1 tape	TH 10 φ	29	Tape channel 1 halt
	14	spare	ET 0 φ	30	Erase tape channel 0
US *x* / *y* φ	15	Use control x, unit y	ET 10 φ	31	Erase tape channel 1

For the Use command, control equipment = 0 for paper tape reader, unit = 0 or 1
= 1 for paper tape punch, unit = 0 or 1
= 2 for magnetic tape channel 0, unit = 0 to 3

Remembering LEO

Frank Land

Emeritus Professor, Information Systems and Innovation Group, Department of Management, London School of Economics and LEO Computers Society
F.Land@lse.ac.uk

Abstract. It is now more than 60 years since the world's first business use of a computer, the valuation of bakery output, was rolled-out on the LEO I computer at Cadby Hall in London, the headquarters of the food production and catering company J. Lyons and Company. LEO I had been designed and built as a computer to be used for business data processing by a team of engineers recruited by Lyons, with a basic design following the design of the Cambridge University EDSAC. The story of the Lyons initiative has been recorded and explanations of how a company in the food business came to build a computer has been told in books and articles in the last decades (- see Appendix 1 for a comprehensive bibliography of material relating to LEO). This chapter remembers the contribution made by LEO.

Keywords: Business computers, J. Lyons and Company, LEO.

The late David Caminer passionately wanted the world to be reminded of the LEO heritage. He had been instrumental in setting up and managing the LEO programming and systems function (Aris 2000, Ferry 2012). He felt that J. Lyons and the LEO team had played an important role in what is now taken for granted – the use of computers in society for a wide range of activities other than mathematical calculations. And he was concerned because it seemed to him that the part played by LEO had been written out of the histories of computing then being published – the 1980s and later, and that as a result there was only a very limited memory of the LEO legacy. A good example is that the 875 page *Concise Encyclopaedia of Computer Science* (Reilly 2004), intended as a reference book for students, devotes half a sentence to LEO in Appendix III on page 832. The UK's Professor Martin Campbell-Kelly, one of the world's leading computer historians, in his *History of the Software Industry* (Campbell-Kelly, 2003) makes no mention of LEO and little of British software despite the pioneering achievements of LEO and other British companies, concentrating largely on American initiatives. He writes, (page 10), "*... I probably have a better knowledge of the British software industry than of the American. However, I felt unable to incorporate much material on the British software industry because it would have appeared disproportionate, would have appeared chauvinistic*". Such modesty can lead to the making of myths.

It was David Caminer who, supported by a small group of old enthusiastic LEO hands, was determined to ensure that the legacy of LEO would be written into computer history. Independently Peter Bird, the ex-head of Systems Development at

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 22–42, 2012.

J. Lyons & Company Limited, had long thought it was important for the LEO story to be part of computer history, and over a number of years, working very much on his own, had compiled a history of the LEO project, which apart from the historical narrative included much technical and personal detail. To publish the book he even set up his own publishing company, and in 1994 published '*LEO: The First Business Computer'* (Bird 1994).

It was in his later years, retired from ICL and Fujitsu, that David Caminer gathered a small band of LEO pioneers and enthused and bullied them into turning the LEO story into a narrative of computing history. The first step was to write an account of the LEO history leavened by the personal reminiscences of many of those who had played a part in the story, including some who had worked for LEO customers. The book, first published by McGraw-Hill in the UK (Caminer *et al* 1996), was subsequently published in the USA (Caminer *et al* 1998), and in 2000 translated into Chinese for publication in China.

Between them these books provide an authoritative history of the LEO story, which helped to bring LEO back into the story of how computer use evolved and the special part played by J. Lyons and LEO.

The publication of the 1996 and 1998 books led to David Caminer proposing the establishment of the LEO Foundation as a charity devoted to promoting the memory of LEO and encouraging academic studies of LEO. The first task the Foundation set itself was to organize a major Conference in 2001 to celebrate the 50th anniversary – November 2001 – of the first time-critical regular job for J. Lyons & Company Limited. The Conference was held at the prestigious London Guildhall, sponsored by a number of organizations including the City of London, Fujitsu, the Wall Street Journal Europe, the IEE (now IET), KPMG and the London School of Economics. It attracted participants and speakers from around the world. A special edition of the Journal of Strategic Information Systems published the proceedings (Journal of Strategic Information Systems 2003).

It was clear that the story of a British catering and food manufacturing business launching the first business computer and setting up a Computer Manufacturing and Service Company as a subsidiary – LEO Computers Limited – was gaining traction at least in the UK and had become more widely known and intrigued parts of the media. The result was the publication of more books telling the LEO story. Science writer Georgina Ferry wrote an account (Ferry 2003) which received the accolade of being the book of the week for BBC Radio 4 in September 2003 being broadcast twice a day for five days on Radio 4.

In 2001 BBC Radio 4 broadcast four 15 minute programs each program devoted to one pioneering initiative in the history of computers. One episode was entitled 'LEO the Lyons Computer'. The series was written, produced and narrated by Mike Hally. Its success led Hally to publish an extended version of the program in a book which demonstrates that computer developments in the 1940 (and earlier) and 1950s took place in many places and were not confined to the USA and Britain (Hally 2003)

Remarkably Channel 5 television was planning a documentary 'Disappearing Britain' screened in 2006 which included the story of tea drinking in Britain with the well-known actor Wendy Craig as presenter.[1] In researching their story they came

[1] "Disappearing Britain: The British Cuppa" with Wendy Craig (2006).

across J. Lyons of teashop and tea marketing fame, and discovered that Lyons was using its own computer LEO for both teashops replenishment and the weekly planning of tea production and blending. Intrigued they followed up the story and devoted a part of one of the programs to LEO and included an interview with the programmer (Frank Land) who had been responsible for the tea blending application.

To celebrate the 60th anniversary of the coronation of Queen Elizabeth II, BBC I television compiled a five episode program 'The 1952 Show' of notable events in Britain around the time of her coronation, and the editors selected the LEO story for episode 5 screened on 30th March 2012. It featured an interview of one of the original team of engineers responsible for the design and construction of LEO I – Ernest Kaye.

The spate of books and media exposure appeared to ensure that in the public sphere in Britain the LEO story had received recognition and admiration even gaining a Guinness Book of Records Certificate as "*The first ever business computer was LEO I (**L**yons **E**lectronic **O**ffice). It began operations in November 1951 at the Lyons headquarters, London, UK*".

Recognition in the wider world of academia was more limited, though many computer history texts, but as suggested above, not all, noted the Lyons initiative and the role played by LEO in the evolution of business computing. Good examples are the IEEE Journal Annals of the History of Computing in the US and the Computer Journal in the UK both of which have carried a number of papers related to LEO over the years (Anderson 2004, Aris 2000, Caminer 1958, Caminer 1997, Caminer 2003, Delve and Anderson 2001, Gosden 1960, Forbes 1965, Lewis 1963, Lewis 1964, Land 2000, Land 2006). Important papers reviewing the achievements of the LEO team and their importance in the history of computing were contributed by senior American Scholars (Baskerville 2003, Mason 2004)).

In 1999 the Association for Information Systems (AIS), the organization representing the academic discipline of Information Systems world-wide with a membership of thousands, established an annual award and called it the LEO Award.

> "*Named after one of the world's first commercial applications of computing (The Lyons Electronic Office), the purpose of the LEO Award is to recognize truly outstanding individuals in the Information Systems community, both academics and practitioners, who have made exceptional contributions to research in and/or the practice of Information Systems. The LEO Award will be a singular honor to recognize seminal work by the Award winner.*"

It is clear that the story of LEO and its pioneering achievements are now much better known. But is it 'mission accomplished'? David Caminer would have said "*no, it is work-in-progress, and you must try harder*". Even as late as August 2011, Eric Schmidt, CEO of Google (Schmidt 2011), in his MacTaggart lecture, noted "*You* (Britain) – my parenthesis - *invented computers in both concept and practice*" and added "*<u>it's not widely known</u>, but the world's first office computer was built in 1951 by the Lyon's chain of teashops*" (my underlining).

In 2011 the LEO Foundation was merged into the LEO Computers Society. The Society had been established primarily as a way of keeping ex-employees in touch with each other and to run regular reunions. It currently (2012) has a membership of over 300, including most of the surviving pioneers. But over time and with encouragement from LEO Foundation members it began to take an interest in ensuring that the LEO legacy was not lost and to take advantage of the knowledge and experience of its membership to retrieve lost elements of that legacy. To aid that process it has set up a LEO History Committee with the remit to take the various ad hoc and opportunistic efforts already in place and put them on a more coherent footing and ensure that outcomes were properly disseminated and made available to scholars, students and the public at large.

There are a number of ways in which the history of computers can be constructed.

- **Via the dedication and enthusiasm of individuals and groups**, including 'institutions' such as the Computer Conservation Society[2] and the LEO Computer Society[3] who volunteer to sift through relics, collating classifying and editing material ranging from hand-written notes, including coding sheets, documents such as minutes of meetings, to artifacts, including technical diagrams and blue prints as well as circuit boards and in some cases complete computers. Others are more concerned with re-constructing computers long out of use or emulating old software.

  One important strand of the LEO Computer Society's activity is building up a group of volunteers drawn from all branches of LEO's activities, but primarily ex-LEO engineers and ex-LEO programmers and systems analysts, to assess the material as it comes to light and determine what items should be saved and where they should be located to provide the best access for later use.

  Perhaps the most important UK institution for remembering the UK computer heritage is the Computer Conservation Society, established jointly by the British Computer Society and the Science Museum. Apart from the outstanding work inspired by the late Tony Sale of reconstructing pioneering computers such as Colossus, the Ferranti Pegasus, ICT 1301 and now EDSAC, they set up a project, 'Our Computer Heritage', headed by Simon Lavington to provide detailed documentation of all the UK computer ranges, including, of course, the LEO range[4]. Though much material on the LEO range has been supplied a great deal of material, much to our regret, is still missing.
- **By locating publications** – newspaper articles, press releases, academic papers, books, and sadly, obituaries, as pioneers pass away. Much documentary material already exists in libraries, archives and museums. These include collections gifted to the archives by individual pioneers or their descendants. However, the collections are widely dispersed, and only rarely transcribed into digital formats required by future scholars and students. Indexing is at best limited. Providing scholars and others with a systematic account of what is available presents the historian with a problem. For example the extensive collection of LEO documents at the Manchester University Rylands library, though well catalogued, is all in its original form comprising manuscripts and typewritten papers as well as some computer print outs.

---

2 http://www.computerconservationsociety.org/
3 http://www.leo-computers.org.uk/
4 http://www.ourcomputerheritage.org/

The bibliography in Appendix 1 should be regarded as a step in the direction of making known what is available and where it is located with respect to the history of LEO, but it is at best work-in progress as more material comes to light. Readers of this chapter are encouraged to help in adding to the bibliography by letting the author know of additional material. Internet searches engines have greatly facilitated the collection of relevant material. But large amounts of documentary material exists in private hoards, unclassified and not indexed, often only coming to light on the death of its collector. A good example is a cache of 100 or so manuscript notebooks found in a cardboard box and left by the original LEO engineer Ernest Lenaerts covering the years from 1949 and now donated to the LEO Computers Society by Lenaerts' sons. Their value cannot be assessed until they have been transcribed, but they have aroused the interest of a number of museums and archives. As noted above it is only the dedication of volunteers that can help rescue such material for posterity.

- **Collecting material**, mainly artifacts, for exhibition in Museums. Many museums of science and technology arrange displays of computer hardware tracing the evolution of computers from their beginnings to the most recent technology available. But there are gaps in the story, exemplified by the story of Colossus the British pioneering code breaking computer during the Second World War that was destroyed, including all documentation relating to its design and construction, on orders from the Government who felt the need not to compromise the secrecy of its operation. The late Tony Sale, the original curator of the Bletchley Park Museum, who led the team which rebuilt the Colossus, wrote:

  "*After VJ Day, suddenly it was all over. Eight of the ten Colossi were dismantled in Bletchley Park. Two went to Eastcote in North London and then to GCHQ at Cheltenham. These last two were dismantled in about 1960 and in 1960 all the drawings of Colossus were burnt. Of course its very existence was kept secret.*"[5]

  Gaps in the collection of LEO (and other) computers are making it difficult for the museums to provide a complete or fully coherent story. Again, the LEO Computers Society is playing its part in plugging the gap. When items are located they frequently carry no identifying label. The society's engineer members try to identify and explain the function of particular computer circuit boards, computer units and blue prints or help museums with their collection of LEO artifacts. One problem is that relics are scattered across a wide range of museums. LEO artifacts are held at the National Museum of Computing at Bletchley Park, and the London Science Museum will be showing pieces of LEO in its new computer exhibits. Recently the Computer Museum in Silicon Valley acquired some LEO pieces and will be showing them as part of its new facilities. Other pieces, including a nearly complete LEO III, are held in the National Museum of Scotland in Edinburgh, and another LEO III in the Museum of London. The bibliography (appendix I) has made a start in showing the holdings in a number of museums and archives.

---

[5] http://www.codesandciphers.org.uk/lorenz/colossus.htm

- **Noting the content of Conferences** and their proceedings, such as computer history conferences, and special events including ones to celebrate anniversaries of notable past events. A good example is the Google sponsored event at the London Science Museum to celebrate the 60th anniversary of the roll-out of the first LEO application on behalf of J. Lyons & Co limited.[6] The IFIP Working Group 9.7[7] provides information about relevant conferences. Conferences provide opportunities for scholar to learn from each other and exchange ideas.
- **Taking Oral Histories**. Historians know that one of the prime resources for historical knowledge is stored in peoples' memories of past events. Unfortunately it is a wasting asset as individuals including those with vital clues about past happenings pass away. This has led to the oral history movement, the attempt to capture memories and knowledge by interviewing and recording the stories of selected individuals. Historians in general rely on documentary evidence culled from account books, tally rolls, legal documents, minutes of meetings and so on supplemented by contemporary accounts provided by commentators and newspapers dating from the time of the event recorded. Whilst such historical accounts can, as historians know, suffer from deliberate distortions and omissions, oral histories in relying on memory at a distance from the events reported can suffer from false memories. In the UK the Oral History Society provides advice and training for novice oral historians.[8] It also sets the standards for recordings to ensure the long lasting quality of the recordings. The taking of oral histories involves interviewing and recording a subject with the interviewer leading the subject through a loose framework of questions. The interview then has to be transcribed, the transcript agreed with the subject, edited and archived in digital format. Transcription represents a major item of cost.

  The British Library has a major Oral History project (the Life Histories project) recording selected individuals from each sector of British society. Two members of the LEO Computer Society were selected from the technology sector, and their histories are now part of the British Library archive. These two were Mary Coombs who joined LEO in 1953, and who may well have been the world's first female commercial programmer, and the author of this chapter.[9] The late John Pinkerton, LEO's chief engineer and designer of LEO I recorded his own history and that is now stored at the Charles Babbage Institute, University of Minnesota, now part of the IT History Society.

---

[6] http://storify.com/lynetter/leo-60th-anniversary
[7] http://www.comphist.org/
[8] http://www.oralhistory.org.uk/
[9] See http://britishlibrary.typepad.co.uk/oralhistoryofscience/2011/08/new-interviews-online.html#tp and http://cadensa.bl.uk/uhtbin/cgisirsi/x/0/0/5?searchdata1=CKEY7328647&library=ALL and http://cadensa.bl.uk/uhtbin/cgisirsi/?ps=4vatGo3loS/WORKS-FILE/22600167/9

The result of these initiatives led the LEO Computers Society to launch its own Oral History project. This required the need to purchase appropriate recording equipment and subsequently to have the recordings professionally transcribed. The society applied for funding from the Association of Information Technology Trust and received a small grant to fund the project. The project was managed by a small team aided by volunteer interviewers who were given notes of guidance (Appendix 2). A schedule of individuals was drawn up giving priority to the oldest subjects. The team experimented with a variety of recording methods, but always retaining the recording standards established by the Oral History Society. These included using Skype and a smart phone to record interviews where the subject was remote from the interviewer, for example the ex-head of LEO in Australia, Neil Lamming. To date (May 2012) some dozen interviews have been conducted.

In addition to the oral histories members of the LEO Computers Society have written their own reminiscences, ranging from a few paragraphs to near monographs describing a particular application, ground-breaking at that time. These too after editing will form part of the LEO archive, to be ultimately placed in a public archive in order to make them as widely available as possible.

Let me finish by paying tribute to those who made the LEO story possible. First John Simmons, then Comptroller of J. Lyons, who saw his task as business process re-engineering to make Lyons a more effective business and who in 1947 sent Thompson and Standingford to the USA to discover if there had been any new ideas which Lyons could use. There they incidentally saw the first US computers, and recognized that they could be the future for business data processing and on their way home sketched how a computer might be used to complete a business process. They wrote the prescient report recommending Lyons to explore the possibility of Lyons using a computer; a report which Simmons thought was right and put it in front of the Lyons Board of Directors. The Board, to their credit, accepted the notion, and set in train the process which led to the Lyons Electronic Office (LEO). Then a tribute to John Pinkerton, the engineer of genius who brought the idea into reality and David Caminer who invented practical systems engineering and whose ideas led to a long sequence of innovative and ambitious applications.

And who are not mentioned? The many people who worked with those noted above and for those who have passed away the obituaries record at least some of their achievements in the LEO story.

In this chapter I am proud to pay tribute to all of them.

References

Anderson, D., Delve, J.: Pioneers of Payroll on computers: LEO, the Army, the Navy Dockyards and De Havilland. IEEE Annals of the History of Computing 24 (2004)

Aris, J.B.B.: Inventing Systems Engineering. IEEE Annals of the History of Computing 22(3), 4–15 (2000)

Baskerville, R.: The LEO principle: perspectives on 50 years of business computing. Journal of Strategic Information Systems 12(4), 255–263 (2003)

Bird, P.J.: LEO: The first Business Computer. Hasler Publishing Limited, Wokingham (1994)

Caminer, D., Aris, J., Hermon, P., Land, F.: User Driven Innovation: The World's First Business Computer. McGraw-Hill, Maidenhead (1996)

Caminer, D., Aris, J., Hermon, P., Land, F.: LEO: The Incredible Story of the World's First Business Computer. McGraw-Hill, New York (1998)

Caminer, D.T.: ...And How to Avoid Them. The Computer Journal, British Computer Society 1(1), 11–14 (1958)

Caminer, D.T.: LEO and its Applications: the Beginning of Business Computing. The Computer Journal, British Computer Society 40(10), 585–597 (1997)

Caminer, D.T.: Behind The Curtain at LEO. IEEE Annals of the History of Computing 25(2), 3–13 (2003)

Delve, J., Anderson, D.: The Pinkerton Lecture. IEEE Annals of the History of Computing 23(2), 68–73 (2001)

Ferry, G.: A computer called LEO: Lyons Teashops and the World's First office Computer, Fourth Estate, London, hardback and Harper Perennial, London, paperback (2003)

Ferry, G.: David Tresman Caminer (1915-2008). Oxford Dictionary of National Biography (2012), http://www.oxforddnb.com/view/article/99795?&docPos=27

Forbes, J.M.: An Introduction to Compiler Writing. The Computer Journal, British Computer Society 8(2), 98–102 (1965)

Gosden, J.A.: Market Research Applications on LEO. The Computer Journal, British Computer Society 3(3), 142–143 (1960)

Hally, M.: Electronic Brains: stories from the dawn of the Computer age, ch. 5, pp. 103–133. Granta Publications, London (2003)

Journal of Strategic Information Systems 12(4) (December 2003)

Lewis, J.W.: Time Sharing on LEO III. The Computer Journal, British Computer Society 6(1), 24–28 (1963)

Lewis, J.W.: The Management of a Large Commercial Computer Bureau. The Computer Journal, British Computer Society 7(4), 255–261 (1964)

Land, F.: The First Business Computer: A Case Study in User-Driven Innovation. IEEE Annals of the History of Computing 22(3), 16–26 (2000)

Land, F.: LEO II and the Model T Ford. British Computer Society Journal 49(6), 650–656 (2006)

Mason, R.O.: The Legacy of LEO: Lessons learned from an English Tea and Cakes Company: Pioneering efforts in Information Systems. Journal of the Association for Information Systems 5(5), 183–219 (2004)

Riley, E.J.: Concise Encyclopedia of Computer Science, 4th edn. Wiley, New York (2004)

Schmidt, E.: Television and the Internet: Shared Opportunity, MacTaggart Lecture (2011), http://www.guardian.co.uk/media/interactive/2011/aug/26/eric-schmidt-mactaggart-lecture-full-text

Appendix 1: Bibliography

Books

- **'LEO and the Managers'**; J.R.M. Simmons, Macdonald, London, 1962. The paperless office concept of the Lyons Comptroller, whose support was vital to the LEO project.
- **LEO, the First Business Computer**; P. Bird, Hasler Publishing, 1994.
  Peter Bird joined Lyons when, as he says, 'the pioneering years of computing were no more than folk history.' Nonetheless, through his 'talking with old-timers' and delving through the Lyons archives, he has made an important contribution to the LEO story. Of particular value are the appendices which, inter alia, give details of the instruction codes, speeds, capacities and deliveries of the different models.
- **User Driven Innovation: The world's first business computer,** (eds.) D.T. Caminer, J.B.B. Aris, P.M.R. Hermon, F.F. Land, McGraw Hill, Maidenhead, 1996. A first-hand account written by thirteen of the early users who developed the disciplines of systems engineering and put LEO to work on economic, time-dependent business applications, starting in 1951. Included is an edited version of the seminal report of the two Lyons executives who, after a tour of the early computer activity in the United States in 1947, recommended that Lyons acquire a computer of their own. Also included is a Science Museum interview with John Simmons.
- **LEO, the Incredible Story of the World's First Business Computer**, (eds.) D.T. Caminer, J.B.B. Aris, P.M.R. Hermon, F.F. Land, McGraw Hill, New York, 1998. The revised US edition of User-Driven Innovation, a Chinese edition was published in 2000.
- **A Computer Called LEO**; Georgina Ferry, Fourth Estate, London, 2003.
  'LEO and its creators deserve their place in history not because of what it was, but because of what it did, for LEO was the first computer in the world to be harnessed to the task of running a business.' A paperback edition was published in 2005, by Trafalgar Square.
- **An ICL anthology**, edited by Hamish Carmichael; Chapter 6, LEO, pp. 91-94, Laidlaw Hicks Publishers, Surbiton, 1996. Chapter 6 presents an anthology of quotations about LEO, mainly from LEO personnel.
- Electronic Brains: stories from the dawn of the Computer age by Mike Hally, Granta Publications, London, 2003. The book is based on 4 BBC radio programmes produced by Mike Hally. Despite its populist title, it is a very readable and informative account of some early computer ventures in the USA, UK, Soviet Union and Australia. Chapter 5 is an account of the LEO story.
- **Early British Computers: The Story of Vintage Computers and The People Who Built Them**; Simon H. Lavington, (Manchester University Press, 1980) at http://ed-thelen.org/comp-hist/EarlyBritish.html#TOC. Chapter 13, pages 68-77, gives a brief history of LEO and English Electric including a timeline.
- ICL – A Business and Technical History; Campbell-Kelly, M., Clarendon Press, Oxford. (1989). The history of ICL is synonymous with the history of the British computer industry. ICL was formed by a series of mergers in response to the increasing market dominance of the large American corporations, particularly IBM. The struggles between these two giants and the inherent problems and implications of competing with US multi-nationals are examined in detail in Campbell Kelly's wide ranging study. At the time of writing in the late 1980s, the author was given unrestricted access to ICL archives and his lucid account of the company, its set-backs and successes makes for a compelling and informative read. This book, which was Winner of the Wadsworth Prize for Business History (1989), will be of great interest to anyone involved in business or the computing industry.

Articles and Other Papers

- Anderson, D., Delve, J., (2004) Pioneers of Payroll on computers: LEO, the Army, the Navy Dockyards and De Havilland , IEEE Annals of the History of Computing, ISSN: 1058-6180.
- Anon (1949); 'A non-technical description of E.D.S.A.C'. How the Cambridge Electronic Calculator works, J. Lyons & Co., June 1949. http://www.leo-computers.org.uk/images/How%20EDSAC%20Works.pdf
- Anon (1954); **'Electronic Abacus'**, The Economist, pp. 789-791, 13th March.
- Anon (1957); **Political & Economic Planning, 'The LEO Computer: a case study in the use of an electronic computer in routine clerical work',** Three Case Studies in Automation**,** July
- Anon (1960), **'Getting to grips with computers'**, The Times Newspaper, reprinted in The Times Newspaper, August 4th 2000
- Anon (1960); **'Notes on Commissioning of LEO Automatic Office at the Ministry of Pensions'**, The Computer Journal, British Computer Society, Vol. 2, No. 4, pp. 198, January.
- Anon (1965); **Daily Mail Obituary for LEO** I 9th January 1965
  Plus other items about LEO initiatives in Eastern Europe
  http://www.ampneycrucis.f9.co.uk/PARK/LEO.htm
- Aris, J.B.B. (1996); **'Systems Design – Then and Now'**, Resurrection, Journal of the Computer Conservation Society, Summer.
- Aris, J.B.B. (2000)**; 'Inventing Systems Engineering'**, IEEE Annals of the History of Computing, Vol. 22, No. 3, pp. 4-15, July-September.
- Aris, J.B., Land, F.F., Mellor, A., (eds.), (2003), **LEO Conference Report**, Journal of Strategic Information Systems, Volume 12, Issue 4, Pages 253-398, December
- Berry, T., (2002), **Lessons from LEO the Lyons**, Financial Director, January 2002 http://www.financialdirector.co.uk/financial-director/feature/1742501/lessons-leo-lyon
- Bidmead, C., (2011) **David Caminer, creator of the first business computer**, Reg Hardware, Unsung Heroes of Tech, 27th September 2011, http://www.reghardware.com/2011/09/27/heroes_of_tech_david_caminer/
- Bird, P., (1990/91), **LEO the Pride of Lyons**, British Journal of Administrative Management
- Caminer, D.T. (1958); **'….And How to Avoid Them'**, The Computer Journal, British Computer Society, Vol. 1, No 1, pp. 11-14.
- Caminer, D.T. (1997); **'LEO and its Applications: the Beginning of Business Computing'**, The Computer Journal, British Computer Society, Vol. 40, No 10, pp. 585-597.
- Caminer, D.T. (2002); **'LEO and the Computer Revolution'**, 2nd annual Pinkerton Lecture, IEE Computing and Control Engineering Journal, Vol. 13.
- Caminer, D.T. (2003); **'Behind The Curtain at LEO'**, IEEE Annals of the History of Computing, Vol. 25, No. 2, pp. 3-13, April-June.
- Cane, A., (2011); 'Perspectives: Unlearnt lessons of LEO', Capgemini, http://www.ft.com/cms/s/0/7c7fc396-168d-11e1-be1d-00144feabdc0.html#axzz1sNr3Iehd
- Coombs, M., (2003) Review: **'A Computer called LEO'** (Ferry, G.) European Journal of Information Systems Vol. 12, Issue 4, 241–24
- Delve, J., Anderson, D., (2001) **The Pinkerton Lecture**, IEEE Annals of the History of Computing, Vol. 23, Issue2, pp. 68-73
- Ferry, G., (2005) **Simmons, John Richardson Mainwaring (1902–1985)**, Oxford Dictionary of National Biography, http://www.oxforddnb.com/index/57/101057059/
- Ferry, G., (2012) **David Tresman Caminer (1915-2008),** Oxford Dictionary of National Biography, http://www.oxforddnb.com/view/article/99795?&docPos=27

- Forbes, J.M., (1965); **'An Introduction to Compiler Writing'**, The Computer Journal, British Computer Society, Vol. 8, No 2, pp. 98-102.
- Gosden, J.A., (1960); **'Market Research Applications on LEO',** The Computer Journal, British Computer Society, Vol. 3, No 3, pp. 142-143.
- Gosden, J.A., (1964), **"The Operations Control Center Multi-Computer Operating System."** Proceedings of the Association for Computing Machinery, E2.2-1-E2.2-9. New York: Association for Computing Machinery.
- Gosden, J.A., (1997); **'Mathematics and Software at LEO Computers'**, Resurrection, Journal of the Computer Conservation Society, No. 17, pp. 15-22, Spring.
- Hendry, J. (1986**); 'The Teashop Computer Manufacturer'**, Business History, Vol. 29, No 8.
- Hinds, M.K. (1981); **'Computer Story'**, Meteorological Magazine, Vol. 110, pp 69-80.
- Kaye, E.J. and Gibbs G.R. (1954); **'LEO – A Checking Device for Punched Paper Tape'**, Electronic Engineering, Vol. 29, pp. 386-392.
- Land, F.F., (1960**), 'Computers in Purchasing and Stores Departments: LEO at the Ford Motor Company Spares Depot'**, Computers in Purchasing and Stores Departments, Purchasing Officers Association, pp 27 – 33.
- Land, F.F. (1996); **'Systems Analysis for Business Applications'**, Resurrection, Journal of the Computer Conservation Society, Summer.
- Land, F.F., (1997**); 'Information Technology Implementation: The Case of the World's First Business Computer: The Initiation Phase'**, in (eds.) McMaster, T., Mumford, E., Swanson, E.B., Warboys, B., and Wastell. D., *Facilitating Technology Transfer through Partnership: Learning from Practice and Research*, p. 3-19, Chapman & Hall, London.
- Land, F.F. (1998); **'LEO, The First Business Computer: A Personal Experience'**, in Glass, R.L. (ed.*), In the Beginning. Personal Recollections of Software Pioneers*, IEEE Computer Society, Los Alamitos, CA.
- Land, F.F. (1999), **A Historical Analysis of Implementing IS at J. Lyons**, in Currie, W.L. and Galliers, R.D., (Eds), Rethinking Management Information Systems, Oxford University Press, Oxford, pp 310 - 325.
- Land, F.F. (2000); **'The First Business Computer: A Case Study in User-Driven Innovation',** IEEE Annals of the History of Computing, Vol. 22, No. 3, pp. 16-26.
- Land, F.F., (2006), **LEO II and the Model T Ford**, British Computer Society Journal, Vol 49, No. 6, pp 650 – 656.
- Lavington, S.H., (1980), **'LEO and English Electric'** in *Early British Computers: The story of vintage Computers and the people who built them,* Chapter 13, pp 68 – 77, Butterworth-Heinemann, London
- Lenaerts, E.H., (1951), **Visual Presentation of Binary Numbers,** Electronic Engineering, Vol. 23. pp. 140-141
- Lenaerts, E.H., (1952?), **LEO, Lyons Electronic Office -- Operation and Maintenance**, Electronic Engineering, Vol. 26, pp. 11-17.
- Lenaerts, E.H. (1954); **'LEO - Operations and Maintenance'**, Electronic Engineering, Vol. 29, pp. 335-341.
- Lewis, J.W. (1963); **'Time Sharing on LEO III'**, The Computer Journal, British Computer Society, Vol. 6, No 1, pp. 24-28.
- Lewis, J.W. (1964); **'The Management of a Large Commercial Computer Bureau'**, The Computer Journal, British Computer Society, Vol. 7, No 4, pp. 255-261.
- Mason, R.O., (2004**), 'The Legacy of LEO***: Lessons learned from an English Tea and Cakes Company: Pioneering efforts in Information Systems'*, Journal of the Association for Information Systems, Vol. 5, No. 5, May, pp 183 – 219.
- Pelling, N., (2002), **The Case For the First Business Computer**, http://www.nickpelling.com/Leo1.html. The business cases behind the five proposals made

to the board of J. Lyons & Co. by Thompson and Standingford in 1947 - which led to the construction of the first business computer - are analysed, but found to be strategically lacking. Both an alternate reading of the case and some contemporary implications are then developed.

- Pinkerton, J.M.M., (1951), **Automatic frequency Control**, Electronic Engineering, Vol. 23. pp. 147-148.
- Pinkerton, J.M.M. (1954); '**The LEO System'**, Electronic Engineering.
- Pinkerton, J.M.M. and Kaye, E.J. (1954); **'LEO – Lyons Electronic Office'**, Electronic Engineering, Vol. 29, pp. 284-291.
- Pinkerton, J.M.M. (1961); **'The Evolution of Design in a Series of Computers'**, The Computer Journal, British Computer Society, Vol. 4, No 1, pp. 42-46.
- Pinkerton, J.M.M. (1966); **'Large-Scale Computing in the Seventies'**, The Computer Journal, British Computer Society, Vol. 10, No. 2, September.
- Pinkerton, J.M.M. (1975); **'Performance Problems with LEO I'**, The Radio and Electronic Engineer, Vol. 45, No. 8, pp. 411-414, August.
- Pinkerton, J.M.M. (1983); **Tape 6 in Christopher Evans's 'Pioneers of Computing'**, IEEE Annals of the History of Computing, Vol. 5, No 1, pp 64-72, January-March.
- Pinkerton, J.M.M. (1987/88); **'The Early History of LEO: The First Data Processing Computer',** The Computer Museum Report, Vol. 21, Winter, ed-thelen.org/comp-hist/TheCompMusRep/TCMR.html
- Pinkerton, J.M.M, (1991), **Taming LEO – Overcoming the Inherent Unreliability of LEO I,** IEE Review, Vol. 37, pp. 13-17.
- Pinkerton, J.M.M., Hemy, D., Lenaerts, E.H., (1992**), The Influence of the Cambridge Mathematical Laboratory on the LEO Project**, IEEE Annals of the History of Computing, Vol. 14, No. 4, pp. 41-48
- Randall, G.A., (1959), **The Potentialities of a Computer in the Smaller Office,** *The Accountant,* September 19th 1959
- Richardson, D.J., (1974), **LEO Computers**, a draft account based on interviews with Anthony Salmon, John Simmons and John Stevens, NAHC/LEO, A7 see http://archives.li.man.ac.uk/ead/html/gb133nahc-leo-p1.shtml
- Smith, D;, (2011), **Innovation in a tea shop, whatever next?,** Fujitsu UK & Ireland CTO Blog, http://blogs.ts.fujitsu.com/uk-ie/cto/2011/11/innovation-in-a-tea-shop-whatever-next
- Thompson, T.R. (1958); **'Four Years of Automatic Office Work'**, The Computer Journal, British Computer Society, Vol. 1, No 1, pp. 106-112.
- Thompson, T.R. (1960); **'Problems of Auditing Computing Data: Internal Audit Practice and External Audit Theory'**, The Computer Journal, British Computer Society, Vol. 3, No 1, pp. 10-11.
- Thompson, T.R. (1962); **'Fundamental Principles of Expressing a Procedure for a Computer Application'**, The Computer Journal, British Computer Society, Vol. 5, No3, pp. 164-169.
- Thompson, T.R., **'The LEO Chronicle, Major Events from 1947 to 1962',** Leo Archive, National Archive for the History of Computing, Manchester, **LEO Computers,** http://www.chstm.man.ac.uk/nahc/contents/leo.htm. As part of the library's special collections, the Archive is located in the main building of John Rylands University Library of Manchester, Burlington Street (building 18 in the campus map).
- Wagner, F. and Wolstenholme, P. (2003**); 'A Modern Real-Time Design Tool; Applying Lessons from LEO'**, IEE Computing and Control Engineering Journal, Vol 14.
- Whitley, P., (2005), **Computerised payroll's Golden anniversary**, Sage Accounting Web http://www.accountingweb.co.uk/topic/tax/computerised-payrolls-golden-anniversary
- Wilkes, M.V. (2001); **'John Pinkerton and Lyons Electronic Office'**, IEE Computing and Control Engineering Journal, Vol. 12, pp. 130-144.

- Journal of Strategic Information Systems, Vol. 12, No. 4. (December 2003), pp. 253-254. The Issue is devoted to the 2001 conference at the London Guildhall celebrating the 50th anniversary of the rolling out of the world's first business application at the Cadby Hall headquarters of J' Lyons and Company on their LEO I computer. URL: http://www.citeulike.org/journal/els-09638687. The contents of the issue are as follows:
  Aris, J.B.B., Preface, pp 253-254
  Baskerville, R. The LEO principle: perspectives on 50 years of business computing, pp 255-263
  Caminer, D.T., LEO and the Computer Revolution, pp 265-284
  Cox, G., Business computing 2001-the state of the art, pp 285-294
  Mowery, D. 50 Years of business computing: LEO to Linux, pp 295-308
  Yapp, C. Conference sponsors' panel: what have we learnt in 50 years? pp 309-320
  Shirley, S. Panel: social and economic consequences of business computing and public policy, pp 321-330
  Ashworth, J.M., Knowledge and digital information, pp 331-337
  Hudson, R. Panel: crystal ball, 2001-2051, pp 339-353
  Ein-Dor, P. The world and business computing in 2051: from LEO to RUR? pp 357-371
  Amaravadi, C.S. The world and business computing in 2051, pp 373-386
  Schell, E.H. The world and business computing at mid-century, pp 387-395
- British Journal of Administrative Management (1991), 75th Anniversary Issue: http://archives.li.man.ac.uk/ead/html/gb133nahc-leo-p1.shtml. The contents include:
  Simmons, the visionary, April 1991
  O&M and Simmons, November 1991

Archives – LEO Documents and Artefacts

- **National Archive for the History of Computing**, Manchester, **LEO Computers,** https://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=history-of-computing-uk;ca4a4c3.0012 provides a listing of documents held by the library.
  http://www.chstm.man.ac.uk/nahc/contents/leo.htm
  http://archives.li.man.ac.uk/ead/html/gb133nahc-leo-p1.shtml
  As part of the library's special collections, the Archive is located in the main building of John Rylands University Library of Manchester, http://rylibweb.man.ac.uk/ Burlington Street (building 18 in the campus map http://www.man.ac.uk/about/campus.html).
- **The Computer Conservation Society** has collected technical and other background Information about all early UK computers including LEO.
  See http://www.ourcomputerheritage.org/CCSprop6.pdf
  http://www.ourcomputerheritage.org/ and
  http://www.ourcomputerheritage.org/acknowledgements.pdf
  The LEO record is incomplete and further technical information would be welcomed by the CCS. The CCS also maintains an index of documents relating to LEO including a complete listing of the **Pinkerton** papers held at the Science Museum http://sw.ccs.bcs.org /iclarch/arch01.html and 65 technical drawings including two patent applications http://sw.ccs.bcs.org/iclarch/arch06.html
- **The Computer History Society** has established an archive search facility of computing history websites. The link to the search facility is: http://ithistory.org/archiveit/archiveit-search.php.

Searching LEO+computers brings up http://ithistory.org/archiveit/search2-realtest-navtest.php?q=LEO%2BComputers&i=1827&p=0&submit=Search with many LEO entries.

- **University of Warwick, Modern Records Centre,** Contact Details: Location **John Simmons Papers (1924-1994)** Ref: GB 152 GB 152 MSS.363, http://archiveshub.ac.uk/data/gb152363
- **Frank Skinner's website**: http://www.ampneycrucis.f9.co.uk/PARK/LEO.htm
- **Andrew Wylie** – Mister Transistor – collects and records information about early transistor computers, including LEO III – see http://homepages.nildram.co.uk/~wylie/trancomp/LEO3.htm
- **Bill Forfar:** Reminiscences at www.groveblue.co.uk/wf.pdf
- **London Metropolitan Archive** holds material relating to J. Lyons and Company, archived under: GB 0074 ACC/3527
- **Museums** A listing of which UK museums hold LEO Artefacts can be found at https://docs.google.com/viewer?a=v&q=cache:PDji3gRefxAJ:www.ourcomputerheritage.org/where%2520to%2520see%2520bits%2520rev.pdf+National+Museum+of+Computing,+Bletchley+LEO&hl=en&gl=uk&pid=bl&srcid=ADGEEShQHZUJK9fLlszQDnyUPVCJHyggZYpZb17rxGSn0sTpHzT-N67c2U8mKwcbndQJCYwUFqjwsZiadPNLwnm9aP2qsDnH8Dyp-JNgYByD7TCatAti0ny37VO2xsBmiE97dQEWtB1N&sig=AHIEtbRt65LTg4sDLd2BvqImwO9oAxlUNA
- **Computer History Museum**, **Silicon Valley**. The Museum holds and displays a number of LEO II relics and various LEO books and documents. Its record of holdings with photos is well organised and can be found on: http://www.computerhistory.org/search/?q=LEO+Computers&site=chm_collection&client=chm_collection&output=xml_no_dtd&submit.x=2&submit.y=2
- **Science Museum, London.** The Museum has a limited collection of LEO relics. A recent grant will enable it to greatly improve its display of computer history items including LEO relics and the LEO story. The curator responsible for the acquisition and display is Dr Tilly Blyth, tilly.blyth@scciencemuseum.org.uk. See also http://objectwiki.sciencemuseum.org.uk/wiki/Leo_II.html
- **The National Museum of Computer History, Bletchley.** Information about LEO with photos on http://www.tnmoc.org/40/section.aspx/24 . No LEO items are listed on the schedule of holdings at the Museums website http://www.tnmoc.org/ . Trustee of Museum and Secretary of Computer Conservation Society is Kevin Murrell, kevin.murrell@tnmoc.org
- **National Museums Scotland, Edinburgh**. The Museum has LEO III/32 (Colvilles Ltd, Motherwell, as well as three LEO III circuit boards, one of which is on display, and three circuit boards from LEO IIs. The Senior Curator of Modern Science and Computing is Dr Tacye Phillipson, : t.phillipson@nms.ac.uk Website: http://www.nms.ac.uk/
- **Museum of London, London.** The Museum has parts of LEO III/45 ((Webb Durlacher Mordant) including the console on display http://www.museumoflondon.org.uk/Collections-Research/Collections-online/object.aspx?objectID=object-49407&start=1&rows=1 Website: http://www.museumoflondon.org.uk/ See also activity sheet for school children: www.museumoflondon.org.uk/.../KS3_Communication_and_technology_world_city_activity_sheets.doc

Private Holdings

Many individuals dead and alive, members of the LEO Computers Society or ex LEO employees have private collections of LEO artefacts, documents and photographs, some of which are of historical importance. They include:

David Caminer: private archive, collected by Ray Hennessey

John Aris: private archive, collected by Ray Hennessey

Colin Tully: private archive

Frank Land: private archive

Ernest Lenaerts: 100 notebooks, quarto, compiled in manuscript, dating from 1949 to the early 1950s. The notebooks have been donated to the LEO Computers Society by Paul and David Lenaerts, Ernest's sons, and are being transcribed into digital form by members of the Computer Conservation Society's team rebuilding EDSAC.

There are many other private hoards and the LEO Computers Society would welcome information about such holdings.

Obituaries and Biographies

Ernest Joseph Kaye 1922-2012
Ben Rooney in Wall Street Journal http://blogs.wsj.com/tech-europe/2012/05/07/u-k-computer-pioneer-dies/
http://www.youtube.com/watch?v=GE6TX70A3Rc
http://blogs.wsj.com/tech-europe/2011/11/14/worlds-first-business-computer-celebrates-60th-anniversary/
https://plus.google.com/106615548454096392167/posts/4v94i6yGPst#106615548454096392167/posts/4v94i6yGPst
http://www.bbc.co.uk/blogs/outriders/2011/11/leo_making_history.shtml

George Manley 1938-2011
http://centaurs-rfc.org/html/web_news_29.html

Gordon Foulger 1942-2011
http://www.gordonfoulger.co.uk/obituary.php

John Aris 1934-2010
http://www.guardian.co.uk/technology/2010/aug/26/john-aris-obituary
http://www.vukutu.com/blog/2010/08/a-computer-pioneer/
http://www.cs.man.ac.uk/CCS/res/res52.htm#i

Mavis Hinds 1929-2009
Mavis Hinds working for the Meteorological Office used LEO I for weather forecasting – the earliest use of computers for modelling the weather in the early 1950s.
http://onlinelibrary.wiley.com/doi/10.1002/wea.502/abstract

David Caminer 1913-2008
http://www.telegraph.co.uk/news/obituaries/2188963/David-Caminer.html
http://en.wikipedia.org/wiki/David_Caminer
http://uk.fujitsu.com/pensioner/topics/obituaries/david_caminer/
http://www.nytimes.com/2008/06/29/technology/29caminer.html
http://www.liverpooldailypost.co.uk/views/obituaries/2008/06/26/david-caminer-64375-21152334/
http://boingboing.net/2008/06/29/computer-pioneer-and.html
http://www.guardian.co.uk/technology/2008/jul/11/1

Colin Tully 1936-2007
http://www.bcs.org/content/conWebDoc/16757
http://comjnl.oxfordjournals.org/content/52/3/388.short
http://www.leo-computers.org.uk/images/colintullytribute.pdf

John Gosden 1930-2003
http://www.leo-computers.org.uk/gosdenobit.html
http://www.cs.manchester.ac.uk/other/CCS/res/res33.htm#f
\\staff2\l_users\LANDF\Leo\Gosden obit in Independent.htm
http://ithistory.org/memoriam/memoriam-detail.php?recordID=80

Leo Fantl 1924-2000
http://www.thocp.net/biographies/fantl_leo.html
http://www.smartcomputing.com/editorial/dictionary/detail.asp?guid=&searchtype=1&DicID=18016&RefType=Encyclopedia

Anthony Salmon 1916-2000
http://www.kzwp.com/lyons.pensioners/obituary2S.htm (page 1)

John Pinkerton 1919-1997
http://www.independent.co.uk/news/obituaries/obituary-john-pinkerton-1144708.html?pageToolsFontSize=200%25
http://www.cs.man.ac.uk/CCS/res/res19.htm#g
http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=707576

Ernest Lenearts 1910-1997
http://www.cs.man.ac.uk/CCS/res/res17.htm#f

Dan Broido 1903-1990
http://www.kzwp.com/lyons.pensioners/obituary2B.htm (page 1)

John Simmons 1902-1985
http://www.oldbrightonians.com/notable-obs/business/john-simmons-bc.-1916.html
http://www.oxforddnb.com/view/printable/57059

Oliver Standingford 1912-1980
http://www.kzwp.com/lyons.pensioners/obituary2S2.htm

Thomas Raymond Thompson – TRT 1907-1976
http://www.kzwp.com/lyons.pensioners/obituary2T.htm
http://www.oxforddnb.com/view/article/101160

Brief biographical sketches of a number of Lyons and LEO people can be found in **LEO, the First Business Computer**; P. Bird, Hasler Publishing, 1994, pages 200-212. The following people – in alphabetical order - are noted: John Barnes, Daniel Broido, David Caminer, Mary Coombs, Leo Fantl, Isidore Gluckstein, Montague Gluckstein, Samuel Gluckstein, John Gosden, John Grover, Derek Hemy, Ernest Kaye, Frank Land, Ernest Lenaerts, Joseph Lyons, John Pinkerton, Anthony Salmon, Ray Shaw, John Simmons, Oliver Standingford, Thomas Raymond Thompson, David Wheeler, Maurice Wilkes, Peter Wood.

Oral and Narrative Histories

- **Title:** Oral history interview with John M. M. Pinkerton **Call Number:** OH 149 **Interviewee:** Pinkerton, John M. M., (John Maurice McLean), 1919- **Repository:** Charles Babbage Institute, University of Minnesota, Minneapolis **Description:** Transcript, 54 pp.

  URL: http://special.lib.umn.edu/cbi/oh/display.phtml?id=121

Abstract: Pinkerton begins by discussing his education and wartime work in radar technology in England. He then describes his movement into the computer industry after World War II and his work on the LEO I and LEO II computers. In this context he discusses the British computer firms J. Lyons and Company, Leo Computers, English Electric Co., and International Computers Ltd.

- **Title:** Life Stories, British Library, An Oral History of British Science

Interviewee: Mary Coombs (nee Blood), LEO Programmer 1953
Interviewer: Thomas Lean
Link: http://cadensa.bl.uk/uhtbin/cgisirsi/Af8OzibrN6/WORKS-FILE/245260056/9

Abstract: This is a full oral history of the life of Mary Coombs as part of the British Libraries Oral History series on the life of selected British Computer scientists.

- **Title:** Life Stories, British Library, An Oral History of British Science

Interviewee: Frank Land
Interviewer: Thomas Lean
Link: http://cadensa.bl.uk/uhtbin/cgisirsi/Af8OzibrN6/WORKS-FILE/245260056/9

Abstract: This is a full oral history of the life of Frank Land as part of the British Libraries Oral History series on the life of selected British Computer scientists

- **Title:** LEO Computer Society Oral Histories

Interviewee: Alan King
Interviewer: Tony Morgan
Link: https://exchange.lse.ac.uk/exchange/LANDF/Inbox/Re:%20Oral%20Histories-9.EML/1_multipart_xF8FF_4_LEO001.html/C58EA28C-18C0-4a97-9AF2-036E93DDAFB3/LEO001.html?attach=1

- **Title:** LEO Computer Society Oral Histories

Interviewee: Tony Morgan
Interviewer: Ray Hennessey

- **Title:** LEO Computer Society Oral Histories

Interviewee: Ralph Land
Interviewer: Martin Garthwaite
Link: https://www.dropbox.com/home/Leo%20Interviews#:::76715808

- **Title:** LEO Computer Society Oral Histories

Interviewee: Bob Gibson
Interviewer: Martin Garthwaite
Link: https://www.dropbox.com/home/Leo%20Interviews#:::76715808

- **Title:** LEO Computer Society Oral Histories

Interviewee: Neill Lamming
Interviewer: Martin Garthwaite
Link: https://www.dropbox.com/home/Leo%20Interviews#:::76715808

- **Title:** LEO Computer Society Oral Histories

Interviewee: Roger Coleman **Interviewer:** Tim-Greening Jackson **Link**: tim@greening-jackson.com

- **Title:** LEO Computer Society Oral Histories

Interviewee: Doug Comish **Interviewer:** Martin Garthwaite
Link: https://www.dropbox.com/home/Leo%20Interviews#:::76715808

- **Title:** Computer History Museum, Silicon Valley, Oral Histories

Interviewee: Chris Date
Interviewer: Thomas Haigh
Link: http://www.computerhistory.org/collections/accession/102658166

Chris Date, **well** known for his work on Data Base theory and practice started his career with LEO, and provides a description of his experience on pages 7 and 8 of the transcript of the interview.

Media Coverage

- **BBC 1 TV** in March 2012 ran a 5 episode series fronted by Len Goodman, reminiscing about the 1959s entitled ***The 1952 Show*** in honour of the Queen's Jubilee. Episode 5 screened on March 30th at 9.15am. It featured the LEO story with an excellent interview of Ernest Kaye. See also http://www.bbc.co.uk/programmes /b01f9qw3/broadcasts/2012/03 The section featuring LEO and Ernest Kaye can be seen on YouTube at http://www.youtube.com/watch?v=GE6TX70A3Rc
- **60th Anniversary event at Science Museum 11th November 2011**
  The event received wide coverage including interviews on the BBC Today programme, BBC World Service, and BBC5 Live Outriders programme. It was also covered by the Daily Telegraph and the Daily Mail website The links below include reports in media, video recordings, radio recordings, photographs:
  http://storify.com/lynetter/leo-60th-anniversary
  http://www.pcauthority.com.au/Tools/Print.aspx?CIID=280510
- **Eric Schmidt in the 2011 MacTaggart Lecture** noted:
  "You (UK) invented computers in both concept and practice. (It is not widely known, but the world's first office computer was built in 1951 by Lyons' chain of teashops!). Yet today none of the world's leading exponents in these fields are from the UK"
  http://s3.documentcloud.org/documents/238974/mactaggart-lecture-2011.pdf
  - The Guardian on 2011 McTaggart Lecture Bridging the arts and sciences divide
  http://www.guardian.co.uk/technology/2011/aug/29/bridging-arts-science-divide
  - **TV Channel 5** ran a series of programmes on **D*isappearing Britain*.** The third episode in December 2006, entitled The BRITISH CUPPA WITH WENDY CRAIG, and included a section on J. Lyons with its Corner Houses and Teashops. This included the story of LEO including an interview with Frank Land. See also http://ftvdb.bfi.org.uk /sift/title/824846?view=synopsis and watch on http://www.ovguide.com/tv_episode/disappearing-britain-season-1-episode-3-the-british-cuppa-with-wendy-craig-536421
  - **BBC Radio 4 commissioned** a programme from Pennine Productions called **Electronic Brains** which was broadcast on 30 October 2001. The programme was compiled by and fronted by Mike Hally and one of the four episodes featured the story of LEO. See http://www.bbc.co.uk/radio4/science/electronicbrains.shtml

Films

- *LEO The Automatic Office,* 1957 promotional film highlighting the way LEOs were constructed and their many varied business uses, ranging from teashop inventories to Ford's payroll. Copyright LEO Computer Society. http://www.youtube.com/watch?v=-8K-xbx7jBM Also held by Computer History Museum Silicon Valley http://www.computerhistory.org/collections/accession/102705993

- *Taking the Punch Out of Input,* 1970s era film by Lyons Computer Services focusing on input devices for LEO III range developed in 1960s. http://www.youtube.com/watch?v=6IPVdHHRc2Q
- *Electronic Data Processing. A series of film strips produced by the Institute of Office Management by its EDP Committee covering LEO I, Elliott 405 and Ferranti Pegasus. Available from Kevin Murrell at the National Museum of Computer History, Bletchley.*

Appendix 2 – LEO Oral History Guidelines

1. Introduction

The LEO Oral Histories are an attempt to capture the knowledge and memories of employees of LEO Computers and its customers for posterity. It is the intention to keep the voice recordings and the digitised transcripts in an archive, possibly at The National Museum of Computing (TNMOC) at Bletchley Park and/or elsewhere. Links to the archive will be provided by a range of museum and specialist websites, including that of the LEO Computers Society, to permit as free access to the material as possible.

The archive will form part of the historical record of the development and evolution of computing in the UK. Its users are expected to be international scholars studying the history of computers and computing, the development of business computing and administrative computing, the impact of computing on society, innovation and innovation diffusion.

Other users will be undergraduate and graduate students in computing and business disciplines engaged in projects related to the history of their discipline, as well as journalists and writers preparing scripts and texts for articles, books, radio and television programs, and films which have computer related themes as well as biographical material involving some of the subjects of the oral histories.

Oral Histories require an interviewer using special audio recording equipment supplied by the Society to record the subject's spoken history. The guidelines below provide a framework for an interviewer to work to. Once an interview has been recorded it has to be transcribed into printed text and the printed text checked by the subject and corrected. Corrections and explanations would only be added to the transcript in the form of indexed Endnotes. A standard comment about the corrections and explanations should be recorded by the Interviewer at the end of the oral recording.

The copyright for the text is vested in the LEO Computers Society with the full agreement of the subject. The standard copyright claim and year of claim will be recorded at the end of the oral recording by the Interviewer as follows:

> *"This interview of <interviewee> has been recorded by The LEO Computers Society as part of an Oral History Project to document the earliest use of electronic computers in business applications. Any opinions expressed are those of the interviewee and not of the Society. Copyright of this interview in recorded form and in transcript remains with The LEO Computers Society, 201x."*

Interviewers are volunteers primarily drawn from the Society membership though other volunteers, such as students studying or researching computer history, may with the approval of the Society carry out appropriate interviews

The LEO Oral History project is supported by funding from the Association for Information Technology Trust (AITT).

2. Guidelines for Interviewers

Because the recordings will be used by a great variety of researchers not all of whom will be familiar with mid-20th century technology or the role played by Lyons, it is essential that interviewers ensure that where a respondent uses technical terms or assumes familiarity with, for example, the names of people or historical events, he or she is asked to explain and clarify the points at issue. A good example would be a respondent noting Kimball Tags or mentioning TRT by the use of those very familiar initials. The recording should explain what Kimball Tags are and how they were used, and spell out TRT's proper name and briefly who he was.

It is essential that interviewers master the recording equipment before embarking on an interview. It is very easy to make a mistake such as forgetting to change tapes, or keeping the recording going even when the formal interview has been concluded and the interviewer and subject gossip about some of their experiences.

Interviewing and responding can be very tiring and place a strain on voice and concentration. It may be best to divide the whole interview into a number of sessions each lasting an hour or so. The important thing is that both interviewer and subject feel comfortable, and that will vary with different people. A very full life history may take quite a few sessions and because it is easy to lose concentration and hence the thread, it may be best conducted over more than one day.

It is the responsibility of the interviewer to 'guide' the respondent by asking questions which lead to the unfolding of the narrative. But at the same time the interviewer must not dominate and should intervene only to get the respondent to clarify and explain where that is felt to be necessary. Above all the respondent must feel that it is his or her story which is the subject of the interview. That means that the interviewee should be encouraged to voice opinions and make value judgements where this will enhance the story.

Some remembered items may on reflection be regarded as being potentially damaging to living persons or slander them. Interviewers and respondents need to be alive to the possibility and be prepared to redact such items or place an embargo on their publication.

2.1 Introduction to the Interview

The oral introduction to the interview recording is indicated above. The following is an example, recorded for the interview of Tony Morgan. The text should be on the lines of:

> It's the 8th of November 2011 and I'm Ray Hennessy. I'm interviewing Tony Morgan to give us the story of his involvement with LEO Computers from the earliest days.
>
> Good morning, Tony. We are recording this interview as a part of The LEO Computers Society Oral History Project. The audio version and the transcript will be lodged at a central archive and made available for researchers and members of the public.
>
> Perhaps you'd like to introduce yourself …

On the transcript, which starts with the above introduction the following should be inserted:

> This transcript of the voice recording has been edited with the agreement of the respondent, Tony Morgan. Mistakes have been corrected and some clarification added within the text in italics; terms which may not be familiar to readers are explained in end notes.

2.2 Conclusion of the Interview

The interviewer should end the formal interview with something along these lines:

> This interview with Tony Morgan has been recorded by The LEO Computers Society and the Society would like to thank Tony very much for his time and reminiscences. The interview and the transcript form part of an Oral History project to document the early use of electronic computers in business and other applications, but particularly in business. Any opinions expressed are those of the interviewee, that is Tony, and not of the Society. The copyright of this interview in recorded form and in transcript remains the property of The LEO Computers Society, 2011.

The following may be added at the end of the transcript, if appropriate:

> As indicated in the introduction, changes and clarifications have been included in the transcript.

3. Guidelines for the Interviewee

It is a very good idea to make comprehensive notes of your career in respect of LEO Computers but also note how your life led you, eventually, to working with LEO, and what happened afterwards. Note the names of anyone you can remember working with at LEO, not necessarily LEO staff, but any other people who were also devising LEO-based systems.

Suggested CV and Interview Framework: The framework is not intended as pro-forma script. Oral histories will vary in the detail provided, ranging from anecdotes about specific events to full life histories including details of domestic and family life. These guidelines are to be regarded as a menu. As far as possible the interviewee should dictate the flow. The role of the interviewer is to provide prompts when needed and to ensure that the interviewee's narrative is as clear and complete as possible without excessive intrusion.

- **Family**: Born, where and when. Occupation of father, mother. Any early memories?
- **Education**: Schools, further education, university. What subjects interested you, fired your imagination. What subjects did you take? Any special events or incidents having a bearing on what you studied? Did you do National Service? If so what did you do and where did you serve?
- **Finding a Career**: Career advice at School or University. What career did you envisage and why. What were your ambitions? Influence of parents on career. How did your career start? Had you heard of computers before you started you career. If so, how had you heard and what did you know? Had you heard of LEO or Lyons? What did you know about those companies or any other computer companies? Did you have friends or family who had gone into computers?
- **Client Companies**: If you worked for a client company, how did you come to work with LEO. [*Pick out relevant elements from the following paragraphs.*]
- **Joining Lyons or LEO**: When did you join? How did that happen? Who interviewed you? Did the interview make any special impression on you? Did you go direct to LEO or another part of Lyons? If so what did you do and when did you tranfer to LEO? What was your first job when you joined LEO
- **Career with LEO**: Take us through your LEO career. Which computers and your role as the years passed? What do you remember of these years? Impressions of your colleagues and managers? Memorable incidents, including funny ones. Any feelings that you were at the forefront of business or computer innovation? Was the job exciting or boring? Did you work with LEO customers? Bureau or purchasers? Who were they? Review the experience of working with customers. When did you leave LEO? What made you leave?
- **Home life**: Where you married before joining LEO or later or not at all. Children? Home/work balance? How did that affect relationships?
- **Career after LEO**: What did you do in your subsequent career? How did it compare to working with LEO? Did your LEO career influence your subsequent career and if so in what way?
- **Professional activities**: Are you a member of the British Computer Society or other body such as IEE (now IET)? If so where you an active member? How? Did you maintain this in your retirement? Have you read learned papers or articles on computers, attended conferences, or read any of the LEO books?
- **Retirement**: When did you retire? What did you do in your retirement? Did you maintain contact with your ex LEO colleagues? Did you attend any LEO or computer history functions after your retirement?
- **Finally**: Reflect on your life experience with LEO. What remains with you of that experience?

NB The above is laid out as a sequence. In practice the interviewee will recollect events and feelings out of order and may want to add elements recalled during the interview. This has to be permitted with the interviewer providing explanations in the recording as this occurs.

A Possible First Use of CAM/CAD

Norman Sanders

Cambridge Computer Lab Ring, William Gates Building, Cambridge, England
ProjX, Walnut Tree Cottage, Tattingstone Park, Ipswich, Suffolk IP9 2NF, England
norman_sanders@onetel.com

Abstract. This paper is a discussion of the early days of CAM-CAD at the Boeing Company, covering the period approximately 1956 to 1965. This period saw probably the first successful industrial application of ideas that were gaining ground during the very early days of the computing era. Although the primary goal of the CAD activity was to find better ways of building the 727 airplane, this activity led quickly to the more general area of computer graphics, leading eventually to today's picture-dominated use of computers.

Keywords: CAM, CAD, Boeing, 727 airplane, numerical-control.

1 Introduction to Computer-Aided Design and Manufacturing

Some early attempts at CAD and CAM systems occurred in the 1950s and early 1960s. We can trace the beginnings of CAD to the late 1950s when Dr. Patrick J. Hanratty developed Pronto, the first commercial numerical-control (NC) programming system. In 1960, Ivan Sutherland at MITs Lincoln Laboratory created Sketchpad, which demonstrated the basic principles and feasibility of computer-aided technical drawing.

There seems to be no generally agreed date or place where Computer-Aided Design and Manufacturing saw the light of day as a practical tool for making things. However, I know of no earlier candidate for this role than Boeing's 727 aircraft. Certainly the dates given in the current version of Wikipedia are woefully late; ten years or so.

So, this section is a description of what we did at Boeing from about the mid-fifties to the early sixties. It is difficult to specify precisely when this project started – as with most projects. They don't start, but having started they can become very difficult to finish. But at least we can talk in terms of mini eras, approximate points in time when ideas began to circulate and concrete results to emerge.

Probably the first published ideas for describing physical surfaces mathematically was Roy Liming's *Practical Analytic Geometry with Applications to Aircraft*, Macmillan, 1944. His project was the Mustang fighter. However, Liming was sadly way ahead of his time; there weren't as yet any working computers or ancillary equipment to make use of his ideas. Luckily, we had a copy of the book at Boeing, which got us off to a flying start. We also had a mighty project to try our ideas on – and a team of old B-17/29 engineers who by now were running the company, rash enough to allow us to commit to an as yet unused and therefore unproven technology.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 43–56, 2012.

Computer-aided manufacturing (CAM) comprises the use of computer-controlled manufacturing machinery to assist engineers and machinists in manufacturing or prototyping product components, either with or without the assistance of CAD. CAM certainly preceded CAD and played a pivotal role in bringing CAD to fruition by acting as a drafting machine in the very early stages. All early CAM parts were made from the engineering drawing. The origins of CAM were so widespread that it is difficult to know whether any one group was aware of another. However, the NC machinery suppliers, Kearney & Trecker etc, certainly knew their customers and would have catalysed their knowing one another, while the Aero-Space industry traditionally collaborated at the technical level however hard they competed in the selling of airplanes.

2 Computer-Aided Manufacturing (CAM) in the Boeing Aerospace Factory in Seattle *by Ken McKinley*

The world's first two computers, built in Manchester and Cambridge Universities, began to function as early as 1948 and 1949 respectively, and were set to work to carry out numerical computations to support the solution of scientific problems of a mathematical nature. Little thought, if any, was entertained by the designers of these machines to using them for industrial purposes. However, only seven years later the range of applications had already spread out to supporting industry, and by 1953 Boeing was able to order a range of Numerically-Controlled machine tools, requiring computers to transform tool-makers' instructions to machine instructions. This is a little remembered fact of the early history of computers, but it was probably the first break of computer application away from the immediate vicinity of the computer room.

The work of designing the software, the task of converting the drawing of a part to be milled to the languages of the machines, was carried out by a team of about fifteen people from Seattle and Wichita under my leadership. It was called the Boeing Parts-Programming system, the precursor to an evolutionary series of Numerical Control languages, including APT – Automatically Programmed Tooling, designed by Professor Doug Ross of MIT. The astounding historical fact here is that this was among the first ever computer compilers. It followed very closely on the heels of the first version of FORTRAN. Indeed it would be very interesting to find out what, if anything preceded it.

As early as it was in the history of the rise of computer languages, members of the team were already aficionados of two rival contenders for the job, FORTRAN on the IBM 704 in Seattle, and COBOL on the 705 in Wichita. This almost inevitably resulted in the creation of *two* systems (though they appeared identical to the user): Boeing and Waldo, even though ironically neither language was actually used in the implementation. Remember, we were still very early on in the development of computers and no one yet had any monopoly of wisdom in how to do anything.

The actual programming of the Boeing system was carried out in computer machine language rather than either of the higher-level languages, since the latter were aimed at a very different problem area to that of determining the requirements of machine tools.

A part of the training of the implementation team consisted of working with members of the Manufacturing Department, probably one of the first ever interdisciplinary enterprises involving computing. The computer people had to learn the language of the Manufacturing Engineer to describe aluminium parts and the milling machine processes required to produce them. The users of his new language were to be called Parts Programmers (as opposed to computer programmers).

A particularly tough part of the programming effort was to be found in the “post processors”, the detailed instructions output from the computer to the milling machine. To make life interesting there was no standardisation between the available machine tools. Each had a different physical input mechanism; magnetic tape, analog or digital, punched Mylar tape or punched cards. They also had to accommodate differences in the format of each type of data. This required lots of discussion with the machine tool manufacturers - all very typical of a new industry before standards came about.

A memorable sidelight, just to make things even more interesting, was that Boeing had one particular type of machine tool that required analog magnetic tape as input. To produce it the 704 system firstly punched the post processor data into standard cards. These were then sent from the Boeing plant to downtown Seattle for conversion to a magnetic tape, then back to the Boeing Univac 1103A for conversion from magnetic to punched tape, which was in turn sent to Wichita to produce analog magnetic tape. This made the 1103A the world’s largest, most expensive punched tape machine.

As a historical footnote, anyone brought up in the world of PCs and electronic data transmission should be aware of what it was like back in the good old days!

Another sidelight was that detecting and correcting parts programming errors was a serious problem, both in time and material. The earliest solution was to do an initial cut on wood or plastic foam, or on suitable machine tools, to replace the cutter with a pen or diamond scribe to ‘draw’ the part. Thus the first ever use of an NC machine tool as a computer-controlled drafting machine, a technique vital later to the advent of Computer-Aided Design.

Meanwhile the U. S. Air Force recognised that the cost and complication of the diverse solutions provided by their many suppliers of Numerical Control equipment was a serious problem. Because of the Air Force’s association with MIT they were aware of the efforts of Processor Doug Ross to develop a standard NC computer language. Ken McKinley, as the Boeing representative, spent two weeks at the first APT (Automatic Programmed Tooling) meeting at MIT in late 1956, with representatives from many other aircraft-related companies, to agree on the basic concepts of a common system where each company would contribute a programmer to the effort for a year. Boeing committed to support mainly the ‘post processor’ area. Henry Pinter, one of their post-processor experts, was sent to San Diego for a year, where the joint effort was based. As usually happened in those pioneering days it took more like 18 months to complete the project. After that we had to implement APT in our environment at Seattle.

Concurrently with the implementation we had to sell ourselves and the users on the new system. It was a tough sell believe me, as Norm Sanders was to discover later over at the Airplane Division. Our own system was working well after overcoming the many challenges of this new technology, which we called NC. The users of our system were not anxious to change to an unknown new language that was more complex. But upper management recognized the need to change, not least because of an important factor, the imminence of another neophytic technology called *Master Dimensions*.

3 Computer-Aided Design (CAD) in the Boeing Airplane Division in Renton, *by Norman Sanders*

The year was 1959. I had just joined Boeing in Renton, Washington, at a time when engineering design drawings the world over were made by hand, and had been since the beginning of time; the definition of every motorcar, aircraft, ship and mousetrap consisted of lines drawn on paper, often accompanied by mathematical calculations where necessary and possible. What is more, all animated cartoons were drawn by hand. At that time, it would have been unbelievable that what was going on in the aircraft industry would have had any effect on The Walt Disney Company or the emergence of the computer games industry. Nevertheless, it did. Hence, this is a strange fact of history that needs a bit of telling.

I was very fortunate to find myself working at Boeing during the years following the successful introduction of its 707 aircraft into the world's airlines. It exactly coincided with the explosive spread of large computers into the industrial world. A desperate need existed for computer power and a computer manufacturer with the capacity to satisfy that need. The first two computers actually to work started productive life in 1948 and 1949; these were at the universities of Manchester and Cambridge in England. The Boeing 707 started flying five years after that, and by 1958, it was in airline service. The stage was set for the global cheap travel revolution. This took everybody by surprise, not least Boeing. However, it was not long before the company needed a shorter-takeoff airplane, namely the 727, a replacement for the Douglas DC-3. In time, Boeing developed a smaller 737, and a large capacity airplane – the 747. All this meant vast amounts of computing and as the engineers got more accustomed to using the computer there was no end to their appetite.

And it should perhaps be added that computers in those days bore little superficial similarity to today's computers; there were certainly no screens or keyboards! Though the actual computing went at electronic speeds, the input-output was mechanical - punched cards, magnetic tape and printed paper. In the 1950s, the computer processor consisted of vacuum tubes, the memory of ferrite core, while the large-scale data storage consisted of magnetic tape drives. We had a great day if the computer system didn't fail during a 24 hour run; the electrical and electronic components were very fragile.

We would spend an entire day preparing for a night run on the computer. The run would take a few minutes and we would spend the next day wading through reams of paper printout in search of something, sometimes searching for clues to the mistakes we had made. We produced masses of paper. You would not dare *not* print for fear of letting a vital number escape. An early solution to this was faster printers. About 1960 Boeing provided me with an ANalex printer. It could print one thousand lines a minute! Very soon, of course, we had a row of ANalex printers, wall to wall, as Boeing never bought one of anything. The timber needed to feed our computer printers was incalculable.

4 The Emergence of Computer Plots

With that amount of printing going on it occurred to me to ask the consumers of printout what they did with it all. One of the most frequent answers was that they plotted it. There were cases of engineers spending three months drawing curves resulting from a single night's computer run. A flash of almost heresy then struck my digital mind. Was it possible that we could program a digital computer to draw (continuous) lines? In the computing trenches at Boeing we were not aware of the experimentation occurring at research labs in other places. Luckily at Boeing we were very fortunate at that time to have a Swiss engineer in our computer methods group who could both install hardware and write software for it; he knew hardware and software, both digital and analog. His name was Art Dietrich. I asked Art about it, which was to me the unaskable; to my surprise Art thought it was possible. So off he went in search of a piece of hardware that we could somehow connect to our computer that could draw lines on paper.

Art found two companies that made analog plotters that might be adaptable. One company was Electro Instruments in San Diego and the other was Electronic Associates in Long Branch, New Jersey. After yo-yoing back and forth, we chose the Electronic Associates machine. The machine could draw lines on paper 30x30 inches, at about twenty inches per second. It was fast! But as yet it hadn't been attached to a computer anywhere. Moreover, it was accurate - enough for most purposes. To my knowledge, this was the first time anyone had put a plotter in the computer room and produced output directly in the form of lines. It could have happened elsewhere, though I was certainly not aware of it at the time. There was no software, of course, so I had to write it myself. The first machine ran off cards punched as output from the user programs, and I wrote a series of programs: Plot1, Plot2 etc. Encouraged by the possibility of selling another machine or two around the world, the supplier built a faster one running off magnetic tape, so I had to write a new series of programs: Tplot1, Tplot2, etc, (T for tape). In addition, the supplier bought the software from us - Boeing's first software sale!

While all this was going on we were pioneering something else. We called it *Master Dimensions.* Indeed, we pioneered many computing ideas during the 1960s. At that time Boeing was probably one of the leading users of computing worldwide and it seemed that almost every program we wrote was a brave new adventure. Although North America defined mathematically the major external surfaces of the wartime Mustang P-51 fighter, it could not make use of computers to do the mathematics or to construct it because there were no computers. An account of this truly epochal work appears in Roy Liming's book.

By the time the 727 project was started in 1960, however, we were able to tie the computer to the manufacturing process and actually define the airplane using the computer. We computed the definition of the outer surface of the 727 and stored it inside the computer, making all recourse to the definition via a computer run, as opposed to an engineer looking at drawings using a magnifying glass. This was truly an industrial revolution

Indeed, when I look back on the history of industrial computing as it stood fifty years ago I cringe with fear. It should never have been allowed to happen, but it did. And the reason why it did was because we had the right man, Grant W. Erwin Jr, in the right place, and he was the only man on this planet who could have done it. Grant was a superb leader – as opposed to manager – and he knew his stuff like no other. He knew the mathematics, Numerical Analysis, and where it didn't exist he created new methods. He was loved by his team; they would work all hours and weekends without a quibble whenever he asked them to do so. He was an elegant writer and inspiring teacher. He knew what everyone was doing; he held the plan in his head. If any single person can be regarded as the inventor of CAD it was Grant. Very sadly he died, at the age of 94, just as the ink of this chapter was drying.

When the Master Dimensions group first wrote the programs, all we could do was print numbers and draw plots on 30x30 inch paper with our novel plotter. Mind-blowing as this might have been it did not do the whole job. It did not draw *full scale, highly accurate engineering lines.* Computers could now draw but they could not draw large pictures or accurate ones or so we thought.

5 But CAM to the Rescue!

Now there seems to be a widely-held belief that computer-aided design (CAD) preceded computer-aided manufacturing (CAM). All mention of the topic carries the label CAD-CAM rather than the reverse, as though CAD led CAM. However, this was not the case, as comes out clearly in Ken McKinley's section above. Since both started in the 1956-1960 period, it seems a bit late in the day now to raise an old discussion. However, there may be a few people around still with the interest and the memory to try to get the story right. The following is the Boeing version, at least, as remembered by some long retired participants.

5.1 Numerical Control Systems

The Boeing Aerospace division began to equip its factory about 1956 with NC machinery. There were several suppliers and control systems, among them Kearney & Trecker, Stromberg-Carlson and Thompson Ramo Waldridge (TRW). Boeing used them for the production of somewhat complicated parts in aluminium, the programming being carried out by specially trained programmers. I hasten to say that these were not *computer* programmers; they were highly experienced machinists known as *parts programmers*. Their use of computers was simply to convert an engineering drawing into a series of simple steps required to make the part described.

The language they used was similar in principle to basic computer languages in that it required a problem to be analyzed down to a series of simple steps; however, the similarity stopped right there. An NC language needs commands such as *select tool*, *move tool to point (x,y)*, *lower tool, turn on coolant*. The process required a deep knowledge of cutting metal; it did not need to know about memory allocation or floating point.

It is important to recognize that individual initiative from below very much characterized the early history of computing - much more than standard top-down managerial decisions. Indeed, it took an unconscionable amount of time before the computing bill reached a level of managerial attention. It should not have been the cost, it should have been the *value* of computing that brought management to the punch. But it wasn't. I think the reason for that was that we computer folk were not particularly adept at explaining to anyone beyond our own circles what it was that we were doing. We were a corporate ecological intrusion which took some years to adjust to.

5.2 Information Consolidation at Boeing

It happened that computing at Boeing started twice, once in engineering and once in finance. My guess is that neither group was particularly aware of the other at the start. It was not until 1965 or so, after a period of conflict, that Boeing amalgamated the two areas, the catalyst being the advent of the IBM 360 system that enabled both types of computing to cohabit the same hardware. The irony here was that the manufacturing area derived the earliest company tangible benefits from computing, but did not have their own computing organization; they commissioned their programs to be written by the engineering or finance departments, depending more or less on personal contacts out in the corridor.

As Ken McKinley describes above, in the factory itself there were four different control media; punched Mylar tape, 80-column punched cards, analog magnetic tape and digital magnetic tape. It was rather like biological life after the Cambrian Explosion of 570 million years ago – on a slightly smaller scale. Notwithstanding, it worked! Much investment had gone into it. By 1960, NC was a part of life in the Boeing factory and many other American factories. Manufacturing management was quite happy with the way things were and they were certainly not looking for any more innovation. 'Leave us alone and let's get the job done' was their very understandable attitude. Nevertheless, modernisation was afoot, and they embraced it.

The 1950s was a period of explosive computer experimentation and development. In just one decade, we went from 1K to 32K memory, from no storage backup at all to multiple drives, each handling a 2,400-foot magnetic tape, and from binary programming to Fortran 1 and COBOL. At MIT, Professor Doug Ross, learning from the experience of the earlier NC languages, produced a definition for the *Automatically Programmed Tooling* (APT) language, the intention being to find a modern replacement for the already archaic languages that proliferated the 1950s landscape. How fast things were beginning to move suddenly, though it didn't seem that way at the time.

5.3 New Beginnings

Since MIT had not actually implemented APT, the somewhat loose airframe manufacturers' computer association got together to write an APT compiler for the IBM 7090 computers in 1961. Each company sent a single programmer to Convair in San Diego and it took about a year to do the job, including the user documentation. This was almost a miracle, and was largely due to Professor Ross's well-thought through specification.

When our representative, Henry Pinter, returned from San Diego, I assumed the factory would jump on APT, but they didn't. At the Thursday morning interdepartmental meetings, whenever I said, "*APT is up and running folks, let's start using it*", Don King from Manufacturing would say, "*but APT don't cut no chips*". (That's how we talked up there in the Pacific Northwest.) He was dead against these inter-company initiatives; he daren't commit the company to anything we didn't have full control over. However, eventually I heard him talking. The Aerospace Division (Ed Carlberg and Ken McKinley) were testing the APT compiler but only up to the point of a printout; no chips were being cut because Aerospace did not have a project at that time. So I asked them to make me a few small parts and some chips swept up from the floor, which they kindly did. I secreted the parts in my bag and had my secretary tape the chips to a piece of cardboard labeled 'First ever parts cut by APT'. At the end of the meeting someone brought up the question of APT. 'APT don't cut no chips' came the cry, at which point I pulled out my bag from under the table and handed out the parts for inspection. Not a word was spoken - King's last stand. (That was how we used to make decisions in those days.)

These things happened in parallel with Grant Erwin's development of the 727-CAD system. In addition, one of the facilities of even the first version of APT was to accept interpolated data points from CAD which made it possible to tie the one system in with the other in what must have been the first ever CAM-CAD system. When I look back on this feature alone nearly fifty years later I find it nothing short of miraculous, thanks to Doug Ross's deep understanding of what the manufacturing world would be needing. Each recourse to the surface definition was made in response to a request form the Engineering Department, and each numerical cut was given a running *Master Dimensions Identifier* (MDI) number. This was not today's CAM-CAD system in action; again, no screen, no light pen, no electronic drawing. Far from it; but it worked! In the early 1960s the system was a step beyond anything that anyone else seemed to be doing - you have to start somewhere in life.

6 Developing Accurate Lines

An irony of history was that the first mechanical movements carried out by computers were not a simple matter of drawing lines; they were complicated endeavors of cutting metal. The computer-controlled equipment was vast multi-ton machines spraying aluminum chips in all directions. The breakthrough was to tame the machines down from three dimensions to two, which happened in the following extraordinary way. It is perhaps one of the strangest events in the history of computing and computing graphics, though I don't suppose anyone has ever published this story. Most engineers know about CAD; however, I do not suppose anyone outside Boeing knows how it came about.

6.1 So, from CAM to CAD

Back to square one for a moment. As soon as we got the plotter up and running, Art Dietrich showed some sample plots to the Boeing drafting department management. Was the plotting accuracy good enough for drafting purposes? The answer - a resounding No! The decision was that Boeing would continue to draft by hand until the day someone could demonstrate something that was superior to what we were able to produce. That was the challenge. However, how could we meet that challenge? Boeing would not commit money to acquiring a drafting machine (which did not exist anyway) without first subjecting its output to intense scrutiny. Additionally, no machine tool company would invest in such an expensive piece of new equipment without an order or at least a modicum of serious interest. How do you cut this Gordian knot?

In short, at that time computers could master-mind the cutting of metal with great accuracy using three-dimensional milling machines. Ironically, however, they could not draw lines on paper accurately enough for design purposes; they could do the tough job but not the easy one.

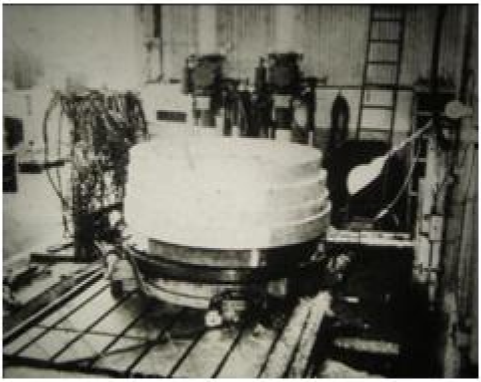
However, one day there came a blinding light from heaven. If you can cut in three dimensions, *you can certainly scratch in two*. Don't do it on paper; do it on aluminium. It had the simplicity of the paper clip! Why hadn't we thought of that before? We simply replaced the cutter head of the milling machine with a tiny diamond scribe (a sort of diamond pen) and drew lines on sheets of aluminium. Hey presto! The computer had drawn the world's first accurate lines. This was done in 1961.

The next step was to prove to the 727 aircraft project manager that the definition that we had of the airplane was accurate, and that our programs worked. To prove it they gave us the definition of the 707, an aircraft they knew intimately, and told us to make nineteen random drawings (canted cuts) of the wing using this new idea. This we did. We trucked the inscribed sheets of aluminium from the factory to the engineering building and for a month or so engineers on their hands and knees examined the lines with microscopes. The Computer Department held its breath. To our knowledge this had never happened before. Ever! Anywhere! We ourselves could not be certain that the lines the diamond had scribed would match accurately enough the lines drawn years earlier by hand for the 707. At the end of the exercise, however, industrial history emerged at a come-to-God meeting. In a crowded theatre the chief engineer stood on his feet and said simply that the design lines that the computer had produced had been under the microscope for several weeks and were the most accurate lines ever drawn - by anybody, anywhere, at any time. We were overjoyed and the decision was made to build the 727 with the computer. That is the closest I believe anyone ever came to the birth of *Computer-Aided Design*. We called it *Design Automation*. Later, someone changed the name. I do not know who it was, but it would be fascinating to meet that person.

6.2 CAM-CAD Takes to the Air

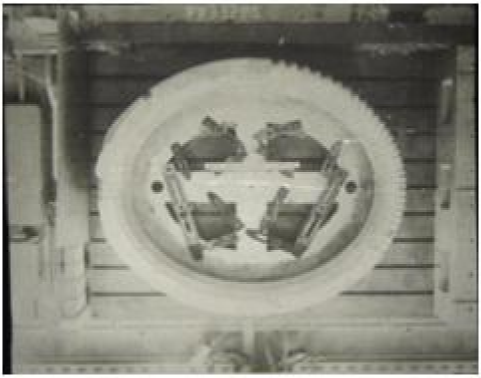
Here are pictures of the first known application of CAM-CAD. The first picture is that of the prototype of the 727. Here you can clearly see the centre engine inlet just ahead of the tail plane.

Seen from the front it is elliptical, as can be seen from the following sequence of manufacturing stages:-



1: Four rough cuts positioned on a Kearney & Trekker milling machine platform prior to cutting

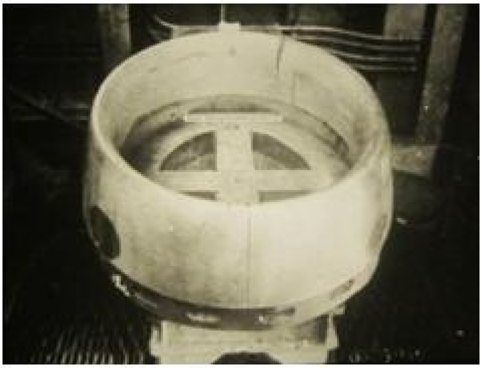
2: Cutting starts on the rough cuts



3: The elliptical first cut has been completed and the transverse parabolic cut has almost been completed.

4: The first two cuts have been completed.

5: The parabolic cut has been repeated with a smaller cutter.



6: The final smoothing cut has been completed and the part is ready for use in forming the inlet sheets.

6.3 An Unanticipated Extra Benefit

One of the immediate, though unantic ipated, benefits of CAD was transferring detailed design to subcontractors. Because of our limited manufacturing capacity, we subcontracted a lot of parts, including the rear engine nacelles (the covers) to the Rohr Aircraft Company of Chula Vista in California. When their team came up to Seattle to acquire the drawings, we instead handed them boxes of data in punched card form. We also showed them how to write the programs and feed their NC machinery. Their team leader, Nils Olestein, could not believe it. He had dreamed of the idea but he never thought he would ever see it in his lifetime: *accuracy in a cardboard box*! Remember that in those days we did not have email or the ability to send data in the form of electronic files.

6.4 Dynamic Changes

The cultural change to Boeing due to the new CAD systems was profound. Later on we acquired a number of drafting machines from the Gerber Company, who now knew that there was to be a market in computer-controlled drafting, and the traditional acres of drafting tables began slowly to disappear. Hand drafting had been a profession since time immemorial. Suddenly its existence was threatened, and after a number of years, it no longer existed. That also goes for architecture and almost any activity involving drawing.

Shortly afterwards, as the idea caught on, people started writing CAD systems which they marketed widely throughout the manufacturing industry as well as in architecture. Eventually our early programs vanished from the scene after being used on the 737 and 747, to be replaced by standard CAD systems marketed by specialist companies. I suppose, though, that even today's Boeing engineers are unaware of what we did in the early 1960s; generally, corporations are not noted for their memory.

Once the possibility of drawing with the computer became known, the idea took hold all over the place. One of the most fascinating areas was to make movie frames. We already had flight simulation; Boeing 'flew' the Douglas DC-8 before Douglas had finished building it. We could actually experience the airplane from within. We did this with analog computers rather than digital. Now, with digital computers, we could look at an airplane from the outside. From drawing aircraft one could very easily draw other things such as motorcars and animated cartoons. At Boeing we established a Computer Graphics Department around 1962 and by 1965 they were making movies by computer. (I have a video tape made from Boeing's first ever 16mm movie if anyone's interested.) Although slow and simple by today's standards, it had become an established activity. The rest is part of the explosive story of computing, leading up to today's marvels such as computer games, Windows interfaces, computer processing of film and all the other wonders of modern life that people take for granted. From non-existent to all-pervading within a lifetime!

7 The Cosmic Dice

Part of the excitement of this computer revolution that we have brought about to these sixty years was the unexpected benefits. To be honest, a lot of what we did, especially in the early days, was pure serendipity; it looked like a good idea at the time but there was no way we could properly justify it. I think had we had to undertake a solid financial analysis most of the projects would never have got off the ground and the computer industry would not have got anywhere near today's levels of technical sophistication or profitability. Some of the real payoffs have been a result of the cosmic dice throwing us a seven. This happened already twice with the first 727.

The 727 rolled out in November, 1962, on time and within budget, and flew in April, 1963. The 727 project team were, of course, dead scared that it wouldn't. But the irony is that it would not have happened had we not used CAD. During the early period, before building the first full-scale mockup, as the computer programs were being integrated, we had a problem fitting the wing to the wing-shaped hole in the body; the wing-body join. The programmer responsible for that part of the body program was yet another Swiss by name Raoul Etter. He had what appeared to be a deep bug in his program and spent a month trying to find it. As all good programmers do, he assumed that it was his program that was at fault. But in a moment of utter despair, as life was beginning to disappear down a deep black hole, he went cap in hand to the wing project to own up. "*I just can't get the wing data to match the body data, and time is no longer on my side*." "*Show us your wing data. Hey where did you get this stuff?*" "*From the body project.*" "*But they've given you old data; you've been trying to fit an old wing onto a new body*." (The best time to make a design change is before you've actually built the thing!) An hour later life was restored and the 727 became a single numerical entity. But how would this have been caught had we not gone numerical? I asked the project. At full-scale mockup stage, they said. In addition to the serious delay what would the remake have cost? In the region of a million dollars. Stick that in your project analysis!

The second occasion was just days prior to roll-out. The 727 has leading-edge flaps, but at installation they were found not to fit. New ones had to be produced over night, again with the right data. But thanks to the NC machinery we managed it. Don't hang out the flags before you've swept up the final chip.

8 A Fascinating Irony

This discussion is about using the computer to make better pictures of *other things.* At no time did any of us have the idea of using pictures to improve the way we ran computers. This had to wait for Xerox Park, a decade or so later, to throw away our punched cards and rub our noses into a colossal missed opportunity. I suppose our only defence is that we were being paid to build airplanes not computers.

9 Conclusion

In summary, CAM came into existence during the late 1950s, catalyzing the advent of CAD in the early 1960s. This mathematical definition of line drawing by computers then radiated out in three principal directions with (a) highly accurate engineering lines and surfaces, (b) faster and more accurate scientific plotting and (c) very high-speed animation. Indeed, the world of today's computer user consists largely of pictures; the interface is a screen of pictures; a large part of technology lessons at school uses computer graphics. And we must remember that the computers at that time were miniscule compared to the size of today's PC in terms of memory and processing speed. We've come a long way from that 727 wing design.

Roberto Busa (1913-2011), Pioneer of Computers for the Humanities

Corrado Bonfanti

Associazione Italiana per l'Informatica ed il Calcolo Automatico (AICA), Italy
corradobonfanti@hotmail.com

Foreword: Roberto Busa, *Societatis Iesu*, passed away on August 9, 2011.

I will commit a memory of such a remarkable and unusual personality to a concise – as well as arbitrary – excerpt from his own writings.

Before doing that, I ought to recall the last opportunity I had to meet Father Busa.

In the springtime of 2008, at the outset of a conference, I had the pleasure to drive him in my small car to the airport whence he had to depart. (Aged 95, he still travelled unperturbed on his own!) After so many years of formally respectful friendship, I suddenly felt the need to loyally disclose to him my personal distance from religions, included the one he so vividly professed. He listened quietly, without any comment. Then, at the moment of departure, he unexpectedly said "It's time we call us simply by name. Corrado – he smiled – you are free to call me father Roberto".

Keywords: Humanities, computing, informatician, life history.

1 A Priest and an Informatician

I am aware that the appearance of a priest[1] into the *ambulacra* of technology might appear rather bizarre: I feel I looked like a dromedary slipping into the stock exchange.

I ought to tell you: firstly, it was not my own choice but obedience to an assignment or mission; and this gives great peace and strength.

Secondly, also and precisely because I am a priest, I feel comfortable there: not only because of sympathy and admiration for informaticians; as a matter of fact, informatics is an interior and spiritual discipline.

Since man is child of God and technology is child of man, I think that God regards technology the way a grandfather regards his grandchild[2]. And for me personally it is satisfying to realize that I have taken seriously my service to linguistic research.

[1] Translated from [1], p.81. Last paragraph taken from [2], pp. 87-88.
[2] To be read 'information technology', according to context.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 57–61, 2012.

2 "It was not my own choice"

I entered the Jesuit order[3] in 1933. I was then 20. Later my superior asked me: "*Would you like to become a professor?*" "*In no way!*" My wish was to be a missionary to take care of the poor. "*Good. You'll do it, all the same*."

By 1941, I had been assigned to work towards a PhD in Thomistic philosophy at the Pontificial Gregorian University in Rome. My research was aimed at exploring the concept of 'presence' according to Thomas Aquinas. While I was involved with this research, two major considerations became evident. I realized first that a philosophical and lexicographical inquiry into the verbal system of an author has to precede and prepare for a doctrinal interpretation of his work.

In 1946 as a result of these preliminary conclusions, I started to think of an *Index Thomisticus*, i.e. a concordance of all the words of Thomas Aquinas, including conjunctions, prepositions and pronouns, to serve other scholars for analogous studies. It was clear to me, however, that to process texts containing more than ten million words, I had to look for some type of machinery.

In 1949, I visited approximately 25 American Universities from coast to coast, asking about any gadget that might help in producing the type of concordance I had in mind. Jerom Wiesner of M.I.T., sent me to IBM in New York City, where someone was assigned to examine my project.

I knew, the day I was to meet Thomas J. Watson, Sr., that he had on his desk a report which said that IBM machines could never do what I wanted[4]. I had seen in the waiting room a small poster imprinted with the words: "*The difficult we do right away; the impossible takes a little longer*" (IBM always loved slogans). I took it with me into Mr. Watson's office. Sitting in front of him and sensing the tremendous power of his mind, I was inspired to say: "*It is not right to say 'no' before you have tried.*" I took out the poster and showed him his own slogan. He agreed that IBM would cooperate with my project until it was completed 'provided that you do not change IBM into International Busa Machines.' I had already informed him that, because my superiors had given me time, encouragement, their blessings and much holy water, but unfortunately no money, I could recompense IBM in any way except financially. That was providential! [5]

Although some say that I am the pioneer of the computers in the humanities, such a title needs a good deal of nuancing. *A propos* of this, Mr. Lee Loevinger in the *Minnesota Law Review* (April 1949), in an article on jurimetrics said: "*Machines are now in existence which have so far imitated thought processes that they can solve differential equations. Why should not a machine be constructed to decide lawsuits?*" And on the stacks of the IBM library in New York City I had spotted a book (whose title I have forgotten), which was printed sometime between 1920 and 1940: in it someone mentioned that it was possible to make lists of names by means of punched cards. Maybe others too may claim that they have worked in this area prior to me. Yet, isn't it true that all new ideas arise out of a *milieu* when ripe, rather than from any one individual? If I was not the one, then someone else would have dealt with this type of

---

[3] A combination of fragments taken from [2].

[4] Thomas J. Watson Sr. was the mighty IBM's father-master. IBM machines, by that time, were punched card equipment.

[5] As a matter of fact, IBM generously supported Busa's cultural enterprise until its completion.

initiative sooner or later. To be the first one to have an idea is just a chance. If there is any merit, it is in cultivating the idea. During the following years I experienced the wisdom of another slogan, attributed to an American, Thomas Edison: "*Genius is 1% inspiration, 99% perspiration*."

In addition to the 10,600,000 words of *Index Thomisticus*, I processed five million more words, Italian, English, German, Russian, ancient Greek and Hebrew, Aramaic and Nabatean. The subjects ranged from nuclear physics and mathematics abstracts to Qumran [or Dead See] Scrolls and works of Dante, Kant, and Goethe. The processing of the Dead Sea Scrolls was publicized throughout the world on the front pages of newspapers in the spring of 1958. I was able to complete my *Index Thomisticus* 33 years after the conception of the project and 30 years after my first meeting with IBM. At that time it was the first undertaking in computer linguistics, the only published work of its kind with such dimensions and such characteristics.

3 Father Roberto in Numbers

Besides 245 congresses[6], I gave courses, lectures or conferences in 60 Italian sites, plus 40 in Europe, plus 17 in Northern America, plus 9 in Asia and Africa, plus 3 in Southern America. At this date [March, 2008], my written works are 421: 116 are books, amounting to 80,000 pages, 305 are articles and papers for more than 5,000 pages. Not counting those in Italian or Latin, 86 are written in or translated into Albanian, Hebrew, French, Georgian, English, Portuguese, Russian, Spanish, German.

> *Father Roberto avoided enumerating the awards he received along his life. I can testify his pride on being bestowed upon the highest honour of the Order of Merit of the Italian Republic; President Carlo Azeglio Ciampi created him a Knight Grand Cross in 2005.*
>
> *It is also worth mentioning that he was an invited speaker in a number of conferences held by AICA, the IFIP's federate Italian Computer Society. On the occasion of IFIP's WCC 2008 held in Milano, AICA issued an illustrated poster to recognize the Italians who markedly contributed to the progress of informatics and its applications; Roberto Busa is among them.*

4 Passages from His Life

Roberto Busa was born in 1913 in Contrada Busa di Lusiana[7] – near Vicenza, Northern Italy – whence the name of his family. He completed high school and the first two years of Theology at the episcopal seminary of Belluno, being there a schoolmate of Albino Luciani, later to become Pope Giovanni Paolo I. In 1933 he entered the Jesuit order, to be ordained priest in 1940. He gained the doctorate in philosophy (1937) as well as in theology (1941). In the years 1940-43 he served as a military auxiliary chaplain in the army and then with the partisan forces. At the Faculty of philosophy

---

[6] Translated from [3].

[7] Translated, with minor adaptations, from [4].

Aloisianum in Gallarate, near Milano, he was full professor of ontology, theodicy and scientific methodology; for several years he also served there as librarian.

In 1946 he conceived the project of an *Index Thomisticus*. In 1949 he started experimenting linguistic computerization at the IBM offices in New York and Milano. In order to manage and to exploit operations, he founded CAAL[8]. Operations were based at Gallarate until 1967; then for two years at Pisa and for two more years at Boulder, Colorado and finally, for nine years, in Venezia. Computerized photocomposition of the *Index Thomisticus* started there in 1974 and was concluded in 1980: 70,000 pages in 56 large size volumes.

During the same decades, the rhythm of international promotion for linguistic research and analysis was marked by the 143 congresses father Busa actively participated in, across three continents.

In 1967 prof. Antonio Zampolli – up to then an assistant of father Busa – founded in Pisa the Institute for computational linguistics which soon gained international reputation.

In 1983 a new association succeeded CAAL: it was named CAEL and housed at the *Aloisianum* of Gallarate[9]. Many Centres, from Israel to Czechoslovakia, Belgium, France and Germany have been inspired by Gallarate experience.

Later on, father Busa established the CIRCSE at the Sacred Heart's University of Milano, where he currently lectures Thomistic lexicography and lexicology and Computer linguistics[10]. He also gives courses of Text analysis by computer and Thomistic hermeneutics at the Gregorian Pontificial University in Roma.

> *This writing dates back to 1991 and could not capture the manifold enterprises to which father Roberto devoted the last and fruitful period of his life. Beside a series of lectures at the Polytechnic of Milano about philosophy and psychology in Artificial Intelligence and Robotics, let me at least mention the weighty project he named Disciplined Languages, aimed at automatic translation, a recurring concern of AI.*
>
> *In this project he likely resumed seminal ideas he conceived during early discussions with the cybernetics Eduardo Caianiello and Norbert Wiener and, more specifically, while participating in research, sponsored by EURATOM, for automatic translation of Russian scientific abstracts. The research was abandoned due to the unfair conclusions of the 1966 ALPAC report, after which the same US government discontinued funding Machine Translation*[11].

5 Sources

1. Busa R.: "Cinquant'anni a bitizzar parole"; in Convegno internazionale di storia e preistoria del calcolo automatico e dell'informatica – Siena, 10-11-12 settembre 1991; AICA, Milano, 1991, pp. 71-82.

---

[8] CAAL: Centro Automazione Analisi Linguistica.

[9] CAEL: Computerizzazione delle Analisi Ermeneutiche e Lessicografiche.

[10] CIRCSE: Centro Interdisciplinare per le Ricerche della Computerizzazione dei Segni dell'Espressione.

[11] ALPAC: Automatic Language Processing Advisory Committee.

2. Busa R.: “The Annals of Humanities Computing: the Index Thomisticus”; *Computers and the Humanities*, Vol. 14, n. 2, October 1980, pp. 83-90.
3. Busa R.: “*Report to my Superiors – March 3rd, 2008*”; not published, friendly copied to author.
4. Busa R.: *Author’s Profile*; annexed to [1].

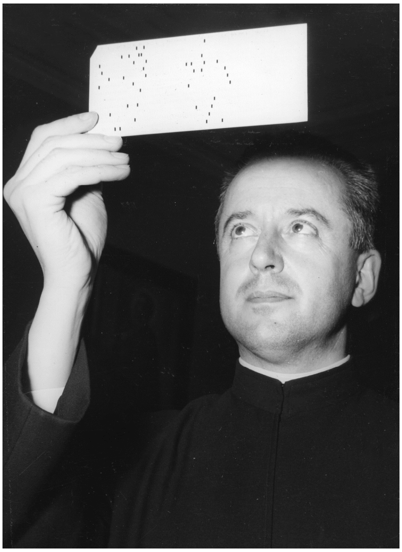


Fig. 1. Roberto Busa while “ostending” a punched card. (About 1965). Courtesy of CAEL

Fig. 2. Together with Norbert Wiener, at CAAL. (1958. Courtesy of CAEL)

Fig. 3. Roberto Busa decorated as Knight Gran Cross. (Roma, 2005. *CAEL Newsletter*, Dec. 2006)

Fig. 4. Father Roberto annually issued a CAEL Newsletter bringing, among other, a parody of himself sketched by artist Marina Molino. This one appeared *in memoriam*, featuring him finally reaching, in heaven, his beloved mentor Thomas Aquinas. (2011. Courtesy of CIRCSE)

Micro Programming

Herman Spanjersberg

h.a.spanjersberg@planet.nl

Abstract. In the 1970s a need arose to perform special arithmetic operations on minicomputers much more quickly than had been possible in the past. This paper tells the story of why micro programming was needed for special arithmetic operations on mini computers in the 1970s and how it was implemented. The paper tells how the laboratory in which the first experiment took place had a PDP-9 minicomputer from Digital Equipment Corporation and how the author, with several colleagues, after attending a course for the technical service of a PDP-9 given by a specialist from Digital Equipment, knew exactly which signals flew through the machine at any time. The paper describes how by having 'programmable' control memory they were able to make changes in the execution of instructions.

Keywords: Micro programming, minicomputer.

1 Introduction

In the late seventies of the last century there came a need for special arithmetic operations with a (mini)computer. One of the reasons was that researchers wanted to collect and adapt measurement data within a few microseconds: in some cases the result of the operations had to be fed back to the process to control the process or the character of the research project demanded a rapid insight in the converted data. For example there was a research project in an ophthalmologic laboratory where one needed an instant fast Fourier transform of the measured signals. With the normal processors this was impossible, or better formulated: it took too long time with normal 'general purpose' computers. Based on an article of Professor Wilkes[1] we started an experiment on a standard minicomputer with the theme *'microprogramming*'. To understand that technique at first a short description is given of processor[2] architecture.

2 CPU Architecture

In Fig. 1 a general architecture of a processor is sketched. The heart of the processor is the arithmetic and logical unit (ALU) to perform the arithmetic and logical operations.

[1] Sir Maurice Wilkes was Director of the Cambridge Computer Laboratory throughout the whole development of stored program computers starting with EDSAC; inventor of labels, macros and microprogramming.

[2] In those days indicated as the central processing unit (CPU).

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 62–68, 2012.

The ALU is connected with three busses: one at each entrance (A and B) and one at the exit (C). The busses form together with the ALU a transport channel for data between several registers (at one combination of control signals the ALU transmits only the data from one entrance to the exit). Some registers are implicit in use, such as the program counter (PC) and the instruction register (IR) while other registers have to be addressed explicitly in the instruction.

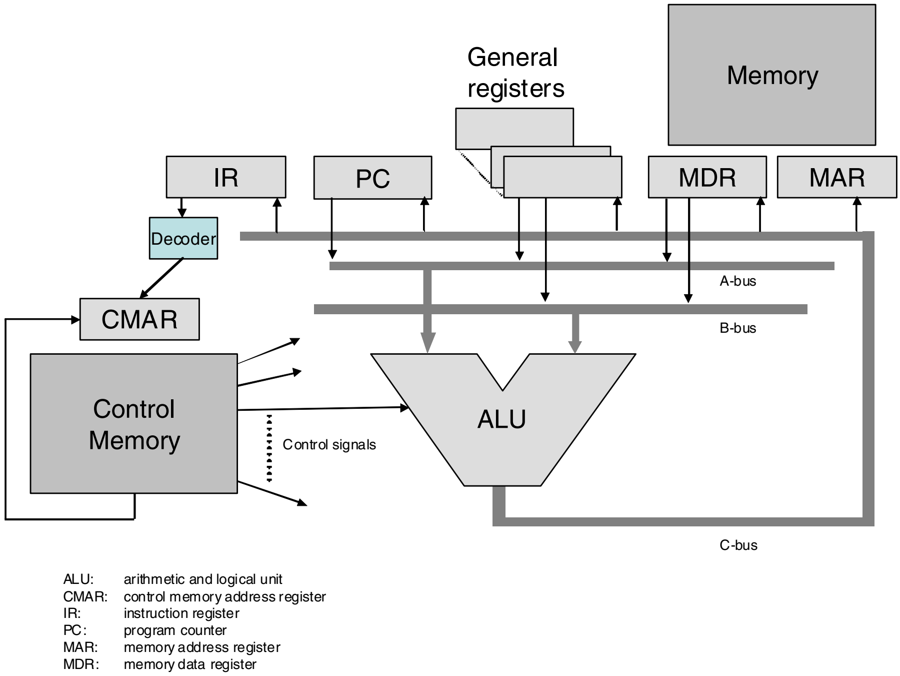


Fig. 1. CPU outline

By placing an extra register at the entrance of the ALU one can delete one bus and by adding a further extra register at the second entrance even one bus suffices. It will be clear that the speed of instruction execution in that case proportionally decreases.

The whole process of instruction execution is controlled by signals coming from the control memory (CM).

3 Instruction Architecture

To fully understand the working of a traditional computer we look not only at the hardware, but also at the instruction format. Fig. 2 represents a general format.

operation code	source address 1	source address 2	destination address

Fig. 2. Instruction format

The first part of the instruction indicates what operation has to be executed. The other parts indicate in which memory places the operands are found and where the result has to be placed. The three addresses are not always specified. To understand the ratio for that one has to consider that in most architectures for reasons of execution speed the length of an instruction (number of bits) is equal to the length of the words in main memory: in that case only one memory cycle is needed to fetch an instruction. But the larger the main memory is, the longer is the address. At the other hand one wants to have a large number of different operations, so ideally the operation code is long. These wishes are conflicting so that the designer has to seek for an optimum.

To begin with one can reduce the number of addresses in the instruction by always placing the result in source address 2. The price one has to pay for that reduction is that one of the two original operands is lost. A further reduction can be reached when by definition one operand is the content of a register (the number of registers is always far less than the number of addresses in the main memory so one needs but a small number of bits to address a register). That requires sometimes an extra operation to place the operand in that register with as a consequence that the program execution slows down. In some computers even one special register was implicitly indicated to contain one operand and afterwards the result.

In the first minicomputers that came on the market in the late sixties (like the PDP-8 from Digital Equipment Corporation) the instruction had one address and the other operand was supposed to be in a special register (called the accumulator), while the result was placed back into the accumulator. Later on there came computers on the market, like the PDP-11 from the same company, that had a complex operand specification: two address parts, each part divided into a register field and a mode field. The register field indicated a register that was supposed to contain a memory address. The mode field specified the manner in which that address had to be used (e.g. direct or indirect).

To keep it simple we will use in the following part of this paper an instruction type with one memory address that points to one operand and an implicit specified register containing the other operand and receiving the result.

4 Instruction Execution

Most instruction cycles can be divided into three phases:

The instruction fetch phase
The data fetch phase[3] and
The execution phase.

[3] Only the instructions influencing the sequence of instruction execution, such as jump instructions, don't have a data fetch phase in general.

Each phase can in its turn be divided into several steps. Each step corresponds with one word from the CM: the microinstruction.

The microinstruction can in general be described with Figure 3.

next CM address	A-bus	B-bus	C-bus	ALU	CS

Fig. 3. General outline of a microinstruction

Each CM word contains the address of the next microinstruction to read. Further there are three fields that control the output and the input of the several registers on the buses. The next field defines the function the ALU has to do. The last field specifies a number of independent control signals (like a command to read main memory). A quite simple architecture required a word length of roughly 80 bits.

The last microinstruction of each execution phase points to the first one in the fetch phase for the next instruction. The fetch phase is the same for every instruction and is found in a fixed place in CM. The program counter (PC) contains the address of the next instruction to be executed. That instruction itself is still unknown until it is fetched from the main memory. So the content of the PC register is fed via the ALU to the memory address register (MAR) and thereafter a read command is given to the memory. As a result of the read operation the word in that place – the instruction to be executed – comes out in the memory data register (MDR) and is further transported to the instruction register (IR). In between the content of the PC has been incremented with one so that the PC already points to the next instruction. The first part of the IR, indicating the wanted operation, is connected to a decoder that gives an address of the control memory. That address is put into the CMAR and by starting a read operation for the CM the first step of the new phase is started.

Knowing now what operation has to be executed the question arises on which operands it has to be done. Finding the operands is closely related to the design[4] of the instruction. As indicated earlier we take the most simple manner in this paper: an address with an extra bit indicating that the address is direct or indirect[5]. The address part of the IR is now transported to the MAR and a memory read operation is started to fetch the specified operand. If the '*indirect bit*' in the instruction is 1 an extra memory read cycle has to be executed because the address part of the instruction does not point to the operand itself but to the place where the address of the operand can be found.

After the operand has been fetched from main memory the content of MDR is offered to the one input gate of the ALU and the content of the accumulator to the other gate. At the same time a command is given to the ALU to perform an arithmetic or logical operation and the output is fed back to the accumulator.

---

[4] In some architectures, like the PDP-11 family from Digital Equipment, there are several addressing modes that require some arithmetic to determine the place of the operands.

[5] With direct addressing the address points to the operand, while indirect addressing means that the address points to a location which contains the address of the operand.

5 The Idea of Microprogramming

In most minicomputers being on the market in the late seventies the CM was constructed as a read only memory (ROM). That was already a step forwards in relation to older machines where the control part was a specially designed complex of counters, gates, etc. – the design with a CM gave a clear architecture as we saw in the preceding paragraphs. But the combination of the need to perform special operations with this well designed architecture gave the idea to replace the CM by another, a variable one if that was possible.

6 The First Experiment

The laboratory in which the first experiment took place had a minicomputer of the type PDP-9 from Digital Equipment Corporation. This machine still was composed of lumped components: transistors, resistors, etc. With several colleagues we followed a course for technical service of that machine, given by a specialist from Digital Equipment so we knew after the course exactly which signals flew through the machine at any time.

Our first aim was to make the machine run as delivered, but now with a new CM. To run the original programs the content of that CM had to be the same as the original one. Because of the wish to make it variable we took a matrix board[6] (Fig. 4) with enough rows and columns to let the machine run its original instruction packet.

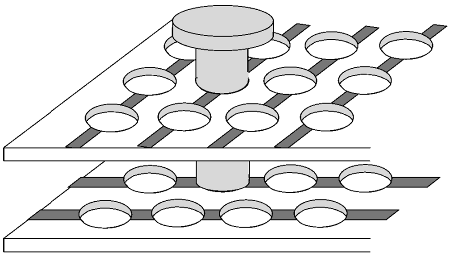


Fig. 4. Schematic detail of a matrix board

We were lucky in so far that the original ROM consisted of separate magnetic cores, so we could use the original electronic circuits that select a row as the result of a handed over bit combination (indicating the CM 'address').

Having a 'programmable' CM we were able to make changes in the execution of instructions. For example we could implement 'incremental indirect addressing' whereby the address of the operand is automatically incremented each time the operand is used. This way of addressing saves an instruction and thus quickens the handling of data in a list: supposing this data handling requires a number of instructions, the whole list can be processed by repeating these instructions again and again with the same instructions and the same addresses as illustrated in Figure 5.

[6] As was used to program analog computers or, in older days, Hollerith machines.

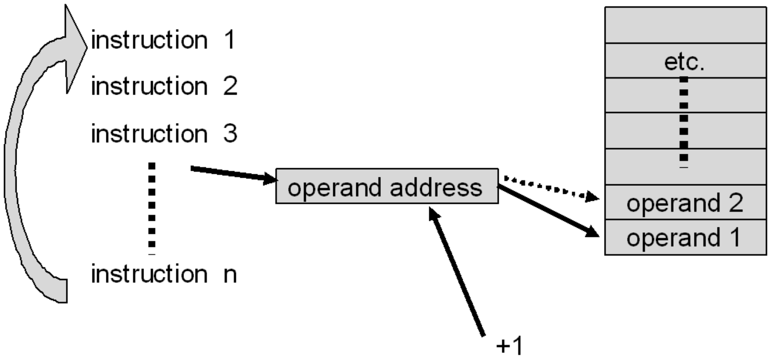


Fig. 5. Incremental indirect addressing

7 Extensions

Having a changeable ROM the idea arose to make the content more flexible and programmable by using a random access memory (RAM).

This idea was compelling but we had to realize that by replacing the whole CM with a different content the original instruction execution was destroyed. So it was decided only to add a new part to the CM with RAM. That however presented several problems. At first there was a slight technical problem: the electric signals that came from the ROM didn't agree with the electric power levels of the signals that came from standard RAM devices. That problem could simply be tackled. But a severe problem was filling the CRAM[7]: how and with what data?

To fill the CRAM we found a solution in regarding the CRAM as a special output device: with I/O-instructions the CRAM could be filled from the general memory via the I/O-bus. The only obstacle was that the word length of the CRAM didn't agree with the width of the data path on the I/O-bus. But that could simply be avoided by using a number of data transfers for one word in the CRAM. We decided always to fill the CRAM with a block transfer because we saw no need to replace single words of the RAM. Filling the CRAM was now comparable with writing a disk drive.

8 Micro-assembler

Now the problem still remained on how to define the content of the CRAM. In a brain storming session the idea arose to look how the general memory was filled with instructions. At a high level we used programming languages like FORTRAN, Algol etc. But at a lower level the manufacturer designed an assembler language that facilitated writing programs – e.g. the drivers for the peripherals – in symbolic instructions (e.g. the instruction to add two numbers was indicated with the acronym ADD), with

[7] To avoid long descriptions we use the acronym CRAM for the control memory with RAM.

symbolic addresses. The assembler program converted the symbolic instructions into rows of bits, so we designed a micro-assembler. Now we could program new instructions and even test them before implementing them.

9 Commercial Products

At about the time we got the micro-assembler, two manufacturers of minicomputers launched a new model of their minicomputer family: Digital Equipment Corporation put the PDP-11/60 on the market and Hewlett Packard a model of the HP-2100A. Both had all the benefits we already discovered.

As far as I know these were the only commercial available machines with micro-programming facilities and they were at the same time the last scions on the family tree of small minicomputers, because microprocessors and PCs soon came along.

Experiences and Reflections

Harold "Bud" Lawson

To appear in the IFIP – Springer LNCS Series – History of Computing – Processing Memories and Sharing Stories, Edited by Arthur Tatnall

Abstract. I have divided this personal history into three phases; namely computer industry, computer-based systems and complex systems. While the years indicated are approximate and there is significant overlap, it provides a structure to conveying the essence of my career. I have worked with many talented people during my career and we have learned from each other. I thank them all and identify many of them. Further, I take the opportunity to provide some reflections that may have occurred during the experience or in several cases later in my career. I also provide a summary of what I consider to be my most important publications.

Keywords: personal history, computer industry, computer-based systems, complex systems.

Phase 1 (1959 to 1974) – Computer Industry

The period involved is from June 1959 to sometime in early 1974. During this time, I was employed first by Remington-Rand Univac, then by International Business Machines, Inc, later by Standard Computer Corporation (Costa Mesa, California) and finally as a consultant to Datasaab (Linköping, Sweden). Also during this period starting in 1967, I began a parallel academic career, first as Associate Professor and later as Professor at the Polytechnic Institute of Brooklyn as well as guest professorships at the University of California at Irvine and Linköpings University in Sweden. This was a very exciting period in the computer industry and it turned out to be a very productive period of my life providing a breadth of experiences and knowledge.

I will attempt to name those persons that I closely associated with during this phase of my career. However, the most important person was the late Rear Admiral Dr. Grace Murray Hopper who was my first boss at Remington-Rand Univac (formerly Eckert-Mauchly Computer Corporation). Amazing Grace has a special place in my heart. From her I learned to question the status quo, to seek deeper meaning and at least implicitly to always try to find the guiding concepts and principles. This has continuously affected my approach to the many problems and opportunities that I have faced in dealing with complex systems for over 50 years.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 69–88, 2012.

Where Did It Start?

I first became exposed to computer technology as a summer student at the United States Census Bureau in Suitland, Maryland during the summer of 1958. A fellow Temple University colleague, Zigmud Decker, who was doing his second summer student tour, and I shared an apartment in Suitland. Ziggy had luckily in both of his tours had the opportunity to learn programming for the Univac 1103 at the Census Bureau. I, on the other hand, was assigned to task for analyzing problems and discrepancies in the reporting of cotton linter production. The summer program administrators did organize a two-day course about computing at the census bureau. This introduction along with many late night discussions with Ziggy about computing got me hooked on the idea of pursuing a career in this up and coming area.

During my senior year at Temple, I was employed as the lab instructor in the Department of Statistics under the direction of Professor Rosella James. That year, Temple decided to purchase an IBM 650 drum memory computer. Rosella arranged that I, along with faculty members, could take an introductory programming course (learning about the assembly language SOAP). Again this confirmed my interest in computing. Also during the year Rosella and I attended a seminar at the University of Pennsylvania given by none other than Grace Hopper. She spoke about teaching computers to understand programs expressed in English. This really aroused my interest. So, when seeking employment I applied to her Automatic Programming Department at Remington-Rand Univac and was employed starting in June 1959.

Remington-Rand Univac

Compiler Experiences

When I arrived at Univac I was issued a pile of documents including the Univac I, Univac II, Flowmatic, Math-Matic and Assembly Language manuals and documentation of the B-0 Flowmatic compiler. At the same time I was given an assignment to modify a code generator in the B-0 compiler. What a swim or sink introduction! I was really glad that I previously had an introduction to computing otherwise I might have sunk. Through lots of hard work and by asking questions of my more senior colleagues, I received a good start in understanding what compilers where all about. As another part of my initiation into the field, I was given the assignment to design and write a sample payroll application in Flowmatic for the City of Philadelphia (they were considering buying a Univac II). Further, I designed and wrote a multi-variable regression (correlation) routine in Math-Matic.

During this period, the CODASYL committee was developing the COBOL programming language. Our Automatic Programming Department at Univac in Philadelphia as well as a correspondent group at RCA in Camden, NJ developed the initial Cobol implementations. The architecture of the compiler for the Univac II (a machine with 2000 words of memory and 10 tape drives) was developed by our senior people including Bill Finley, Tom Jones, Dick Miner and Dan Goldstein. These gentlemen assigned two vital phases of the compiler implementation to me; namely the initial source program parser as well as the most vital code generation phase.

During the latter part of 1959 and during 1960, I worked diligently (many 60-70 hour weeks) on designing and implementing these critical elements. Most interesting was the code generation aspect for which I developed a service network of procedures that generated the machine code. I dealt with each type of package of verb and arguments via simple control sequences that invoked the service network. (A service oriented architecture). Most amazing was that I managed to contain the service network in about 1k words leaving adequate space for the control sequences for all of the common cases. An overlay area was used to swap in (from the compiler program tape file) control sequences for more rarely used cases. I called this phase of the compiler DUZ, named after a popular detergent advertisement at that time (DUZ does everything). Since the Univac II was a decimal machine with only decimal arithmetic (no floating point) it was necessary to develop algorithms for calculating shift distances in decimal alignment. I developed and wrote an internal paper on these generalized decimal alignment algorithms.

A most interesting part of the Univac II Cobol implementation is that it was written in a higher-level language, namely, Flowmatic. This was due to the fact that Grace demanded that we view the compiler as a data processing system. Even the small amount of code for the DUZ service network was the result of Flowmatic compiled code. Some critical machine instructions were incorporated in the Flowmatic source code through a mechanism I introduced into Flowmatic called Supplex code.

The Univac II Cobol compiler was composed of about 60 runs (defined transformations with one or more tape input files and one of more tape output files). It took a long time to compile programs and so there where several checkpoints in the compiler. Thus if a machine failure occurred, it would not be necessary to restart, but instead back up to the last checkpoint.

The day of reckoning was in December 1960 when the CODASYL dignitaries showed up one day at RCA in Camden and the next day at Univac in Philadelphia. For the first time in computer history a source program was compiled and executed on two different machines from two computer manufacturers.

A personal note about my relationship with Grace Hopper: On the 9th of December 1960 my daughter Catherine Louise Lawson (now faculty member at Rutgers University) was born on Grace's birthday. Grace was very happy about this and knitted a suit (she really enjoyed knitting) and also presented Cathy with a gold necklace.

During the spring of 1961 Remington-Rand Univac was acquired by the Sperry Corporation. Plans were being made to provide a Cobol Compiler for newly announced Univac III (which had a different architecture than its predecessors the Univac I and II). At this time we had moved from the primitive facilities at 19th and Allegheny Avenue (an old dusty and dirty Exide battery warehouse) to a new luxurious building in Blue Bell, Pennsylvania. Unfortunately, Grace Hopper fell into disfavor with the new Sperry management (she was not a very "manageable" person). This made the future of the Automatic Programming Department uncertain and several employees including myself left at varying points during 1961.

Reflections

I earlier mentioned the tremendous impact Grace Hopper had upon my career, but my other colleagues at Univac provided significant guidance and counseling as well. A period of my life that is hard to forget with many laughs, serious moments and a hell of a lot of hard work.

I often long for the simplicity of the Univac I and Univac II architectures. Being decimal machines with 12 digit words and a limited but powerful instruction set, it was easy to program even at the assembly language level. This simplicity has certainly remained in my mind since then. On the other hand, one had to be programmer, operator and at times repairman for the machines. I thank Norman Rothberg for teaching me the practical skills needs for operation and even repair during many late hours of the night and early mornings.

Fig. 1. Univac II – Fond Memories

Cobol for the Univac II was not widely used since the machine was no longer produced after 1961 and Sperry went over to the Univac III as well as eventually a Cobol implementation for the 1100 series of machines.

In 1999, at the invitation of Len Shustek (a former student of mine and Chairman of the Board of the Computer History Museum), I along with Howard Bromberg (earlier also an employee of Grace Hoppers department) and later manager of the RCA Cobol effort presented a seminar at Stanford University. This was a stimulating event where I presented the architecture of the Univac II Cobol compiler and Howard presented several interesting perspectives concerning the development of COBOL and Grace Hopper. Amongst many well-known attendees was Donald Knuth who will be mentioned again later. Gwen and Gordon Bell graciously provided a buffet dinner in their home after the seminar. I met up with many old friends and had a chance to reflect on the good old days with Don Knuth.

International Business Machines, Inc.

Further Compiler Experiences

After having very positive job interviews and job offers from RCA in Camden and General Electric in Phoenix, I decided to join IBM at Poughkeepsie, New York and started in August 1961. The goals for my employment included research work in the area of compiler automation technology (compiler-compilers as they became to be known). I first worked with Rainer Kogen exploring the application of the ideas as developed in X-Tran (a string parsing language). However, early in 1962 I was asked to transfer to an ongoing project at IBMs Federal Systems Division in Bethesda, Maryland. IBM does much to earn its alternative designation (I've Been Moved). The SLANG (Systems Language) project was started by Robert Sibley and there were four researchers in the group including the now famous but tragically deceased Don Estridge (mentioned again later).

The SLANG compiler-compiler had the goal of describing and generating defined parsing algorithms (from a notation similar to Backus-Nauer Form) that could be used in quickly building translators for any programming language. A further goal was a general-purpose back end that could generate machine code for any of IBMs machines. The SLANG compiler was first implemented first in Fortran and then implemented in typical compiler-compiler fashion in SLANG itself. SLANG operated on the IBM 7090 and could be ported to other machines. Thus it was aimed at supporting compiler development for traditional second-generation machines like 7090, 7070, 7080, 1401 and 1410.

First Professional Publication

While working on the treatment of data declaration processing in the SLANG project and also largely based upon experiences in the Cobol implementation at Univac, I produced my first professional publication and presented it at the ACM National Conference in Syracuse, New York in 1962.

The paper entitled: ***"The Use of Chain List Matrices for the Analysis of COBOL Data Structures"*** is referenced in Donald Knuth's "Art of Computer Programming "Volume 1, as being the first publication of algorithms for multi-linked data structures.

Reflections

The basic concepts of SLANG were pioneering. Later, parser generation and compiler-compilers were further developed by other research groups. For example as a part of the Unix effort at Bell Telephone Laboratories compiler-compilers became more or less a standard well-known approach.

The SLANG project was moved from Bethesda, Maryland to Poughkeepsie, New York (from where I had come) and was then in the Systems Development Division. Also, several new people where employed. In addition to Robert Sibley and Don

Estridge other colleagues I recall included Bob Rosenblom, Jim Young, Dick Maeir and Richard Bailey. Further, I promoted the recruitment of three former members of Grace Hoppers team into the SLANG project; namely John Tronoski, Stanley Park, and Vincent Della Valle.

We worked hard with many late nights to gain computer time access. Further playing bridge became an obsession with many of us and we played at every chance we could both at work and in the evenings. Touch football also was a means of recreation.

One event I vividly remember while the project was still in Bethesda is a weekend that Don Estridge enlisted our support in painting the outside of his house. Don bought a lot of beer and we were about 5 people that showed up to paint. The house sidings (shingles) just sucked up the paint and it was very slow going. The net result of the weekend was that we drank all of the beer and painted about one third of Don's house. Don spent the rest of the summer painting himself. You might remember that it was Don Estridge who later became a Vice President of IBM and moved IBM into the pc era in the early 1980s by establishing agreements with Bill Gates. This made Microsoft.

System/360 Experiences

In the early 1960s, a new computer family design was being developed within IBM. Actually there were two parallel projects developing alternative solutions. One group including Fred Brooks and Gene Amdahl having experienced the development of the Stretch and Harvest computer systems went on to develop a proposal related to a family of compatible computers with different bus widths, memory sizes and performances. This became the basis for the System/360 series. Another group was working on a solution that would be based upon lifting the level of instruction set implementation to permit convenient implementations of higher-level languages. Inspiration for this approach came from a competitor; namely the Language Directed Architecture efforts under the direction of Robert Barton and the Burroughs Corporation in producing the Burroughs 5000 and later 5500 and 6500. This was dubbed, I believe, to become the IBM 8000 series (following the second generation 7000 machines).

This was a turning point in the computer industry. I am certain that had the 8000 series been selected instead of the Sytem/360, the world of computing would be different today.

The Instruction set architecture of the System/360 was implemented in various hardware configurations via the usage of microprograms. The microprograms provided for achieving compatibility of various members of the System/360 family. At the same time the hardware implementations of various System/360 series members utilized, for the first time, integrated circuits. The instruction set architecture while consistent amongst System/360 models turned out to be quite difficult for the various compilers in generating effective code. This had an important impact upon the operating system OS/360 and related systems software as described later.

PL/I and the Pointer Variable

In an effort to unify programming languages, IBM together with is user groups Share and Guide defined, by committee, a new programming language. Originally called NPL it became PL/I. It was to be used to replace Fortran, Cobol and even Algol. The language became quite large even though there were many good concepts and principles that were used in its development. The compilers for various System/360 models where assigned to groups in Hursley, England and in Boeblingen, Germany. It was also planned that the SLANG compiler-compiler would be used to provide an implementation.

In 1964, after the implementations were started, I was asked to join the design control board for the PL/I language. The language had been defined but it did not contain any facilities for treating heterogeneous linked lists. This was an important omission since it was envisioned that PL/I would be used in implementing graphics and system software. In particular, one large IBM customer; namely General Motors Research Labs wanted to write graphics driver programs in PL/I and one of the compiler implementations was to use PL/I to develop (bootstrap) the compiler.

Due to my earlier published works on linked lists for handling multi-linked data structures, I was assigned the task of developing a proper facility for PL/I. It is at this time that the pointer variable concept was invented and then, after much hard work, integrated into PL/I. George Radin, who led the development of PL/I, pointed out in an article concerning the History of PL/I that the pointer variable turned out to be one of the most utilized facilities of PL/I. To document the pointer variable concepts, I wrote an article published in the June, 1967 issue of the CACM. The paper entitled: ***"PL/I List Processing"*** is also referred to in Donald Knuth's Volume 1 on Fundamental Algorithms.

While adding the pointer variable to the already large PL/I programming language was accomplished, had the facilities been defined earlier and integrated, the complexity of the PL/I language could have been radically reduced; especially in the area of storage classes.

I pointed this out in an internal publication, but since the implementations where in full swing, there was no opportunity to back out. On the other hand, some subsets of PL/I where the pointer variable became a central facility were implemented by IBM as a systems programming language and even by microcomputer manufacturers in the language PL/M.

Reflections

It is unquestionable that the pointer variable has had a significant impact upon programmers around the world. When used properly, it provides a vital capability to handle complex data structures. When used improperly is can lead to bugs that are hard to detect.

One of the most successful uses of PL/I for systems programming was its utilization in the Multics Operating System at MIT, most likely the best OS ever developed. Since the introduction of the pointer variable concept, variants have been implemented and used widely in such popular languages as C, C++, Pascal, Ada and even Cobol and Fortran.

That significant attention has been given to the “pointer variable” is indicated in a Google search nowadays yielding many millions of hits. It would be hard to envision implementing systems software and advanced applications these days without these facilities. In summary, this invention has certainly proved to be one of the more significant contributions to the field of computing.

In year 2000 I was honored as a Charles Babbage Computer Pioneer by the IEEE Computer Society for this invention that has had a long lasting impact upon the computer industry.

Operating System/360

While working on PL/I, I was asked to represent PL/I compiler efforts in a control board established to deal with standards (including calling sequences and parameter passing) for the system programs to be provided with OS/360. This was both enlightening and frightening.

Fred Brooks has documented the problems with OS/360 in his famous book “The Mythical Man-Month”. I saw this tragedy develop. Prior to the large-scale systems programming effort at IBM there was a small group that included George Mealy that put forth a consistent set of concepts and principles that were to be followed in implementing OS/360 and the related systems software. As the systems programming efforts, including compilers got into full swing, the problem of complex code generation named earlier became evident. So, many systems programming projects requested exceptions in order to generate more effective code. The control board was established to review such requests. But as a result the complexity of OS/360 starting from a single quite readable design notebook evolved under a period of about 6 months into a wall filled with documentation that nobody read. Chaos was a fact.

Reflections

My experiences with System/360, PL/I, the Pointer Variable and OS/360 provided a lasting impression on my mind concerning complexity. I have often called this period the beginning of the march into the “black hole of complexity”. This certainly has led me to always value the ability to establish a small set of driving concepts and principles for everything I have “architected” since then.

An interesting observation concerning System/360 that leaves many questions unanswered is the fact that in the 1970s the US Government gave the go ahead to IBM to license System/360 to the Soviet Union. One can question why the government gave this permission to export prime computer systems technology to its cold war enemy? The Soviet Union established the ES EVM (ЕС ЭВМ, Единая система электронных вычислительных машин, meaning “Unified System of Electronic Computers”) project and a variety of hardware implementations were done in various countries. Naturally, they received OS/360 as a part of this agreement. It turns out that they were never able to manage to deal with the complexity and, although over 15000 mainframes were produced, the costs where enormous. Eventually these very expensive efforts where abandoned with the net results of most likely setting Soviet computer technology back at least a decade. Again, one can question if this was the intended plan from the US government side? It is interesting to note that Soviet computing, like the alternative 8000 series at IBM mentioned above, was earlier following an architectural approach that would simplify software development by using higher-level approaches to the hardware architecture.

Computer Architecture and Microprogramming

After experiences with compilers and programming languages, in 1965, my interests turned to microprogramming and computer architecture. I led a research group at IBM's New York Programming Center that studied the utilization of microprogramming in implementing programming languages. My colleagues included Lou Levine, Bob Flynn, George Mandler and George Awad. As microprogramming became a more established approach in hardware, there developed interest in higher-level language oriented instruction set implementations along the lines developed by Burroughs. While our research was interesting it was difficult to have an impact upon IBM that had already marched into the black hole of complexity with System/360 and was not interested in backing out.

So, in 1967, I decided to leave IBM to pursue a career in academia at the Polytechnic Institute of Brooklyn (described below). However, my interests in microprogramming continued to evolve and I became a founding member of SIGMICRO. I worked with Sam Husson and others including the famous computer pioneer Professor Maurice Wilkes who first introduced the concepts of microprogramming.

Standard Computer Corporation

In 1969, I took a leave of absence from Brooklyn Polytechnic to join Standard Computer Corporation in Newport Beach, CA. Standard had already developed microprogrammable computers. There I was one of the designers (along with Dave Keefer and Burton Smith) of a unique flexible central processing unit called the MLP-900. Due to a change in corporate direction, only one prototype of this advanced machine was built. It was delivered to the Rand Corporation and then moved to the University of Southern California Information Science Institute. At ISI, it was further developed and integrated into the original ARPA network as a computer architecture research resource that was utilized up to the late 1970s.

Reflections

The development of the MLP-900 aroused the interest of Alan Kay who visited us in Newport Beach. He was developing a language he called FLEX and saw the advantages of implementing a FLEX machine via microcode. Of course, his efforts later led to the Smalltalk programming language.

The MLP-900 went onto the ARPA network and was placed in a dual configuration with a PDP-10. In addition to being able to emulate the PDP-10 it could be used to download microcode implementations of other machines and, for example, ANYUK (military real-time computers) were emulated remotely. This certainly is a unique aspect of the ARPA network.

Datasaab

The MLP-900 aroused the interest of several computer manufacturers including ICL in England and Datasaab in Sweden where it was planned to license the design. This did not happen, but I came to Sweden in 1971 as a consultant and deigned a new processor called the FCPU (Flexible Central Processing Unit). This processor utilized a hardware implementation of Hoare's semaphore variables in order to provide an asynchronous structure for synchronizing independent hardware modules. It may be one of the first processors to use the idea *local synchronous – global asynchronous* units that nowadays are key properties of low power hardware. The properties of the FCPU where reported in a number of publications including a contribution at the Second International Symposium on Computer Architecture in 1975.

The paper entitled: ***"The Advantages of Structured Hardware"*** was selected for the "best paper award". There was significant interest in the computer architecture community. The FCPU was utilized in the Datasaab D23 Computer System series. However, its planned true potential for more directly implementing higher level languages was not exploited. There were about nine FCPUs produced and several D23s delivered to customers. However, the microprocessor era entered in the mid-1970s and changed hardware economics. It is interesting to note that the published prior-art design features of the FCPU have been cited in two patent infringement cases in which I have been involved as a consultant.

Reflections

The leaders at Datasaab: Gunnar Lindström, Bengt Asker and Wigo Wentzel were very supportive in establishing the environment in which the development took place. Bengt Malm, Häkan Niska, Torbjörn Granberg, Bengt Magnhagen, Rolf Flisberg, Lars Blomberg and others contributed to the design and implementation effort. An important reflection of this experience was related to working with a small group of qualified professionals. We were able to design and implement this advanced central processing unit with a small staff and in a short period of time. Since the processor was divided into asynchronous units, unit testing could be neatly performed without timing constraints and then the units where integrated where the semaphore registers provided for controlled execution. I have often reflected on this design and development experience with small groups and can mention that I think this is why Sweden, as a small country of about 9 million, can develop advanced technological systems in many areas including cars, trucks, telecommunication, etc.

Academic Career

As mentioned above, in 1967, I left IBM and accepted an Associate Professor position in the Department of Electrical Engineering at Brooklyn Polytechnic. The contact with Brooklyn Polytechnic was promoted by Bob Flynn who worked in my IBM research group. He was completing his PhD in Mathematics at the time. At Poly, I had the responsibility of initiating a computer-engineering program within the department for which a number of new undergraduate and graduate courses were created. Further, I designed a Load and Go compiler for a subset of PL/I called PLAGO and led a group of students including Stan Habib, David Doucette, Aron Kirshenbaum and Len Shustek that implemented the compiler. Several of my former students from Brooklyn Polytechnic entered the computer industry, particularly at Digital Equipment Corporation, and some went on to pursue graduate education. In particular, I am proud to have provided guidance to Len Shustek who completed his PhD at Stanford, contributed to very successful new startups in Silicon Valley and is now Chairman of the Board of the Computer History Museum as mentioned earlier.

In order to provide a useful introduction to computing based upon PL/I, I teamed up with a former IBM colleague from Vienna; namely Dr. Erich Neuhold and wrote a book entitled: **The PL/I Machine – An Introduction to Programming**, published by Addison-Wesley.

During the time I was at Standard Computer Corporation (1969-70), I was a part time guest professor at the University of California, Irvine where I contributed to their graduate program and advised some students including Larry Rowe, a very successful professor at Berkeley.

After my consulting activities at Datasaab terminated in 1973, I became a guest professor at Linkoping University in Sweden. There in addition to providing graduate courses, I assisted in defining a new computer engineering degree program (the first in Sweden).

During the next six years I worked as a consultant and part-time professor. In addition to Linkoping, I had short term appointments at the University of Stuttgart, Universidad Politecnica de Barcelona and the Royal Technical University in Stockholm.

In 1979 I was selected for a permanent professor's chair of Computer Systems at Linkoping University. Interestingly, when installed in the presence of Carl Gustaf, King of Sweden, Grace Hopper participated as she was selected for an honorary doctorate. In addition to innovative educational activities, I led a research group in exploring asynchronous circuit technology. I left Linkoping University in 1986 in order to pursue full time consulting activities working from my new base in Stockholm.

Reflections on Phase 1

My experiences in the computer industry and academia yielded a great appreciation for the relationship of hardware and software (especially system software). Observing the mismatch between the Instruction Set Architecture of the System/360 and the needs of the systems software provided the impetus to seek better solutions. Of course, even though I pointed to these during my last appointment at IBM, there was no return. The march into the black hole of complexity was impossible to stop.

Well, I tried in several processor designs; namely the MLP-900 and the Datasaab Flexible Central Processing Unit to provide the basis for improving the hardware/software relationship. They were interesting architectures and aroused significant interest in the computer architecture community, but by the mid-1970s, the microprocessor showed up and change hardware economics.

While the integrated circuit advances of the microprocessor where outstanding, the early microprocessors could only implement a simple instruction set. Eventually, this led to the X86 architecture of Intel as the dominant microprocessor. Building simple applications on top of the X86 architecture (or others such as the Motorola 6800) was quite OK. However, today we are building the world's most complicated software systems on top of the X86 base that was never designed for this goal. A complete mismatch between the hardware and software leading to extremely complex and not well understood software systems. The net result being: bugs that are hard to find, viruses, as well as opportunities for hackers to exploit the complexity.

I have written several articles to point out these problems including one calling for the Rebirth of the Computer Industry as a viewpoint article in the Communications of the ACM. However, it may take a major crash of the internet or something equivalent before society drives the industry into taking fundamental root cause corrective measures that will include a better hardware/software relationship.

Phase 2 (1974-1996) - Computer-Based Systems

From the mid-1970s, I shifted focus and concentrated on advanced applications of computer technology resulting in a variety of successful systems. This has included high-voltage power dispatching for ENHER of Barcelona, the automatic train control system (still used) by the Swedish Railways, and a limited slip-coupling device (now used in virtually all four wheel drive vehicles) for Haldex. In addition, I assisted the Swedish Defence Materiel Administration in establishing and verifying safety requirements for embedded systems.

High Voltage Power Dispatching

Working with a former graduate student from the Polytechnic Institute of Brooklyn, Miquel Bertran, I developed a structured approach to implementing the complex software necessary to provide the control room function for the high-voltage power supply system of ENHER of Barcelona. In this case, we further developed the idea of monitors (invented by C.A.R. Hoare) and applied them as a fundamental means of structuring the software. Due to this structuring the software system could be developed and maintained by a very small staff. I also contributed to the development of a formal means of specifying and generating program code for the human-machine interface.

Automatic Train Control

During the years 1976-77, I was engaged as a consultant in developing the architecture for the implementation of the world's first microprocessor based Automatic Train Control system. This on-board system for signal interlocking and speed control was developed by Standard Radio of Sweden (owned by ITT) and utilized in the Swedish railways (SJ).

This work resulted in a highly simplified and robust architectural solution that has since 1980 been utilized successfully on most all locomotives in Sweden. Via an engineering approach to the software, based upon viewing the system as being continuous time-driven (as opposed to discrete event-driven), the solution resulted in a surprisingly small amount of software code. The first version occupied just over 10K bytes of read-only memory. Even as successive versions have been implemented the code size is extremely small. The current system that significantly extends the system functionality occupies about 27K bytes of read-only memory.

During the first thirteen years of its existence in virtually all locomotives in Sweden not a single line of program code was changed. Probably a world's record! It has successively been upgraded to new versions for faster trains. The system has been implemented for railways in Norway, Finland, Malaysia and Australia. The Standard Radio developed ATC system solution is now owned by Ansaldo of Italy. Ansaldo utilized the architectural concepts in the implementation of the New Jersey Transit Automatic Train Protection system. The key to providing this small but powerful solution (about 30K bytes of ROM memory in the latest Swedish version) was viewing the system as being continuous in time instead of being discrete. This engineering view paid high dividends and has resulted in an important safety facility for trains.

At Standard Radio, I worked with Sivert Wallin and Berit Brintse who later continued to further develop the technology in a company called Teknogram. In addition to providing continued support to Ansaldo, they used the same architecture to generate train operator training simulators for a variety of customers.

The success of the Automatic Train Control system architecture stimulated personal research related to the generalization of the approach in building real-time control systems that later was exploited in the following projects.

Distributed Control Systems for Vehicles

The architectural concepts developed in the Automatic Train Control project and my further development of the concepts and principles were inputs to a research project initiated by the Swedish National Board of Technical Development (NUTEK). I was a member of the research team that further developed the concepts and principles of the continuous time-driven solution I had introduced. The solution resulted in accommodating both continuous and discrete functions in a distributed vehicle control context. In this effort these two approaches became known as the Red and Blue application functions. (This identification is now used internationally in discussing automotive distributed control networks). As a part of the research work, I introduced the concept of Software Circuits as an abstraction for developing program logic. It is based upon viewing software via a hardware-oriented paradigm where short well-defined program sequences are provided to perform circuit like transformations of input variables to output variables. This, once again radically reduces software complexity.

I worked closely with Kurt-Lennart Lundbäck of Arcticus Systems in this project. He integrated many of the concepts and principles of deterministic solutions into a real-time operating system called Rubus. Further, his company has developed an approach for model-based design utilizing the concept of Software Circuits.

Limited Slip Coupling Device

Along with Arcticus, I became involved in an innovative project, led by Anders Cederberg, to develop a coupling device for vehicles for Haldex in Landskrona, Sweden. Haldex had purchased a patent related to controlling the degree of stiffness between the front and back axels of vehicles. Then they received an order from Volkswagen to develop a computer-based control system to implement this function. Rubus was used as the basis for controlling this combined mechanical, electronic, and software based product. At the time, I had learned of the ISO/IEC 12207 for developing processes related to software systems. I tailored a version of this so that it could be applied not only to software but to the other technologies as well. (Note: later I participated in the development of ISO/IEC 15288 that provides processes for all types of man-made systems.)

It took some time for Haldex to achieve a successful product, but now this device is used by virtually all automotive manufacturers around the world and provides a very useful safety and efficiency function for four-wheel drive vehicles.

Swedish Defence Materiel Administration (FMV)

In 1986 I resigned from my professor's position at Linköping and moved to Stockholm where I founded my own consulting company, Lawson Konsult AB. I was contracted by the Technical Director of FMV, Ingemar Carlsson for about half of my consulting time. Based upon experiences with safety critical real-time systems, I developed an approach to assessing safety properties in real-time systems for the Swedish Defence Materiel Administration. In addition to this project, I have been involved in assessing the potential of new computer architectures as well as in assisting in the mid-life update of the Viggen jet fighter plane developed by Saab. This latter assignment involved producing documentation of the computer control system since the original solution did not provide sufficient information for those involved in updating the control system. This is the backside of small projects done by a small group of competent engineers. The knowledge lies mostly in their heads and often leads to maintenance problems.

Technical Committee on the Engineering of Computer-Based Systems

I have made significant professional contributions related to spreading knowledge on the Systems Engineering of Computer-Based systems. This has been accomplished by publications as well as active participation in organizations in Sweden and internationally including the IEEE Computer Society Technical Committee on the Engineering of Computer-Based Systems (ECBS) and INCOSE. I was a founder of the Technical Committee on the Engineering of Computer-Based Systems and served as its chairman (1999-2000).

Reflections

My experiences with embedded real-time systems from this phase convinced me that many commercially available event-driven real-time operating systems add significant complexity and non-determinism into applications. There are better solutions for a variety of embedded system applications. The experience with ATC (Automatic Train Control) was a significant development that then was exploited in the automotive context and in other commercial products from Arcticus Systems.

During the Nutek sponsored research project we had cooperation with Professor Herman Koptez at the Technical University of Vienna. Herman shared the view concerning time-driven deterministic solutions.

Academic Career (Continued)

As noted above, during the period 1974 to 1979, in addition to active consulting at ENHER and with ITT I was both a part time and guest professor at Linköpings University, at the University of Stuttgart, at Politecnica de Barcelona, and at the Royal Technical University in Stockholm (KTH).

In 1979 I was appointed as a full time Professor of Computer Systems at Linköping University. There, in addition to both undergraduate and graduate education efforts, I led research efforts in the area of asynchronous circuits as well as software engineering.

I can mention that during 1984 I had a sabbatical leave from Linköping and was guest professor at the University of Malaya. During this time, I together with the Dean of Engineering Tengku Azzman and with the support of the Prime Minister Dr. Mahathir established the Malaysian Institute of Microelectronic Systems (MIMOS). The goal was to improve knowledge and capabilities in electronics and software engineering so that Malaysia could lift itself up on the value chain instead of simply being a provider of inexpensive assembly labor for microelectronic companies. Today, MIMOS is a government owned company employing over 600 people and has placed Malaysia as a leading developer and user of ICT technologies. One month in 1984 was also spent as a guest professor at Keio University in Yokohama, Japan where I continued a project with Kee Yamamoto, a former Linköpings University guest researcher.

During my visit in Malaysia I came upon a new idea of providing graduate education for foreign students in Sweden. Thus upon returning, I successively developed proposals of how to provide a program that would attract foreign students and faculty in providing Masters Degree education in English. The proposal attracted significant attention of the University Chancellor and many of the leading universities in Sweden. So, I took a leave of absence in 1985-86 and further developed the concept that became known as Swedish International University. It was a virtual university developed as a catalogue and presented courses from participating institutions; namely, Uppsala, Lund, Royal Technical University, Chalmers, Linköping, Agricultural College, Stockholm School of Economics (Handelshögskolan) and Lulea. I had significant support from the Swedish Employer Organization (SAF) and the concept aroused significant interest. Unfortunately, the funds necessary to sustain the concept where never provided. All organizations, including the Minister of Education thought it was a great idea, but no champion arose to provide the necessary funding. It is interesting to note that today the ERASMUS program provides some similar possibilities.

Phase 3 (1996 – Present) – Complex Systems

With a solid background in the computer industry as well as experiences in designing and developing several successful embedded systems, I was well prepared to enter this third phase where my focus has widened to the general area of Complex Systems. It is certainly true that hardware and software systems always exist in some wider system context from which requirements are provided. So, this was a natural step in my career.

International Standards

I joined the ISO/IEC JTC1 SC7 WG7 effort aimed at developing an international standard for System Life Cycle Processes at its beginning in 1996. My participation as Head of the Swedish delegation was at the request of Dr. Raghu Singh, at that time convener of WG7 and was sponsored by the Swedish Defence Materiel Administration and the Swedish National Board for Technical Development (NUTEK).

After various practical difficulties in this project, a restart was made in 1999. At that time, Stan Magee became the new convener and Stuart Arnold was appointed as the sole editor of the emerging ISO/IEC 15288 standard. Further in an election involving the fourteen member countries, I was elected to be the architect of this standard. From this point, Stuart Arnold and I worked very closely in pinning down the concepts and principles upon which ISO/IEC 15288 is based. Further, we worked on spreading the concepts in WG7 and elsewhere and defended them during the following years that led to its initial publication in 2002.

The experience in WG7 offered the opportunity both to learn about standards and the standardization process as well as to work on applying the standards of WG7. After learning about the ISO/IEC 12207 standard on Software Life Cycle Processes, I helped three companies in tailoring the standard to meet their needs for life cycle processes.

Cambio AB - A medical information system provider. The processes of 12207 were tailored as a means of defining the details of agreements between Cambio as a supplier and the hospitals and clinics to which they supply their products and services. It defined roles and responsibilities over the range of software related processes.

Gambro AB – A medical equipment supplier specializing in kidney dialysis equipment. The tailoring of 12207 was required for Gambro to attain Federal Drug Administration approval of its software development life cycle. This was successful and the FDA reviewer at that time stated that it was the best set of software life-cycle processes he had experienced. It is also interesting to note that this stimulated the development of a specialized standard for software in medical devices that is based upon 12207.

Haldex AB – Supplier of a Limited Slip Coupling Device for four-wheel drive vehicles. As described above, the concepts and principles of the train control application were used in this product. In addition, the 12207 standard was tailored in order to deal with the system as well as software aspects. The system being composed of mechanical hardware, electronic hardware, and software elements. (This was prior to the existence of 15288).

While working with 12207 and during the earlier developments related to 15288, I cooperated with Johan Bendz at the Swedish Defence Materiel Administration (FMV) on how these standards could be of assistance in implementing Change Management. Johan also joined into the SC7 standardization activities and has had a convener role in the ISO/IEC 42010 standard on Architecture Description.

Since the 15288 standard can be applied to any type of hard and/or soft man-made system, Stuart Arnold and I decided to write a joint paper in order to convey the message of a changing scope for systems engineering. It points to the fact that the standard can and is showing to have an impact upon business management as well as systems engineering.

The Swedish Defence Materiel Administration (FMV) was the first organization in the world to implement ISO/IEC 15288. I served as a mentor to the implementation project.

Consulting Partner at Syntell

In 1999 I started my cooperation with a Logistics and Systems Engineering company in Stockholm called Syntell AB. We have co-operated in spreading the knowledge of systems engineering and in particular that ISO/IEC 15288 standard in Sweden and internationally. I have worked closely with Tom Strandberg and others. The managing director of Syntell Mats Bjorkeroth also asked my to join the board of directors and I served in this role for five years. I have helped Syntell open new markets for their services, in particular in Russia where there has been increasing interest in applying the 15288 standard. Dr. Asmus Pandikow has taken over the managing director role and we continue to cooperate in joint ventures.

Continued Academic Connection

As mentioned above, I had a parallel academic career, beginning in 1967, when I joined the Polytechnic Institute of Brooklyn as an Associate Professor in the

Electrical Engineering Department. I have continually had contact with a variety of universities in the USA, Europe and the Far East.

In the early 1990s I assisted the Mechanical Engineering department led by Sören Andersson at the Royal Institute of Technology (KTH) in making the transition to Mechatronics. I held a graduate course, advised students and helped them establish contact with Professor Herman Koptez and the University of Vienna. I also assisted graduate students in their research including the current Professor of Embedded Control Systems, Martin Torngren.

In 2002, Professor Dinesh Verma requested that I become an Academic Fellow at Stevens Institute of Technology and to develop a new graduate course on Systems Thinking. In response to this request, I developed and have delivered the academic course at Stevens as well as at other universities and as professional development training in organizations in various countries.

As the course material matured, successive versions where produced and in 2010 I published a book entitled: **A Journey Through the Systems Landscape** as a Print-on-Demand book published by College Publications of Kings College, UK. The book is aimed at developing a discipline independent capability to think as well as to act in terms of systems. This is accomplished by unifying concepts of principles of systems thinking and systems engineering.

The course can and has been taught for scientific, engineering, military, medical and management related participants with a wide diversity of backgrounds. The course develops a common conceptual view that can be applied for any type of man-made system (hard, soft, or mixtures thereof). The study of the dynamics of systems thinking is based upon work of Peter Senge, Peter Checkland, and others. Through systems thinking, it is possible to identify system problems and opportunities. After identification of problems and/or opportunities, the acting part is accomplished by prudent decision-making and identifying required structural changes in systems. The acting part is presented in the context of the life-cycle management of systems based, for example, upon the ISO/IEC 15288 standard. Coupling the thinking and acting together is a paradigm for change management.

Some of My Publications

I have published over 100 papers at various conferences, in professional journals, as technical reports. Further, I have provided several contributed chapters and published books. Here I present those that I personally consider to be the most important.

- "The Use of Chain List Matrices for the Analysis of COBOL Data Structures", Proceedings of the ACM Conference 1962, Syracuse, New York.

  (First publication of multi-linked data structure algorithms as cited by Donald Knuth in his classic book The Art of Computer Programming, Volume 1: Fundamental Algorithms)
- "PL/I List Processing", Communications of the ACM, Volume 10, Number 6, June 1967.

  (First publication of the pointer variable concepts)

- “Programming Language Oriented Instruction Streams”, IEEE Transactions on Computers, Volume C-17, Number 5, May 1968.

  (An early presentation of the relationship between programming languages and processor instruction set architectures)
- “Functional Characteristics of a Multi-Lingual Processor”, IEEE Transactions on Computers, Volume C-20, July 1971. (Co-author: Burton K. Smith).

  (Description of the first truly general-purpose microprogrammable processor, the MLP-900 that was an ARPA Net resource in the early 1970s)
- “Advantages of Structured Hardware”, Proceedings of the Second Annual Symposium on Computer Architecture, Houston, Texas, January 1975. (Co-author: Dr. Bengt Magnhagen).

  (Description of the Datasaab FCPU (Flexible Central Processing Unit) that utilized asynchronous control of the microarchitecture - Selected as the Best Paper Award at the Symposium)
- “Function Distribution in Computer System Architectures”, Invited paper appearing in the Proceedings of the Third Annual Symposium on Computer Architecture, Clearwater, Florida, January 1976.

  (Describes the importance of taking a holistic view of computer architectures in respect to the distribution of functions between hardware and software levels)
- “Philosophies for Engineering Computer Based Systems”, IEEE Computer, Vol. 23, No. 12, pp. 52-63, December, 1990.

  (Discusses the importance of identifying and utilizing driving concepts and principles in developing and sustaining complex computer-based systems and includes 3 case studies)
- “CY-CLONE - An Approach to the Engineering of Resource Adequate Cyclic Real-Time Systems”, Real Time Systems, The International Journal of Time-Critical Computing Systems, Vol. 4, No. 1, February, 1992.

  (Based upon experiences with the Automatic Train Control system, this extends the use of a deterministic approach in the context of multi-processor systems – Can be used as a basis to provide safety critical solutions using multi-core microprocessors)
- Parallel Processing in Industrial Real-Time Applications, Prentice-Hall series on “Innovative Technologies”, ISBN 0-13-654518-1. 1992.

  (The result of a sponsored project to introduce engineers in Sweden to the utilization of parallel and distributed processing in real-time applications)
- “BASEMENT: A Distributed Real-Time Architecture for Vehicle Applications”, Real Time Systems, The International Journal of Time-Critical Computing Systems, Vol. 11, No. 3, November, 1996. (Co-authors: H. Hansson, M. Strömberg, and S. Larsson).

  (Describes the project involved in developing a distributed architecture for vehicles and includes the introduction of the concept of “software circuits”)

- “Salvation from System Complexity”, IEEE Computer, Vol. 31, No. 2, Feb 1998, pp 118-120.
- “Infrastructure Risk Reduction”, Communications of the ACM, Vol. 40, No. 6, June 1998, pp120.
- “From Busyware to Stableware”, IEEE Computer, Vol. 31, No. 10, Oct 1998, pp117-119.

  (All three of these brief contributions point to the problems that have arisen as the result of using the limited capability microprocessors from the 1970’s in today’s complex computer system applications)
- “Rebirth of the Computer Industry”. Communications fo the ACM June 2002/Vol. 45, No. 6.

  (Describes in detail the problems caused by the transition to microprocessor based solutions in respect to the complexities that have evolved resulting in unreliable software solutions. The article points to a path for rectifying these problems within the computer industry. Unfortunately, this has not happened and probably will not happen in my lifetime – We need some real catastrophe’s to initiate such a transition)
- “Twenty Years of Safe Train Control in Sweden”, Proceedings of the International Symposium and Workshop on Systems Engineering of Computer Based Systems, Washington, DC. April, 2001. (Co-authors: S. Wallin, B. Bryntse, and B. Friman).

  (Documents the result of this very successful Automatic Train Control system, the world’s first microprocessor based solution)
- “Viewing Systems from a Business Management Perspective: The ISO/IEC 15288 Standard”, The Journal of Systems Engineering, Vol. 7, No. 3, September, 2004. Co-author: Stuart Arnold

  (Describes the change role of Systems Engineering from traditional product orientation towards the needs of organizations and enterprises)
- “A Journey Through the Systems Landscape”, College Publications, Kings College, UK, ISBN 978-1-84890-010-3. 2010.

  (A book that introduces the use of Systems Thinking, Change Management and Systems Engineering as proper approaches to dealing with complex systems. Provides a number of useful concepts, principles and paradigms aimed at providing personal as well as group (including organization) competence)

Final Reflections

Well – it has been a long and interesting career. Certainly it has been non-deterministic since life for me has been a happening with changes in my place of work, the orientation of my work, the countries I have worked in and the family I have shared my life with. I have learned a lot and have dedicated my final years to sharing the knowledge I have accumulated. But it is never too late to learn so in parallel I keep learning from colleagues, my consulting activities and participants in my academic and professional development courses.

Information Systems Degrees in Australia: The Genesis

Audra Lukaitis, Stasys Lukaitis, and Bill Davey

School of Business IT and Logistics, RMIT University, Melbourne, Australia

Abstract. This paper traces the birth of the Information Systems degrees offered by RMIT University in Melbourne, Australia. The paper argues that university curriculum in new areas is a reflection of forces both social and individual. The case of Information Systems curriculum is distinctly different from Computer Science both in the extent of these influences and in the nature of the relationships between the University and industry in general. The analysis presented here is a social history. It does not concentrate on pivotal moments of invention so much as the people and influences that culminated in programs that persist until today.

Keywords: information systems curriculum, information systems history, information systems.

1 Context: Australia, An Early Adopter

The environment in which this study commences is of broad experimentation with computers becoming widespread in both government and industry. A significant dearth of qualified personnel in the computing industry brought experts from all over the world to a vigorously growing Australia. People from this burgeoning area were also recruited into universities at a very early stage.

At the Royal Melbourne Institute of Technology (now RMIT University) Brian O'Donahue (from the Commonwealth Public Service) was recruited to head the Data Processing Group. People like Cliff Forrester, who had become engaged by computers after working on EDSAC (Electronic Delay Storage Automatic Calculator) in England as a result of his aeronautical jobs at Farnborough, drifted from some part time teaching at Caulfield Institute (one of the first Colleges to offer formal degrees in Information Systems) into a full time role at Royal Melbourne Institute of Technology (RMIT).

These industry experienced academics reflected the early entry of the country into computing. Australia has a relatively long history of computing. The CSIR Mk1 (CSIRAC – Council for Scientific and Industrial Research Automatic Computer), built in the late 1940s was Australia's first internally-stored-program computer and is acknowledged to be the world's fourth (Tatnall & Davey, 2004).

1.1 RMIT University Evolution

RMIT is quite old in Australian terms. The University opened for business on 7th June 1887, as the Working Men's College with an initial enrolment of 600 students.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 89–107, 2012.

In 1934 the Working Men's College was incorporated and its name change to 'The Melbourne Technical College' and in 1960 the 'Royal Melbourne Institute of Technology' was adopted as the new name by the council. Over the life of the institution the various names included:

- RMIT University (1992-)
- Royal Melbourne Institute of Technology (1960 – 1991)
- Melbourne Technical College (1934-1959)
- Working Men's College (1887-1933)

Merged institutes and colleges incorporated into RMIT University include:

- Phillip Institute of Technology
- Preston Institute of Technology
- State College of Victoria (including Coburg Teacher's College)
- Emily McPherson College (and College of Domestic Economy)
- Melbourne College of Printing and Graphic Arts
- Melbourne Institute of Textiles

By 1988 The College offered classes in technical, business and arts subjects with an emphasis on applied skills relevant to trades including architectural and mechanical drawing, theoretical and applied mechanics, plumbing, carpentry and painting, as well as studies in arithmetic, algebra, bookkeeping, shorthand and physics.

Currently (2012) there are more than 75,000 enrolled students (with 19,600 Vocational Education and Training students) including 28,000 international students representing more than 100 countries. The heritage of starting as a union supported college for working men has seen the University orientate itself in such a way where curriculum and applied research are heavily influenced by current industrial practice.

The clear industrial orientation and interest in applied rather than pure research provides an interesting beginning and backdrop to the history of the creation of information systems programs. The story of curriculum development presented in this narrative involves academics who were recruited to the University because of their industrial experience rather than their traditional academic experiences such as research publications and grants success.

2 Research Method

The themes that form this paper were derived from text analysis and analysis of interview transcripts. To achieve some validity in the process the three researchers produced independent analyses from a subset of the sources used. These were then compared, and an iterative process of refinement used, to result in the themes presented here, sources included:

- **Documents:** This history was constructed from original documents such as brochures, Faculty of Business handbooks, course guides and internal University documents, Australian State and Federal Government education reports, published policies and papers.
- **Interviews:** Face to face interviews were also conducted with Cliff Forrester, Audra Lukaitis and Stasys (Stas) Lukaitis which were recorded.

Cliff Forrester was employed by the University after leaving Caulfield Institute in 1965 to help teach in the courses set up by Brian O'Donoghue as part of the then accounting degree. Audra Lukaitis and Stasys Lukaitis worked for different parts of the University, and were brought together by the formation of the new Department of Business Information Systems on the centenary of RMIT's opening in 1987 and were founding members of the new Business Information Systems Department. This means that they were able to observe the evolution of existing courses into degree programs in information systems first hand.

3 Australian Government Education Policy

Higher education policy, dictated by the national government, is the first major influence on the environment. The aims and missions of higher education institutions such as RMIT University and the birth of information systems education must be seen in the context of this environment. State and federal government higher education policy has influenced the mission and direction of higher education institutions throughout Australia particularly in the post-World War II era.

How government has responded to population growth, economic boom times and recessions, labour market trends, upswings and downturns, university overcrowding, increasing demand for higher education, the financing of higher education, internationalisation of education provision and other various economic or political imperatives through higher education policy is critical to understanding most higher education institutions in Australia.

Where originally higher education was mostly funded by the state government, changes in funding mechanisms from state to the commonwealth government meant that control shifted away exclusively from state control to national control, from institutional autonomy to national government accountability. As demand for higher education increased, and traditional labour markets disappeared, the higher education system moved from an elite to a 'massified' system (today also about 50% self-funded).

3.1 The Murray Report (1957)

In post-World War II Australia, The Murray Report (1957) prompted far reaching changes to Australian universities "*from largely undergraduate institutions to modern research universities*" (Lindsay, 1987). The Murray report is also significant in that it informed much of the higher education policy and debate for almost the next forty years. In his investigations, Murray found severe overcrowding in universities, initiating expansions in the sector via Commonwealth funding matching state funding.

The key outcomes of the Murray Report included the establishment of the Australian Universities Commission (AUC) in 1958, the Commonwealth Government helped fund universities for recurrent and capital purposes, the funding of scholarships particularly for teacher training was introduced, and Commonwealth funding of postgraduate education and research.

In his report Murray stressed the roles and functions of universities and technical colleges – the technical college is to produce technicians and craftsmen (of which there was an urgent national need), the function of universities is professional training including research, which should promote national interests and is vital to the academic health of staff and students.

In the 1960s the economic, demographic and social expansion started in the 1950s continued. Demand for university places and severe overcrowding, highlighted by Murray almost a decade earlier, increased. According to Meek, "*political and social pressures to further expand higher education intensified*" (1991:465).

3.2 The Martin Report (1964-1965)

The Martin Report recommended the creation of Colleges of Advanced Education (CAEs) rather than the expansion of universities, thus creating a new binary system for higher education. The CAE was "*based on a new kind of institution, more applied and less research-oriented than universities*" (Karmel, 1988:121). Martin reiterated and reaffirmed boundaries between academic and vocational education. Meek (1991:466) notes that despite the recommendation of the Martin Report the binary system transformed itself into a 'trinary' system in just over ten years with colleges of Technical And Further Education (TAFE) taking up apprenticeship training and sub-diploma courses. RMIT's new status as a College of Advanced Education was reflected in a name change to the Royal Melbourne Institute of Technology which it held from 1960-1991.

3.3 The 1970s

In 1974 the Whitlam Labor government initiated further major changes in higher education. Tuition fees were abolished and the generous Tertiary Education Assistance Scheme (TEAS) was introduced. Moreover, all recurrent and capital funding of universities and colleges of advanced education was taken over completely by the commonwealth government from the states. This meant that tertiary education could now be controlled nationally, policies would be set nationally, and funding would be controlled at the national level. According to Harman (1989:4) the federal government had now become the "*dominant partner in policy determination and planning for higher education*"

After the dismissal of the Whitlam Government after denial of 'supply', Prime Minister, Malcolm Fraser's so called 'razor gang' tried to respond to increasing political and public pressure for funding constraints across the board nationally in the context of a global recession, industry shut downs and massive redundancies in the workforce across Australia. A climate of accountability and cost cutting prevailed, as the new agenda of restraint in government spending set in, particularly in education, the modus operandi became how patterns of post-secondary education satisfactorily matched labour markets.

3.4 The 1980s

The Malcolm Fraser Liberal government specified amalgamation rounds of targeted colleges of advanced education (CAEs) linking these to federal funding. One third of CAEs were told to amalgamate, and by 1983 all but four of the CAEs had amalgamated. In such a climate of economic restraint issues of quality and efficiency moved to the top of the higher education agenda. How could more be done with less?

By the 1980s continuing changes in the global economy and other factors – for example the Bob Hawke/Paul Keating Labor government 'level playing field' policy of reducing tariffs, particularly in manufacturing and textile industries – saw a decline in Australia's manufacturing base and an erosion of traditional overseas markets, increasing global competition, resulting in the decline or disappearance of many skilled and unskilled jobs and industries, while other traditional skills were changing with the rise of technology in the workplace. Under this government, state sector industries were dismantled, the financial system deregulated and the Commonwealth Bank of Australia sold off.

In higher education, further major rationalisations were carried out through amalgamations. There was a rapid growth in advanced education and by the mid-1980s CAE students outnumbered university students. Until the late 1980s the commonwealth continued to provide 90% of higher education funding.

By the early 1980s Australia was again in recession, and by the end of the term of the Fraser government, unemployment topped 10%. With government cutbacks in higher education funding, CAEs and universities began to seek ways to raise revenue and make up government funding shortfalls through full-fee short courses, international student revenue and initiating partnerships running off-shore accredited education programs. CAE academics in particular were encouraged to become involved in consultancy work and full-fee overseas students were enrolled in greater numbers across campuses in Australia. As demand for tertiary education continued following on from increasing school retention rates, Australia mirrored other comparable OECD countries in the provision of tertiary education from an elite to mass system (the next transition being a 'universal' provision).

3.5 Dawkin's Green Paper 1987

In late 1987-1988 further influential and far reaching changes to higher education were proposed and ratified by Dawkins in the so called '*Green Paper* 'of 1987 (Higher Education: a policy discussion paper) and ratified by the '*White Paper*' of 1988 (Higher Education: a policy statement). The report noted that industry had increased its demand for postgraduates in the areas of management, business and commerce, proposing that education and training would play a *central role* in responding to Australia's major economic challenges. Dawkins abolished the Commonwealth Tertiary Education Commission introduced by the Fraser government and formed the National Board for Employment, Education and Training.

Perhaps the most dramatic of Dawkins' reforms (under the Hawke/Keating Labor government) concerned the further amalgamations of many universities and CAEs throughout Australia, effected through funding allocations based on institutional profiles and students numbers, giving rise to large multi-campus universities and CAE institutions across Australia, via the so-called 'Unified National System'.

The distinction between colleges of advanced education and universities was removed, in effect granting former colleges of advanced education university status. The catch cry of Hawke/Keating government was that they were setting up conditions to foster a 'clever country'. In effect this led to today's 'corporatisation' of universities.

Dawkins further introduced the Higher Education Contribution Scheme (HECS). Through this scheme students paid university fees via the Australian tax system, either upfront receiving a discount or later through deductions from their wages. There was a fixed charge for various programs with arts and humanities at the cheaper range of the spectrum, computer science and information technology in the middle spectrum, and medicine, dentistry and law at the most expensive end.

With a change in government, Amanda Vanstone (under the Howard liberal coalition government) increased HECS charges to students, and reduced grants to universities.

Simon Marginson (2001) in analysing trends in the funding of Australian Higher Education and concludes:

> *"The Dawkins reforms remade higher education as a competitive system of self-managing institutions with control over their own resources, while subject to accountability requirements and limits on numbers in relation to government-funded places. The underlying objectives of the shift to mixed funding were to provide fiscal relief for the government, and to strengthen economic relationships between universities and industry so that higher education would contribute to national competitiveness. The first objective was successfully achieved. The HECS was introduced in 1989 at an average 20 per cent of course costs; later, in several stages, the level and rate of repayment were increased. The DETYA data also record a rapid increase in incomes from international student fees, vocational postgraduate fees, especially in Business Studies, and continuing education ... The number of international students grew from 21,112 in 1989 to 83,111 in 1999 ....*
>
> *By 1998, 33.2 per cent of all income received by higher education institutions was derived from the HECS plus university-determined fees and charges, compared to about 2 per cent derived from fees and charges in 1983. In 1998 income from international students constituted 8.3 per cent of all income, and more than one dollar in five in institutions with greatest exposure to the market. Whereas governments provided 90 per cent of funding in 1983 and 70.3 per cent in 1989, by 1998 the public share was down to 51.9 per cent ... Though most OECD countries saw increases in private funding during this period (Williams 1992), the Australian change was remarkable in its speed and universality."*

These influences are critical in understanding the development and growth of RMIT University and the school as it is today. Autonomy, direction, funding and control have shifted from the school to the university as the university now relies heavily on international tuition fees, general tuition fees and other fund raising activities in order to operate in the face of diminishing government fiscal support and heavy reporting and assurance activities.

4 Computing before the Inception of Information Systems

The history of computing at RMIT commences in 1962 with the lease of an Elliot 803 computer (Tatnall, 2006). In 1986 the Faculty of Business (subsequently the Business Portfolio and now College of Business) was composed of the Department of Accountancy, the Department of Administrative Studies and the Department of Applied Economics.

Noel Anthony was appointed Foundation Dean of the Faculty of Business in 1977 and encouraged the Faculty of Business Computing group to acquire and deploy a new HP250 minicomputer, funding the setup of the first terminal labs in the Business Faculty. Noel Anthony retired in 1986, with the new Dean Dr. J. Milton-Smith appointed 1986, commencing in 1987.

In the first instance the Department of Accountancy employed some people from industry to deal with the problem of students, particularly rising numbers of postgraduate students, wanting to be able to design, manage, oversight, implement or participate in the introduction of data processing equipment and systems into their organisations.

Eventually there were enough people to create a data processing group and for a leader of that group to be employed. The first leader was Cliff Forrester who became so disenchanted with organisational administrative tasks that he applied for leave to undertake a Masters in Computing at the University of Texas. His replacement, Brian O'Donaghue, took the data processing group, now considerably enlarged because of the demand for courses, into the newly created Department of Administrative Studies.

The group provided significant service teaching throughout the Faculty of Business as well as offering streams into some degrees. Eventually the Department of Administrative Studies offered graduate diploma degrees, starting first with computing programing: the Graduate Diploma in Commercial Data Processing. This timeline compares with the creation of the RMIT Department of Computer Science in 1981 (Tatnall, 2006). In 1984 the information systems group was created in the Department of Administrative Studies under the leadership of Tony Adams (replacing George Sutherland), co-opted from the Department of Computer Science. By the end of 1984 the data processing group consisted of Tony Adams (Principal Lecturer), Neville Stern and Nigel Thomas (Senior Lecturers), and lecturers Cliff Forrester, Hugh Ballantyne, Philip Crutch and Stasys Lukaitis (Tatnall, 2006).

5 1988 – The Department Is Created

In 1988 the Faculty of Business under Noel Anthony saw the demand for information systems courses to be too large (with demand growing) for a group within the Administrative Studies department. The Department of Business Information Systems was created under the leadership of Tony Adams. Tony had previously worked in the Department of Computer Science at RMIT, and before that had come from industry where he was the Data Processing Manager for Monash University in Melbourne.

It was also in 1988 that Stas Lukaitis was tasked with the replacement of the HP250 minicomputer and a new technology DISC symmetric multiprocessor UNIX minicomputer was acquired.

The first Windows Personal Computers were being deployed along with early Apple Macintosh computers.

5.1 1988 – Secretarial Studies Fights Back

By the mid-1970s and 1980s Australian university places were created and mostly funded by the federal government. This funding arrangement was especially important for the formation of undergraduate degrees at that time. A new degree could only be created by taking student places away from other programs (or courses as they were called then).

RMIT had very successfully run secretarial business studies degrees and graduate diplomas in the 1960s and 1970s and had an outstanding reputation in this area. The first secretarial business studies degree was launched in 1976, and 1979 saw the first students graduate in the three year Bachelor of Business in Secretarial Studies course where students studied sociology or psychology, general accounting, macro-economics, commercial law – contracts, company law, principles of finance, marketing, data processing, labour relations in addition to secretarial studies and supervised professional practice.

According to the secretarial studies group, these graduates readily found employment in secretarial/administrative positions and many had multiple job offers. The secretarial undergraduate degree brochure recognised that technological change in the office was a challenge to management to understand and seek out the advantages of technology for their organisations and personnel; they were challenged as to whether they would "*let it happen*" or "*make it happen*" emphasising that RMIT is producing people "*to make it happen*".

In the late 1970s the University hierarchy attempted to close down the secretarial studies degrees, wanting to release the funding for other purposes. RMIT had underestimated the 'fight back' response of the secretarial studies group who galvanised a large number of alumni, many of whom worked for powerful people in powerful places, including leading business leaders, politicians, and members of Parliament. There was a large 'penultimate' city protest/demonstration with news coverage.

The secretarial group consisted mostly of a small number of older (not too far off from retirement) women who had given good service to the Tech (as RMIT was fondly referred to), students and the community. Their educational standards were rigorous and their group was run along 'traditional' lines, from another era. The women were well educated and had various backgrounds including experience in industry. There was a strong hierarchy, a very strong commitment to 'standards' from which the group did not swerve; there were many students who would never graduate as they could not meet specified required minimum skill standards (which were actually quite 'high' by today's standards). In their way they kept up with technology (for example they had championed and overseen the installation of memory typewriters, then a dedicated word processing laboratory stocked with a small number of IBM dedicated word processors and a laboratory of dictaphone machines and electric typewriters, however, were finally stumped by the introduction of the personal computer (PC) and the new software. The Tech had to ‘back down’ from the total shutdown of the secretarial group, reviewing their position regarding redundancies.

5.2 The Impact of the PC on Secretarial Studies

There was much happening during the 1980s in respect of the impact of personal computers and changes to work processes. Fine old business traditions such as typing pools, comptometrists and ledger machine operators were being slowly replaced with electric typewriters, memory typewriters, word processing centres, dedicated word processing machines then the ubiquitous personal computers with their cheap and user-friendly word processors; calculators had arrived and instead of ledger cards we saw the arrival of departmental computing and database systems.

The secretarial staff mostly had accounting qualifications, two staff members had extensive industry experience but no university qualifications. One particular staff member, Eileen Gueho, was also very active in the Australian Institute of Management, and other business organisations. She also had a large network of important industry contacts that were beneficial to courses, subjects, staff, students and the RMIT all round.

Eileen Gueho was also a cooperative education pioneer, the pathfinder who sourced, negotiated, coordinated and supervised industry job placements for students’ cooperative education year (one year of work in industry) drawing on her vast network of contacts and alumni to place students in jobs. She was also a champion public speaker winning competitions in national toastmaster events; she had originally trained as a concert pianist and had worked in industry.

Needless to say, in the context of a seeming upheaval in how office tasks were now being accomplished in industry, particularly the larger global institutions and the rapid changes in technology and the encroachment of personal computers, there were some tensions between the traditionalists and the change agents who could see what was happening in industry.

5.3 New Bachelor of Business – Office Systems/Business Information Systems Streams

With RMIT’s re-evaluation of its position due to publicity and unexpected fight back response, the old secretarial undergraduate degree was incorporated into a new hybrid Bachelor of Business degree offering two streams: the Bachelor of Business (Information Systems) and (Office Systems).

Therefore the new Department was built on the secretarial studies government funding. The funding was split 50/50 between two streams: Office Systems and Business Information Systems (each stream consisting of 8 full semester units).

It is important to remember that running an undergraduate degree was determined by the Government funding allocated to the University and divided amongst the competing Departments. These changes were championed by Tony Adams, who lead the dramatic changes and instigated the new undergraduate business information systems degree, incorporating the secretarial group into the office systems streams. Much of the curriculum and course philosophy of the office systems stream was founded on the research and writings of Rudi Hirscheim and Enid Mumford, and from the beginning were founded on a human factors perspective.

In the office systems program, explanations of the trend towards automating office functions and the need to educate personnel to meet the demands of this evolving business *function* to "*analyse office activities, plan for and design appropriate office systems, implement new office technologies; evaluate and manage the new office systems*" was made explicit. However, to differentiate themselves from information systems the program emphasised the human side of the office "*remembering that all offices depend on people, how they work and interact and people cannot yet be automated. Thus the education of office employees for all levels is viewed with technology, communication and management all given equal focus in the curriculum.*" (Department of Business Computing postgraduate programs brochure 1995:8)

5.4 New Department Built on Large Compulsory Core Subject Enrolment

Equally important, the school was also built on the large compulsory enrolment of all Faculty of Business students into the business computing foundation subject 'Computer Applications in Business'. At that time over 1,300 students were enrolled in this subject. The whole Department taught into this subject, including the Head and the Principal Lecturer both on campus and overseas, and every staff member contributed content, direction and teaching, including lectures and laboratory sessions. Curriculum included word processing, spreadsheets, database, expert systems and was for a time 'bilingual' – teaching software on Microsoft Windows PCs and Apple Macintosh computers. A series of supporting lectures expounded on the latest developments in information systems and technology, raising issues about security and ethics from the very beginning. In fact, Tony Adams, the founding Head of Department, pioneered the Australian Computer Abuse Research Bureau (ACARB), which was visionary for its time. It was also at this time that the Faculty of Business started to experience a leap in enrolments. Demand was burgeoning across all business programs and courses.

5.5 A Fresh Paradigm of Working with New Technology in Business

This introductory computing subject was always considered with some importance to the Department, as it was the cornerstone and portal to other information systems and business computing subjects offered, also it was an important source of university funding. It was a core subject of all Faculty of Business degrees. Incidentally the content and style of the subject was built on industry consultancies delivered by RMIT's Australian Microcomputers Industry Clearing House (AMIC) – a consulting arm of the University staffed by department academics (who also were required to consult in industry) and other computer specialists. It was through this organisational group that RMIT and the Department were able to generate income to fund resources, software and equipment.

The basis of this subject was that it addressed the impact of the (then) new technology Personal Computer and its associated personal productivity software and how working in business would change forever. The term 'paperless office' was also coined about that time.

Although generally the software and other concepts have changed with natural developments and in computing in business, the framework is essentially the same as it was 25 years ago. Like the London underground map, the introductory computing subject map put together by the data processing and office systems groups has stood the test of time.

One of the reasons for its success is its elegance and simplicity addressing the way people would think and work with this new technology in a business context.

5.6 1988 – The first Bachelor of Business (Business Information Systems) Degree and Growth in Postgraduate Diploma Enrolments

By 1988 the new Bachelor of Business (Business Information Systems) consisted of two major streams: Business Information Systems and Office Systems. The new program extended over four years (of which there was a one year work placement component). The office systems stream looked at the human factors in and around the implementation of information systems, whereas the business information systems specialised in the technical design and implementation of information systems.

Each stream also offered a one year full-time, or two-year part-time, graduate diploma (a postgraduate qualification) consisting of eight subjects. The jobs the Office Systems postgraduate course targeted for their graduates were Systems Training Officer, Market Planning Manager, Information Manager, Personnel Officer, Projects Coordinator, Change Management Consultant, Office Technology Consultant, Database Systems Officer, International Marketing Specialist and such like. Business Information Systems targeted jobs such as Systems Analyst, Database Programmer/Analyst/Administrator and information technology consultant and advisor.

The Postgraduate Diploma in Secretarial Studies continued (with large numbers of students still applying), though the program was to be rebadged and restructured as the Graduate Diploma in Business Systems reflecting its predominantly business curriculum of accounting, marketing, law and management..

The Postgraduate Diploma in Commercial Data Processing also still continued (with large numbers of students applying and growing).

5.7 Close Ties with Industry

The Department and its academic staff had arisen from demands of industry for job specific training. This was (and is) reflected in working conditions and priorities. Until recently PhD qualifications were seen as irrelevant as the role of the College of Advanced Education (CAE) was "*more applied and less research-oriented than universities*" (Karmel, 1988:121). The Martin Report tabled in 1964-65 recommended the creation of Colleges of Advanced Education rather than the expansion of universities to cope with the demand for more universities places. The mission of the college of advanced education was therefore very different from that of a university.

Rather than research, RMIT CAE academics were required to teach and consult into industry and maintain close links with industry. In her interview Audra Lukaitis describes this priority:

> *"It was a very heavy load to work in industry as a consultant and then work full-time as an academic. Nevertheless this experience was to prove absolutely invaluable to me and has informed my whole approach to computing education. To industry I was able to bring fresh ideas, concepts and theories garnered from research in my fields, and from industry I was able to bring back real world case studies that informed all assignments, approaches and curriculum content. I was able to see firsthand what really went on in industry as a participant and bring it back to the classroom, enlivening the student educational experience."*

In an interview, Stasys Lukaitis remarks:

> *"It might be of interest to note that AMIC – Australian Microcomputer Industry Clearinghouse - was created by an enterprising Tony Adams and others in response to the 'sudden' arrival of microcomputers onto the scene, much like the recent 'cloud and mobile computing' phenomena. The demand for short courses and consultancies on the use and deployment of micros, their software e.g. – Visicalc, Lotus, MultiMate, WordStar, Dbase was huge and AMIC blossomed. Several BIS staff were recruited from the ranks of the industry experienced consultants who worked at AMIC (e.g. Peter Viola who was a Manager at AMIC for many years).*
>
> *Academics who worked at AMIC were actually paid at standard industry consulting rates. It was this that created issues with the Chancellery who were concerned that academics were being paid too much. That was the start of the demise of AMIC, the birth of the MDC[1] and the end of academic input into University industry education engagement."*

Thus in the past under the leadership of Tony Adams and Eileen Gueho and the academic staff at the time there was considerable engagement with industry almost on a day-to-day basis. There was a prevailing atmosphere of excitement and anticipation.

---

[1] MDC: The Management Development Centre – the University's official window to business and industrial education. The MDC was more structured and bureaucratised and the role and influence of academics somewhat reduced.

5.8 Overseas Demand

Right from the formation of the Department of Business Information Systems (now School of Business Information Technology and Logistics) the founding Head, Tony Adams along with others in the Faculty began to pioneer and establish international education initiatives onshore and offshore, namely in Malaysia, Singapore and Hong Kong; particularly with Barry Cooper from Accountancy and James Hurley from Administrative Studies who set up the original Malaysian offshore business degree at Taylors College. Hugh Ballantyne was also instrumental in promoting the Department's courses overseas at education fairs and such like.

In retrospect it can be seen that this was partly a response to changes in government higher education policy and early decline in government funding, and in parallel the new demands for 'research' and computer equipment which had to be funded.

Staff were now required to teach overseas (in addition to consulting in industry), which certainly added another challenge, but without a doubt yet another valuable dimension. Groups of students from around the world appeared in classrooms in increasingly large numbers. This brought many new challenges for which many staff were unprepared - academic standards, assumptions, English language proficiency, writing and comprehension, extremes in diversity, cultural differences - all had to be dealt with. The international student market was grown very aggressively and successfully by Barry Cooper from Accountancy, Colin Bent from Economics and Finance and Tony Adams from Business Information Systems (onshore enrolments and offshore partnerships). They were initially the ones who had the vision to expand RMIT's activities globally; they were the pathfinders, the global thinkers and doers. Once again, teaching overseas was considered to be 'above load', so academics were paid separately for overseas teaching while still putting in full time work back home, nevertheless all academics were required to teach overseas.

In hindsight it is clear why RMIT Business, and Business IT in particular, were so successful in recruiting overseas students and in expanding offshore education. The answer lies in the type and quality of the academic staff who engaged with the students. Staff almost entirely deeply understood and were experienced in industry and business and enthusiastically transferred that quintessentially Australian educational experience to the overseas students. As one student remarked in a survey at the time "*the most interesting thing about this course is listening to the war stories and experiences from the real world*".

5.9 Heads of School – The Champions

It is useful to view the evolution of the School of Business Information Technology in terms of the Heads of Schools who have contributed so much to its continued success and continuing relevance in terms of the context of the times and developments in communication and information systems technologies, also in the context of evolving national government higher education policies and RMIT University's response to changing times.

Table 1. RMIT Department of Business Information Systems Timeline

Period	Head	Events
1984-1990	Tony Adams	Information Systems Group formed inside the Department of Administrative Studies, amalgamated with Secretarial Studies and eventually becomes an Independent Department of Business Computing. The PC and Macintosh become ubiquitous, local area networking starts becoming popular and desktop software becomes accessible and user friendly. AMIC and external consulting extensive. Staff appointments from experienced industry sources. Global education initiatives. Ever increasing demand for business information systems education, particularly at the postgraduate level.
1990-1993	Neville Stern	Adams becomes Acting Dean of Business, while Stern becomes acting head. Turmoil in Australian academia as Universities forced to amalgamate by government as cost cutting measure. RMIT's proposed amalgamation with Victoria University aborted, Philip Institute (PIT) amalgamation alternative proposed. The Internet is born. More industry sourced staff appointments. AMIC ceases and the MDC is born, industry consulting at its peak. Student demand continues.
1993-2001	Ken Millar	PIT amalgamation proceeds and PIT's Ken Millar becomes Head of Department. Department moves from LaTrobe Street campus into central business district. Amalgamation with Department of Information Management and Library Studies. Forced amalgamation with VET (Vocational Education and Training) sector computing group. Staff appointments now from both industry as well as career academics (research and publications experience alone). MDC demise, industry consulting diminishes and is eventually discouraged in favour of research and publications. Various attempts to create a Faculty of Information Technology amalgamating Computer Science and Information Systems, fail. BIS Department creates business oriented networking subjects.
2001-2002	Tom Yardley	Yardley appointed as acting Head upon Millar's retirement. Ongoing failed attempts to create a Faculty of Information Technology.
2002-2003	Kevin Adams	Adams appointed as acting Head from Department of Accountancy. Staff appointments are now only academic with no further industry experience considered appropriate. The demise of the postgraduate programs Information Systems and Office Systems, all postgraduate programs amalgamated into the Master of Business Information Technology. Prerequisite of prior industry experience dropped.

Table 1. (*continued*)

Period	Head	Events
2003-2005	Carolyn Dowling	Career academic Dowling appointed from Australian Catholic University as Head. Subsequent staff appointments now only career academics. VET sector amalgamation abandoned. University policy phases out postgraduate diplomas (8 courses) and slims down 16 course Master degrees to 12 courses. All graduate diplomas in the school amalgamated into the one Masters of Business Information Technology.
2005-2005	Barry McIntyre	McIntyre from the original Information Management and Library Studies Department appointed acting head.
2006-2010	Brian Corbitt	Career academic Corbitt variously from Thailand, University of Melbourne and Deakin University appointed as Head of the now School of Business Information Technology. All new staff appointments now young career academics with little or no industry experience. Research, publications and grants are now top priority. Corbitt instrumental in developing a stronger research and innovation culture. All staff encouraged to do more research and publish in academic journals and attend conferences or undertake or finish off higher degrees. Faculty of Business renamed as College of Business.
2010-2012	Caroline Chan	Chan from Deakin appointed as Head of the now renamed and re-tasked School of Business IT and Logistics. Logistics group of a dozen career academics absorbed into the School. Centralised administrative and strategic functions strip course and program development and strategic industry direction out of the School into College Office. No Information Systems representation left at the strategic college level.

6 Conclusion

This paper has presented the creation of Information Systems programs form the viewpoint of the significant forces that shaped events.

The demise of secretarial studies and the degree programs that emerged with the creation of the information systems department seem illogical. Managers continue to have executive assistants and support staff who in addition to a sound business education in accountancy, economics and finance, marketing, business law, human resources and management need to have particular additional skills in using technology and an understanding of information systems. The argument used against programs for these types of people seems to have been based upon a belief that secretarial studies consisted of training in technology that had been completely replaced by the computer. This argument concentrates on the replacement of the typewriter and shorthand with the word processor and dictaphones.

Another outcome may be reflected in the nature of information systems curriculum. At RMIT Information Technology based courses took two paths right from the beginning. Computer science was developed from the machine, its design, and programs that could run on it. Computer Science was based in the disciplines of mathematics and applied physics and electronics and, as such, could be seen as a valid integration of those two areas. The Computer Science Department arose from the Department of Mathematics and Computer Science, originally just Mathematics. It is of interest that most Computer Science people originated from the maths and science disciplines where it was based. Much of the curriculum was scientific and mathematical with scant regard paid to business and ‘commercial data processing’. Computer Science degrees in the 1970s studied compiler design, the solutions of differential equations, operations research, calculus, artificial intelligence and a form of software engineering. This, of course, was to be expected given that the academic staff of Computer Science was either scientists or mathematicians.

On the other hand Information Systems and Office Systems arose from the demand of industry to provide education and training for staff required to understand and implement new technologies in business. In the late 1960s and early 1970s there was a plethora of courses aimed at producing workers in business capable of playing a part in the data processing of the business. The early programs offered by the data processing group and by the Information Systems Department were heavily skewed towards postgraduate education. Vast numbers of students enrolled in postgraduate programs part-time so as to gain the business-oriented computing skills and education needed in their new jobs.

It is also of interest that it was the office systems group that introduced a human factors perspective in the design, implementation and management of information systems education at RMIT. In hindsight, the basis for this is quite obvious – it was in fact the office systems group who had the most extensive business contacts and experience with professional engagement.

It is no coincidence that the Accounting Department was an early sponsor of these courses and the creation of the data processing group. The Accounting Department would be the first to understand the changes happening in business due to the introduction of computers in business data processing.

When the first head of the Information Systems Group, Tony Adams came from Computer Science he may have been tempted to import some computer science flavour into the courses. His recruitment record clearly indicated that he understood the difference between computer science and information systems. He hired senior people from IBM, HP and other businesses who would understand the needs of the graduates of these courses. This meant that the courses developed by the Information Systems Group were heavily biased towards current business practice in industry. The programming languages were Cobol and forms of business oriented Basic rather than FORTRAN, C and Pascal. Systems subjects were prominent in many subject names and used the words data processing.

The use of business-oriented databases was pre-eminent and was the foundation of much curricula. A key difference between the Computer Science view of databases and that of Information Systems was that IS was concerned with modelling the business and its processes rather than the calculus and mathematical aspects of the database.

With the foundation of the business information systems degree incorporating office systems curricula a stronger ‘human factors’ and sociotechnical systems design dimension was added to business computing studies in and around human computer interactions, end-user systems, useability and accessibility of human systems, the impact of computerisation on work processes, management and change management.

Data communications was taught from the business perspective of connectivity and ubiquitous data access rather than queuing theory and sliding window protocols.

Today’s Information Systems is the result of blending the best from the modernised Office Systems program which was concerned about how a business operates effectively and from the old data processing that was concerned with the modern business and the technology-enablement of its processes and operations. This outcome is the result of a 25 year old vision of Tony Adams who foresaw the importance of academia and industry working together with new technologies to solve tomorrow’s business problems. Tony Adams also had the experience and vision to understand the impact of computing on work processes and that the ‘human factor’ was of vital consideration in the design and implementation of computer systems in business. Both Tony Adams and Eileen Gueho understood the importance of industry engagement very early on, seeing a vital link between theory and practice, industry engagement is today RMIT’s enduring brand.

Their vision was able to be brought into being because the University at that time encouraged leadership, innovation and creativity to flourish at a local level. Adams and Gueho recruited and employed talented individuals with extensive industry understanding and experience. Individuals who had deep links with industry and who nurtured and grew them. The Dean at that time was Noel Anthony who trusted his staff with the responsibility of keeping RMIT abreast and indeed ahead of those heady days.

This paper recounts the history of a single university’s information systems degrees. It would be interesting to look at the history of other information systems programs to see if similar backgrounds have produced an environment similar to that of the current information systems programs at RMIT.

References

Dawkins, J.: Higher Education: A Discussion Paper. AGPS, Canberra (1987)

Dawkins, J.: Higher Education: A Policy Statement. AGPS, Canberra (1988)

Harman, G.: The Dawkins reconstruction of Australian higher education. Paper Presented at the 1989 Annual Meeting of the American Education Research Association, San Francisco, March 27-31 (1989)

Karmel, P.: The Role of Central Government in Higher Education. Higher Education Quarterly 42(2), 119–133 (1988)

Lindsay, A.: Doctoral Education: Continuity & Change. Research & Development in Higher Education 9, 361–372 (1987)

Marginson, S.: Education and Public Policy in Australia. Cambridge University Press (1993)

Martin, L.: Report of the Committee on the Future of Tertiary Education in Australia (the Martin Report), vol. 3, Australian Universities Commission. AGPS, Canberra (1964-1965)
Meek, V.L., O'Neill, A.: The transformation of Australian higher education from binary to unitary system. Higher Education 21, 461–494 (1991)
Murray, K.: Report of the Committee on Australian Universities. the Murray Report. AGPS, Canberra (1957)
Tatnall, A.: Curriculum Cycles in the History of Information Systems in Australia. Heidelberg Press, Melbourne (2006)
Tatnall, A., Davey, B.: Streams in the History of Computer Education in Australia. In: Impagliazzo, J., Lee, J.A.N. (eds.) History of Computing in Education. IFIP, pp. 83–90. Kluwer Academic Publishers, Assinippi Park (2004)
RMIT, Faculty of Business Handbook 1987, Undergraduate and Postgraduate Programs, Melbourne, Australia (1987)
RMIT, Faculty of Business Handbook 1988, Undergraduate and Postgraduate Programs, Melbourne, Australia (1988)
RMIT, Faculty of Business Handbook 1989, Undergraduate and Postgraduate Programs, Melbourne, Australia (1989)
RMIT, Faculty of Business Handbook 1990, Undergraduate and Postgraduate Programs, Melbourne, Australia (1990)
RMIT, Faculty of Business, Department of Business Information Systems, Proposal for Accreditation of a New Teaching Development Bachelor of Business (Business Information Systems) (1988)
RMIT, Faculty of Business, Department of Accountancy Studies, Secretarial Studies, Bachelor of Business in Secretarial Studies Brochure (1980)
RMIT, Faculty of Business, Department of Administrative Studies, Secretarial Studies, Bachelor of Business in Secretarial Studies Brochure (1983)
RMIT, Faculty of Business, Department of Administrative Studies, Secretarial Studies, Bachelor of Business Administration (Office Systems Specialization) (1987)
RMIT, Faculty of Business, Department of Business Information Systems, Graduate Diploma in Secretarial Studies Brochure (1987)
RMIT, Faculty of Business, Department of Business Information Systems, Graduate Diploma in Office Systems Brochure (1988)
RMIT, Faculty of Business, Department of Business Information Systems, Bachelor of Business (Business Information Systems) Brochure (1988)
RMIT, Faculty of Business, Department of Business Information Systems, Graduate Diploma in Office Systems Brochure (1989)
RMIT, Faculty of Business, Department of Business Information Systems, Graduate Diploma in Office Systems Brochure (1988)
RMIT, Faculty of Business, Department of Business Information Systems and Department of Information Services, Master of Business in Information Technology Booklet (1988)
RMIT, Faculty of Business, Department of Business Information Systems, Graduate Diploma in Office Systems Brochure (1991)
RMIT, Faculty of Business, Department of Business Computing, Graduate Diploma in Applied Information Systems, Graduate Diploma in Office Systems, Master of Business in Information Technology (Business Information Systems stream) Brochure (1995)
RMIT, Faculty of Business, Department of Business Computing, Graduate Diploma in Applied Information Systems, Graduate Diploma in Office Systems, Master of Business in Information Technology (Business Information Systems stream) Brochure (1996)

RMIT, Faculty of Business, Postgraduate Studies in Business Information Systems, Department of Business Computing, Graduate Diploma in Applied Information Systems, Graduate Diploma in Business Systems, Master of Business in Information Technology (Business Information Systems stream) Brochure (1996)

RMIT, Faculty of Business, Postgraduate Studies in Business Information Systems, School of Business Information Technology, Graduate Diploma in Applied Information Systems, Graduate Diploma in Office Systems, Master of Business in Information Technology (Business Information Systems stream) Brochure (2000)

RMIT, Faculty of Business, Postgraduate Studies in Business Information Systems, Department of Business Computing, Graduate Diploma in Applied Information Systems, Graduate Diploma in Office Systems, Master of Business in Information Technology (Business Information Systems stream) Brochure (2002)

Looking Back

Martha E. (Hinds) Crosby

Department of Information and Computer Sciences, University of Hawaii
marthaecrosby@gmail.com

Abstract. In this chapter, I am reminiscing on some of my experiences from my 53 years of living with computers. I am currently a professor and chair in the department of Information and Computer Sciences at the University of Hawaii. Over the years, I have only experienced computing environments in three locations (Colorado, Washington, D.C. and Hawaii) but they have all been memorable. The computers that I describe are a part of computing history but are rarely documented.

Keywords: Computer science, geophysical parameters, mapping program, mathematics.

I was not aware of computing before I went to college. I majored in mathematics and I had a professor who believed that computers would be the wave of the future so he gave some short courses on machine language and numerical analysis techniques. Little did I realize that this would make me a qualified applicant for a computing career. I graduated with a Bachelor of Science degree from Colorado State University (CSU) in 1959. At that time, calculators were mechanical and very large and heavy and no one would consider owning one. Also, very few commercial digital computers could be found in the United States and the ones that existed usually occupied a room and were primarily for calculations. I planned to teach mathematics in high school after graduation from CSU. However, I finished my classes a quarter early so teaching positions would be at least a half year away. I was offered positions at Martin Marietta in Denver, Colorado and the National Bureau of Standards (NBS) Central Radio Propagation Laboratories (CRPL) in Boulder, Colorado. Since my job at Martin Marietta would have primarily been data entry and CRPL had a variety of interesting projects that seemed to require some mathematics, I accepted what I thought would be a temporary position in a government research laboratory.

CRPL was responsible for producing monthly maps that represented characteristics of the ionosphere that might cause problems in radio communication. Methods developed for creating these maps by hand or with calculators were tedious and frequently required keeping track of extensive tables. In addition, creation of the maps was time consuming and could take as much as six months to complete. The need for the use of computers to produce world-wide mapping methods had been felt for many years. Roger M. Gallet, a CRPL physicist had obtained a grant for that purpose and had been working with William B. Jones, a CRPL mathematician, to develop numerical methods that would represent the world wide complex variations of

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 108–114, 2012.

ionospheric characteristics as a function of latitude, longitude and time over a 24 hour period. Development of this method was influenced by: statistics and information theory; functional approximation and orthogonal series for representation of the real physical variations; inclusion of the empirical knowledge accumulated from ionospheric research; and the necessity that the method could be adapted to the available digital computers. The method developed was general enough that it could apply to any world-wide geophysical parameters that vary continuously with time. After a few months, I postponed my plans to teach mathematics because I was offered an opportunity to continue the work of Anne Hessing, a CRPL mathematician-programmer, who was moving to California. Anne had begun programming the 'mapping problem' methods on the IBM 650, a decimal computer leased by CRPL. Other features of the IBM 650 included an extra address in the instruction that specified the location of the next instruction. This meant that it could take the entire rotational speed longer for the instructions to be programmed sequentially than if they were optimized to correspond to the rotational speed (approximately 6 milliseconds per revolution) of the drum. The amount of storage in the CRPL IBM 650 consisted of 2,000 10bit (+sign) words. It also had floating point capability; this meant it had a mantissa of base 10 and the exponent was of a different base. The number representation and the vast range in the magnitude of the geophysical data required at least some knowledge of numerical analysis. The IBM 650 had vacuum tube technology that is not known for its reliability and, as a result, the computer could only be counted on to run reliably for approximately two hours at a time. Thus, it was essential to formulate a plan for storing intermediate results. Punched cards were the only medium of input and output. CRPL did not have printer but it had a plotter. The punched card intermediate results had to be organized and read back in to the computer. If the computer happened to fail during this process, the most information that would be lost was the last two hours of intermediate results. Once the coefficients for each map were completed, the contour maps could be drawn on the plotter. The size of the code for the mapping program, with its many subroutines, far exceeded the capacity of the IBM 650. It was only by using this scheme of partitioning that it was possible to produce the results. The process took approximately a week to make a map. The good news was that the map from the numerical methods represented the ionosphere almost as well as the corresponding map made by manual methods. The numerical methods developed by William B. Jones to represent the ionosphere may have been the first computer-based use of numerical mapping. This was a major breakthrough because it meant that computers could be used to map almost any geophysical parameters that varied continuously with time.

The time required to produce a map was reduced from months to hours, however, it became obvious that a larger computer would be needed to fulfil CRPL's requirement to produce monthly maps. The IBM704 was the largest commercial computer available at that time and there were approximately 50 of them in the United States (Bell and Newell, 1971). However, even using the one leased by Martin Marietta in Denver (which was only 19 miles from CRPL) was impractical either because of the cost or availability. Fortunately, CRPL was a laboratory of the National Bureau of Standards (NBS) in Washington, D.C. and NBS not only had an IBM704 but it had been developing computers since May 1950, when a group of employees from NBS built the 'Standards Eastern Automatic Computer' (SEAC). Also, NBS

Mathematicians had written a general purpose program for statistical and numerical analysis and had devised many algorithms that were published in journals and NBS technical reports. Although the IBM 704 still used vacuum tubes, it had 32,768 (32K) 36-bit words and other advantages such as core storage and a 2 microsecond cycle time. From 1959 through 1962, I worked refining the algorithms of the mapping program by commuting from CRPL to NBS. I wrote the programs in assembly language first on the IBM 704 and IBM 709 then on the IBM 7094 (which had transistors instead of vacuum tubes). For a complete description of the program, logic, instructions for use and additional methods of mapping see Hinds and Jones (1963). The initial mapping program took the entire capacity of the IBM 704 so I had to reserve the computer for 2-3 hours, usually during the 12:00pm-6:00am shift. The output from the NBS computer was written onto 2,400 foot reels of magnetic tape. However, the data had to be converted to the punched card format that was needed for the plotter at CRPL in Boulder. This meant I had to physically take the tapes to another government agency across town, get the several thousand cards of coefficients that represented the ionosphere and mail them to CRPL. The mapping problem continued after I left CRPL. Figure 1 shows William B. Jones and me transferring the artifacts to Ronald P. Graham (center). Despite the implementation improvements on the increasingly powerful computers, these basic numerical methods developed over 50 years ago are still used to map several time-varying geophysical parameters.

Fig. 1. October, 1962 Transfer of Mapping Program Artifacts

In 1962, I transferred to another government agency, The Harry Diamond Laboratories (HDL) from CRPL. HDL is an Army research facility that was closely affiliated with the NBS in Washington, DC. I was a hired as a research mathematician primarily to work on problems involving risk analysis and reliability. I also began taking graduate courses in statistics that were offered at HDL through the University of Maryland. I was also given the opportunity to attend national conferences such as the Spring and Fall Joint Computer Conference where I learned more about the computer science discipline. Most programs were written by scientists who worked for large corporations or the federal government. Part of my duties included writing general-purpose mathematical computer programs and making them freely available. Algorithms and their implementations were not viewed as commodities so intellectual property rights hadn't been articulated for individuals or groups of individuals. Users of the IBM 701-704 series formed a group called SHARE that helped with the exchange of software. Many of these routines that in today's environment might be called 'open source' became part of a Scientific Subroutine Package that IBM provided for IBM's 360 System of computers. The emphasis of these basic computational functions was to provide many of the tools necessary to solve commonly encountered problems that required mathematical or statistical solutions. In 1966 IBM published a manual that included the code for single or double precision mathematical and statistical applications including matrix manipulations (IBM internal document H20-205-3). Today, many of these same functions are found in commercial packages such as SPSS and SAS.

In 1970, my husband accepted a two year assignment in Hawaii (43 years later we are still here). Once it was clear that our stay in Hawaii was no longer temporary, I began looking for employment opportunities. In 1971, I went to work part time as a scientific programmer for Kentron, a branch of LTV aerospace that provided support to the Kwajalein Missile Range. Kentron used a Control Data Corporation (CDC) computer that was located in the CDC building that was approximately 10 miles from Kentron. Since Kentron employees didn't have easy access to the CDC building, we had to rely on daily courier service as we debugged our programs. This meant that the test cases had to be carefully planned in order to get as much as possible out of each test otherwise you would be delayed another day. After Kentron, I began working in the department of Oceanography at the University of Hawaii (UH) where I continued mapping other geophysical parameters. Some of my Kentron colleagues had told me that the department of Information and Computer Sciences (ICS) at UH offered a Master of Science (M.S.) degree. Although I had worked with computers for 11 years, I thought this program would be an excellent opportunity to learn more theory and be involved in the emerging discipline of computer science so I enrolled in the ICS graduate program in 1973. The ICS graduate program was initiated in 1968 as a degree in Information Sciences but by 1970 it had become a M.S. in ICS. Courses were offered in the background areas of Information Processing Machines, Logical Analysis and Probabilistic Analysis. Working with Joe Ganino, another ICS graduate student, I volunteered to debug a partially implemented BASIC/Fortran compiler on an unusual computer system developed by the Berkeley Computer Corporation (BCC) in the late 1960s. In 1971, Professor Norman Abramson was funded by DARPA to link Hawaii to the ARPANET so he and Dr W.W. Lichtenberger brought this advanced time sharing computer to Hawaii. The computer was called the BCC-500 because it could accommodate approximately 500 simultaneous users and over 2,000 terminals. This system had significantly more capability than other commercial time sharing systems that could only support 32 interactive users at the same time. The BCC-500 permitted a large number of interactive users to compute without experiencing degraded service because of the following features: critical parts of the operating system were built into the hardware; memory swapping ran concurrently with operations in the two central processing units; the communication system was an integral part of the system; high transfer rate drums provided fast swapping and access to files. The system had nanosecond capability; a speed that was accomplished by a combination of the 90bit fast memory, rotating drums and an array of one meter in diameter disks (BCC-500 System Manual, 1971). In addition to the higher level Systems Programming Language (SPL) for the BCC-500, the computer also incorporated a simulation of the SDS940. The SDS940 was originally designed as a time sharing computer for the Genie project at the University of California, Berkley. Running in the SDS940 mode, all of its software was available for further development and debugging of the BCC500 software. I had taken a short course on Multiplexed Information and Computing Service (Multics), a mainframe timesharing operating system developed at MIT in 1968 so I had some understanding of distributed computer systems.

Researchers with access to Advanced Research Projects Agency Network (ARPANET) used a process called Request for Comments to develop telecommunication network protocols. This was similar to the open standards that applied to computer software. This collaborative process of the 1960s along with research projects such as the one that follows contributed to the birth of the Internet in 1969. Dr. Norman Abramson, a UH Electrical Engineering (EE) professor led the ‘Additive Links On-line Hawaii Area’ (ALOHA) network project. It was a new form of communications architecture that provided transmitters with multiple accesses on the same digital channel. Researchers from the departments of EE and ICS had designed an experimental UHF radio, packet switched computer network. Those packets, transmitted in June, 1971, on the UH campus, between terminals in the EE building and an IBM 360/65 in the ICS building marked the first use of what is now known as an ALOHA channel (Kamins and Potter, 1998). The ALOHA system used many of the design concepts of the ARPANET but it differed from it and other networks of the time because it used a burst random access method and multiplexing between the transmitting antenna at each user station and the receiving antenna at the central station. The ALOHA system used two channels for all remote units, one for data into a central machine and the other for data out of the central machine. Data packets from all remote users access the same 24K bits/sec. radio channel in 30ms bursts; each user multiplexes their data onto that channel when it transmits the packet (Abramson, 1973). Because of the ALOHA network and the BCC-500 computer, distance education has existed at UH for over 40 years. In 1971, because of the cost, there was a moratorium on leasing computers until an implementation plan could be completed. This meant that computers were not available to Hawaii public schools.

Yet, high school students from the Hawaiian neighbour islands and other areas of Oahu used distance education to learn programming because access to the BCC-500 computer was available, free, and only required leasing a teletype machine. I worked with some high school mathematics teachers who wanted to teach programming concepts to their students. The teachers were able to convince their principals to rent the inexpensive teletypes and they were able to connect to UH remotely and their students became the primary users of this 'on-line' system by using the simulation of the Basic compiler on the BCC-500.

After earning an MS. Degree from UH in 1975, I began a non-tenure track position teaching computer sciences courses for the EE and ICS departments. At this time, it was possible to buy components such as an Intel 8080 and, with some hardware experience, put together a computer. In order to teach a class of 40 how to program microcomputers, we wrote an emulator for the 8080 on the BCC-500 that the students used it to write their programs. They had all of the debugging capabilities and speed of the time-sharing computer and the experience of writing 8080 assembly language. The slowest part of the process was transferring the paper tape output from the BCC-500 to the Intel 8080. The students really enjoyed the experience of working on a 'personal computer' but they were happy to use the emulator.

In 1979, I enrolled in the Ph.D. program in the department of Educational Psychology at UH. The goal was to gain an understanding of the difficulties many students have in understanding algorithms so my dissertation research was on program comprehension. I compared how humans process both natural and computer languages. I originally expected the process to be like reading but it turned out to be more similar to problem solving. In 1984, my advisor, Dr. Peter Dunn-Rankin received a grant from the Air Force Research Laboratories to purchase an eye-movement monitor. It took the engineering expertise of the ICS chair, Dr. W.W. Peterson to make it really work. Once I was certain that we had accurate fixation algorithms I conducted research in the areas of the human use of computing systems, individual differences of users, cognitive styles and evaluation of innovative educational environments. After completing my doctorate, I joined the Information and Computer Science department as a tenure-track professor in 1986. With backgrounds in computer science and educational psychology, I gravitated to the emerging field of Human-Computer Interaction, one of the fastest growing sub-fields within computer science at the time. I was one of the first researchers to use eye movements to investigate program comprehension, a methodology that has now become highly influential, with the greater availability of eye-tracking technology. Dr. Jan Stelovsky and I developed one of the first open-source analysis tools to visualize data collected from eye-movement monitors using the algorithms of Dr. W. Wesley Peterson (Stelovsky and Crosby, 1997). This software gave observers the ability to see the coordinates of the eye fixations super-imposed over the viewed scene, in real time. This environment facilitated more naturalistic evaluations. We have performed several cognitive psychological experiments including how people read algorithms, how they search lists, how they search large database systems, and how they view data models. Most recently, Dr. Curtis Ikehara and I have extended this work to the use of other physiological measures such as heart rate, electrodermal activity, temperature and the pressure applied to a computer mouse. It was our objective to move toward a set of passive physiological sensors that provide real-time cognitive state measures independent of task performance measures and without requiring extensive sensor calibration. These sensor systems could be used to augment or replace task performance measures in cases where task performance measures are not available. We received a patent on an ‘input devise to continuously detect biometrics’ where physiological data is examined at four critical points of a task: pre-task physiological resting state, the initial physiological response upon starting the task, the physiological response to increasing task difficulty and the physiological response at task completion. We have used these measures to determine several potential indicators of cognitive load and found them more sensitive to interaction effects with task difficulty than some task performance measures.

Most of these examples of computing environments that I have described took place as the computer field was rapidly expanding and it still is. The implementations are always changing but the theory remains steady. Instead of working with ever bigger machines, my recent work involves smaller and embedded. Whatever the size, my experience with computers has been an amazing adventure.

References

Abramson, N.: The ALOHA system. In: Abramson, N., Kuo, F. (eds.) Computer Communication Networks. Prentice Hall (1973)

Bell, C.G., Newell, A.: Computer Structures: Readings and Examples. McGraw, Hill Book Company, New York (1971)

Crosby, M., Ikehara, C.: Using real-time physiological monitoring for assessing cognitive states. In: Ghinea, G., Chen, S. (eds.) Digital Multimedia Perception and Design. Idea Group Inc., Hershey (2006)

Hinds, M., Jones, W.B.: Computer Program for ionosheric mapping by numerical methods, NBS Tech. Note 181 (1963)

Kamins, R., Potter, R.: A History of the University of Hawaii. University of Hawaii Press, Honolulu (1998)

Stelovsky, J., Crosby, M.: A WWW environment for visualizing user interactions with Java Applets. In: Smith, M., Salvendy, G., Koubek, R. (eds.) Design of Computing Systems: Social and Ergonomic Considerations, vol. 21B, pp. 755–758. Elsevier Science, Amsterdam (1997)

The Impact of the Y2K Event on the Popularity of the Pick Database Environment

Stasys Lukaitis

School of Business Information Technology RMIT Melbourne Australia
stasys.lukaitis@rmit.edu.au

Abstract. At Pick's heyday there were over 3,000 business applications available across a very wide range of hardware platforms supporting from one to thousands of real time users. The tentative economic recovery of the 90s and the Y2K fears created cautious and conservative corporate decision-making. During those tumultuous years there were startling leaps in information and communications technology rewarding those who invested in the future and in themselves. The Pick community at the time were fragmented and somewhat narrow-minded in their view of the future and were unable to collectively invest in developing new technologies. Intense marketing by 'mainstream' relational database vendors combined with ERP software vendors brought executive peer group pressure to adopt 'vanilla' relational technologies and the desire for homogeneity and perceived immunity from the impending Y2K event. A new corporate jargon was developed to further seduce executive corporate decision makers.

Keywords: Pick, Universe, Unidata, Prime, Revelation, jBase, Reality, Multivalue, Correlative, D3, SAP, Oracle.

1 Introduction

Decision making in ICT is rarely on the basis of cold hard logic or actual facts, but more often based on prejudice, comfort and expediency. Lots of decisions are sometimes made in the absence of, or despite important information.

This paper seeks to highlight an interesting historical period in the development of Information and Communications Technology in Australia and to notice the various influences of (then) emerging factors on the Information Systems that were in use at that time, and the Pick Operating Systems and Database environment as an interesting example. The period under investigation is the last twenty years of the twentieth century 1980-2000, with the emphasis on the early 1990s and the period approaching the Y2K event.

A simple Google search for the PICK operating system or database will reveal a plethora of information about the history of the Pick Operating System and DBMS over the years. It is not the intention of this paper to repeat what is already freely available. Suffice it to say that PICK was 'invented' in the USA by Richard (Dick) Pick and Don Nelson as contractors for TRW on the Cheyenne Helicopter parts and maintenance project.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 115–126, 2012.

This paper will look at the state of Information Systems that were in broad use during that period, the impact of the oncoming Y2K event, and the emergence of American and German software juggernauts Oracle and SAP and why they appeared to win the hearts, minds and wallets of enterprises across the globe.

1.1 Research Approach

This is a hermeneutical analysis of a brief period in Australian IT history using first-hand knowledge of the Pick environment and being an actor in that period as the driver. Hermeneutics is a philosophy of enquiry that seeks to gain understanding about an issue or question using techniques that attempt to deal with a researcher's biases and prejudices, and in particular the effects of historicality[1] - not taking into consideration the historical milieu and social events and thinking of the time, and the way that language and its use and interpretations can colour understanding and interpretation [1].

In this usage of hermeneutics I use original documents in the form of books, reports, magazines, articles, quotations from the industry leaders of the time, reflections from practitioners and personal experience [2]. The reason for the use of the hermeneutic philosophy is to glean understanding of the historical milieu from a variety of data sources.

This research will review the 'forgotten factors' of the time, the drivers that pushed the Australian IT industry, the key decision makers and what was happening to technology then. Historical investigations such as this should be free from the emotion and biases because, as Gadamer stated, the passage of time has allowed the events to be 'closed' [3].

The understanding and appreciation of history in its unsanitised form can be helpful in avoiding mistakes and errors already committed. It is a fact that with "*all histories they are the tales of the winners who always rewrite history to their image, leaving many stories untold*" [4].

2 A Quick Summary of the 'PICK' Concept

Unless one was versed in the idea of 'Pick', there was general confusion about what 'Pick' was. This is quite understandable because one's vision and potential comprehension is governed by knowledge, experience and a suitable vocabulary (set of paradigms) with which to understand and articulate that understanding.

Operating systems in the 1970s and 1980s were mostly proprietary and non-portable between hardware platforms. As an example, HP minicomputers were released with a proprietary operating system called MPE-IV (1980s) [5] that only worked on HP manufactured equipment. Similar examples existed with IBM and DEC (VMS).

[1] Historicality: A term coined by Hans Georg Gadamer to describe the effect of time and cultural distance between the investigator and that being investigated. An example might be the difficulty in understanding life in ancient Rome from the perspective of a 21st century New Yorker.

Open systems were emerging and one called Unix was starting to be seen available on several different hardware platforms. This portability was exciting the ICT community at the time because being tied to a particular hardware vendor was seen as undesirable for many reasons, one of which was being locked into a scalability ceiling[2]. IBM was famous at that time by offering relatively continuous scalability solutions right through to their mainframe systems.

It was quite normal for hardware vendors to also package their own proprietary operating system as a bundle resulting in the prevailing idea of an 'IBM' solution or 'HP' solution.

This section briefly outlines the Pick environment from four perspectives of a) the data model; b) the operating system; c) applications development environment; and d) the data retrieval model. Readers interesting in a more thorough explanation of this environment are referred to the bibliography for a range of material.

2.1 The Data Model

The actual database was modeled as a 'Hashed Indexed Sequential Access Method' (HSAM) mechanism. Each data record was indexed by a unique primary key that is used to calculate which frame (or bucket) is its home location inside a given fixed-size file. So for a given file and frame the read and write commands 'compiled' directly to an absolute disk, head and sector address.

Unlike other data models Pick's record structure is not predetermined by a Data Definition Language. Traditionally a database is created with the required number of tables, each table having its own peculiar structure. In Pick, one created files as needed, each one equivalent to a table. The database was then all the files that were related. Typically one would create an account called 'Student Records' and all the related files would be stored there.

Pick differs from other database models because it allows fields to have repeating values and for one field to be a 'controlling' field with others defined as 'dependent' fields. This allows alignment of repeating fields. It is also possible for any of these individual repeating fields to themselves store repeating 'sub-fields'.

Whereas people using the relational model must normalize their structures into strict two dimensional tables, the Pick model allows avoiding first normal form and the consequential join tables. Details of how Pick allowed data to be structured are well detailed by Lukaitis [6].

Thus one of the key differentiators of the Pick data model is that each 'master record' can also contain all the detail elements associated with transactions on that record. There are many examples that illustrate that the Pick model allows quite complex data structures to be represented that even today, using a strict relational model, would create very cumbersome join table proliferation and consequential management and expensive index maintenance collateral issues.

[2] Scalability ceiling: Reaching the performance limits of the hardware/operating systems platform and being unable to sensibly expand beyond that capability.

2.2 The Operating System Model

There was a stranglehold by the major vendors on their proprietary hardware and attached host operating systems. IBM, HP and DEC were but a few of the major players. It is now of historical interest the difficulties that IBM found themselves in with OS360 described by Brooks [7].

The Pick operating system was multi-user and time-sharing with the ability to run dozens of serial users on an Intel 486 computer with 512k RAM and an RS232 expansion card. Pick was an early implementer of code reentrancy which enabled efficient working set management to be implemented [8].

Perhaps from the Pick perspective, two important events were the release of the RISC engine as popularised by the Motorola M68x and IBM RS6x series chipsets and the implementation of SCSI hard disk technology which allowed very fast disk access. Pick was ported to the M68xx chipset and arrived in Australia as the Wicat computer. This was one of many ports of the operating system.

It is wise to remember, however, that in those days random access memory was very expensive and CPU speeds were only just starting to become acceptable (by those standards). The emergence of SCSI[3] technology released the hard disk bottleneck and allowed really quite fast disk operations on the smaller departmental-sized machine.

Pick's ability to leverage very fast disks, its ability to move large amounts of data quickly on and off the hard disk was its strong point. When combined with an elegant multi-user code re-entrant operating system model it was found possible to run quite large numbers of real time users simultaneously.

Of course, all operations back then were using green screen[4] technology with serial I/O along quite slow RS232 communications links.

2.3 Applications Development Environment

The applications development environment was implemented with a programming language that was tightly integrated to the host operating system and the database management system. The language syntax included very sophisticated string manipulation capabilities and dynamic arrays that mimicked and implemented the database's fundamental record structure, and internal conversion routines that allowed quite advanced date and time manipulation.

Specialised syntax allowed rapid read/write access to any files that your security level allowed. You could read an individual record or even fields within that record. Locks could be applied on a record to prevent file integrity problems surfacing in a multi-user environment to the degree where an optimistic locking strategy could be programmed to avoid a race condition and ultimate deadlock occurring.

[3] SCSI. Small Computer Systems Interface. A quite revolutionary interface design that allowed hard disks to perform at speeds normally associated with larger mainframe style channel architecture disks.

[4] Green Screen. Character oriented screens typically 80 characters by 24 rows. No graphical user interfaces or mice as pointing devices.

The language compiled to a p-code[5] that was remarkably efficient in a multi-user environment and later into native chipset executable code which increased execution speed markedly. It supported run-time relocation of subroutine code. This was keenly exploited by programmers storing the names of candidate subroutines in files that were read during program execution and loaded and run according to state conditions. Recursion was also supported, although not greatly used as it (still) is a dangerous tool in the hands of any novice programmer.

Most Pick developers were conscious of the power in the programming environment and took appropriate precautions. Nevertheless, as Dick Pick was once quoted as saying that his system was replete with features he called 'rope'. There was plenty there with which to hang yourself [9].

The only problem with this development environment and language was its name – Pick/BASIC. Most developers in the Pick community thought nothing of its name, but were constantly embroiled in bickering about the language with non-Pick developers who imagined it to be a form of Dartmouth Basic.

This unfortunate name was also responsible to large degree for its lack of acceptance, even derision by a 'sophisticated' academia who felt that anything called Basic was just that – Basic and too simplistic to even be considered. When this was compounded with a data base model that allowed non first normal form, a heresy at the time, the whole environment was simply discarded as irrelevant.

2.4 Pre SQL Environment

The query language, called ACCESS, used against a given file would be driven by the data dictionary associated with that file. In other words, using the student record example earlier, you could readily query the student file for reports or information from the target file name (student) and the list of fields (dictionary definitions) in which you were interested. This is consistent with most modern query languages such as SQL. Unlike modern day SQL, Pick ACCESS is strictly a reporting language.

3 The 1980s and 1990s Context

3.1 Database Choices

In those early days of databases there were several choices of database model. Indexed Sequential Access Model (ISAM) databases were the most popular amongst the larger machines and were used by IBM in their offerings and made 'famous' by the leading database writers of the time like James Martin [10]. The other popular database model at the time was the network model, a precursor to the relational model of today. This was popular amongst mini-computer vendors typified by Hewlett Packard and the IMAGE databases [11].

The iconic Ted Codd published a number of papers describing the two-dimensional relational database model, based on predicate logic and a relational algebra and

[5] P-Code. Pseudo code. A code that was not unlike assembler, but was interpreted at runtime by the internal Pick OS engine, not a p-code compiler.

calculus [12-14]. Numerous implementations of this relational database were spawned with some examples being DB2, Informix, Ingres, Sybase, Unify, Progress, Oracle (various hardware platforms) and later, some Open Source Unix implementations such as PostreSQL and MySQL[6].

3.2 Computer Science Departments' Impact

Universities around the world became enamoured of a DBMS that was based on a mathematical model. This was understandable as many computer science departments and their academics were born of mathematics schools [15].

Database courses were taught in Universities at the time used the popular book by Tsichritzis (Data Models) [16] that identified three main data models – the relational, network and hierarchical models and further treated on the E-R[7], Binary, Semantic and Infological models. These were the only models taught.

Only now it becomes evident that many of these computer science career academics did not have any real industrial or commercial exposure with real world databases. This lack of experience was manifested in the sorts of data models and environments that were used to demonstrate the power of the relational model.

3.3 The Drive away from Centralised DP Departments

Mainframes were becoming increasingly unpopular because of the hold by DP[8] departments on business applications development (COBOL, PL/I, RPGII, etc.) and delays and errors in delivery of systems. Anecdotal evidence at the time estimated the applications development lead time to be about 4 years. The inability of centralized DP departments to provide satisfactory service levels (sic) led to local 'Departmental' solutions leveraging off the emergence of the new mini-computers. Many of these solutions were sourced from enterprising companies who could see the need for bureaus, a service that companies or Departments could buy to solve pressing IT problems.

The same themes are emerging today with offshore development now quite popular, and the emergence of cloud computing that promises applications, infrastructure and even complex information systems solutions at the click of a mouse.

3.4 Accountants and Managers Freed from Their DP Departments

Spreadsheets[9] empowered the accounting fraternity with the ability to create complex budgets and perform 'what if' scenarios. This released accountants from the control of their DP groups and mainframe-based computer financial models and gave them the independence to plan, model and forecast without the constant DP engagement.

---

[6] MySQL: Bought out by Sun Microsystems which is now owned by Oracle.
[7] E-R: Entity Relationship data model.
[8] DP: Data Processing departments as they were known at that time.
[9] And later and more ubiquitous was LOTUS-123.

This independence from DP departments had a liberating effect on the executives of organisations. What became clear to them was that their DP people were not as critical as once thought; and that the corporate world did not collapse once these PCs were were deployed.

3.5 The Internet

The second thing after PCs and spreadsheets was international data communications, Usenet and email. In hindsight this event created the birth of the global village. By subscribing to your local ISP of the time you could plug into the wisdom of thousands of savants and exchange electronic mail with them and others instantly. Back in the 1980s some quite serious problems could be tackled by joining the appropriate user group and taking a few tentative steps in asking for help. Like today's Wikipedia and Web2.0 the newsgroups were dominated by the loudest and most shrill voices and those with the most apparent authority. It is not surprising then that many vendors spent considerable budget ensuring that their message was being received loud and clear.

It is also important to remember that the Internet of the 20th century was not yet indexed as is today's Internet and search engines were still not even a dream. If you wanted information you needed to know where to get it. Vendors were very quick to see this Internet thing as a new marketing opportunity.

3.6 Australian SMEs

Because of simple scale factors, Australia had a lot of small to medium enterprises (SMEs) for whom mainframe solutions were inappropriate and who might have turned to a bureau solution. The new mini-computers [17, 18] became attractive options and were actively pursued. Companies whose IT was agile enough to 'move with the times' often gained significant competitive advantage by simply having better IT solutions than their competitors.

Examples of this were the burgeoning Credit Union movement in Australia, sophisticated Insurance and Library solutions, and very popular manufacturing and distribution systems reminiscent of the original TRW product and precursors to the MRP and MRP-II[10] products.

In a later section I will mention the impact of 4th generation languages. However, Australian SMEs took great advantage of the flexibility of the data model, the power of programming and reporting and the rapid applications development opportunities afforded by the these 4GL environments by developing bespoke applications that genuinely met their specific business needs.

3.7 Voices from the Australian Pick Community in 1992

The Australian Pick community was quite vibrant in those days, led mainly by competing vendors of specific Pick flavours. A snapshot of the conversations of the day identified surprisingly few controversies.

[10] MRP: Materials and Resource Planning – precursors to modern ERP packages.

Peter Fenwick suggested that "*Pick's future is rosy because of its acceptance of Unix (Open Systems) as a host operating system. Pick is far more efficient than its competitors so one can run more users on equivalent hardware*" [19]. He was one of many who were moving away from Dick Pick's view that a hand written operating system was more powerful than Open Source (aka Unix in its various flavours). Rob Coulsen, a major reseller of the time agreed and added that "*Pick's success will be driven by its breadth of applications*" [20].

Others like Terry Leister from vMark [21], Al Dei Maggi of Sequent [22] and Charles Cave from Unidata [23] also saw that a GUI (Graphical User Interface) was key to Pick's ongoing success. Tom Couvret of Prime [24]. And Bob Highland of GA [25] felt that the new data communications technologies needed to be addressed.

Many like Tim Chianti from Apscore [26], Mike Ferris from Unipix [27] and John Buchanan of Triad [28] all thought that Pick needed to embrace interoperability with SQL, the 'mainstream' relational database vendors as well as integration with the emerging Microsoft Office suite.

An interesting comment of that time was from Barry Churchill of the NRMA who stated "*Pick must employ the principles of TQM in continually addressing seamless integration with today's (1992) technology. Pick needs to listen to its users to understand what their needs are. After all, it's the 'users' who will buy the business systems*" [29]. But a more light-hearted comment was made by Alan Glassman from BIX who suggested "*Pick needed to be ported to the 'state-of-the-art' 3270/RJE and HASP protocols. Microsoft to re-introduce command line for Windows/LANMan and Client/Server systems. ADDS terminal division releases a toast-r-oven connection to their 9000 series terminal*" [30].

The recurring themes from the Pick community were that Pick as an operating system was doomed and that Unix was to be the host of the future. Interoperability with the burgeoning PC marketplace and other systems (RDBMS[11]) was the next theme followed by the emerging industry standards such as SQL. Networking was addressed by some and its relative low profile in these discussions indicated how little the Pick community thought about TCP/IP, LANs and distributed databases.

3.8 Pick's Popularity

During the 1980s through to Y2K Pick boasted that it had more business solutions available than any other environment. There was even a publication called 'The Business Software Catalog' that detailed over 3,000 such business applications [31]. The vast library of systems written in Pick/BASIC could be ported from single user machines through to high-end symmetric multi-processors with redundant non-stop architecture capabilities supporting thousands of real-time users.

In addition there were several integrated development environments based on Pick called fourth generation languages (4GL) or CASE[12] tools that ranged from

[11] RDBMS: Relational Database Management System.
[12] CASE: Computer Aided Software Engineering.

elementary program generators such as Wizard[13] to truly configuration and data driven enterprise-class engines such as Cuebic[14], SB+[15] and Posh[16].

In the 1990s it was not uncommon for an organization to identify a software solution that it needed and to purchase (separately) the hardware platform upon which to mount an operating system (such as Unix), a version of Pick (such as Universe), a 4GL (such as Posh) and the accounting, finance and HR system written in Posh. The average CFO could not understand why it was not possible to get everything from one shop on the one invoice, like with an IBM solution.

There was great end-user flexibility in the choices available and numerous vendors each able to supply the 'perfect solution'. The competition was consequently brisk and in high value cases, immensely pressured. The various 'Pick' vendors robustly vied with each other for the prize of the Operating System licence that was often tied to a hardware platform because of the limited ports made by that vendor.

The other edge of the sword was that when errors or difficulties arose it was often difficult to identify the culprit – hardware, operating system, DBMS, 4GL or applications software and it was not uncommon for each to blame the other.

3.9 Just before Y2k

As the Y2K event approached, the 'mainstream' RDBMS community was mounting a major marketing campaign along with emerging larger software houses like SAP guaranteeing Y2K compliance and promising many years of trouble-free use with their 'best of breed' and 'world's best practice' products. This idea appealed to many 'C' level executives and a lot of organisations who were struggling with their legacy COBOL, RPGII and PL/1-based systems, elected to adopt typically SAP or Oracle solutions.

There was comfort in conforming to peer group pressure that was vigorously supported by so many seemingly knowledgeable people in so many forums. After all, if everyone was going that way then surely everyone can't be wrong?

The emergence of a new marketing jargon embracing such concepts as best of breed and world's best practice started to bite. It was commonplace then to hear these terms used by vendors who claimed them for themselves. Their international marketing penetration brought the use of these terms into the current corporate language, but most importantly, associated with their particular products.

Furthermore, these vendors bypassed the normal ICT decision-making processes by presenting their sales pitches to the senior executives like CEOs and CFOs bypassing the impotent IT and DP managers. Company directors were targeted directly through their associations, a practice just unknown amongst the Pick community.

---

[13] Wizard: An early program generator that 'created' PICK/Basic code from user entered parameters to describe transaction processes against Pick databases.

[14] Cuebic: A name alluding to the three dimensional nature of the Pick data model.

[15] SB+: System Builder plus.

[16] Posh: An acronym for 'Port Out Starboard Home', the preferred window allocation on trans-Atlantic boat trips. The name was adopted by its designer and developer Warren Dickins, a boating enthusiast.

3.10 What Changed?

At the Y2K event on Saturday January 1st 2000 the world did not end, electricity kept coming out of power points, water flowed as did gas. The IT community rejoiced in their collective wisdom and ability to prevent the disasters that were predicted. However, a number of important new influences or orthodoxies had emerged and had taken root…

- Alignment between cost and value and quality:
  - The acceptance of products such as Oracle and SAP and their million dollar price tags by financial controllers seeking Y2K immunity.
  - How can something costing $150k possibly be as good as something costing $1.5m?
- Microsoft establishes universal acceptability of faulty software:
  - The now infamous EULA states unequivocally that the software you have bought is not guaranteed to work…
- A new corporate jargon emerges:
  - Multi-million dollar applications from Europe and the USA with labels such 'best of breed', 'world's best practice' and 'enterprise class'.
  - Would you be brave enough to argue against a product solution that was 'acknowledged the best of breed' for your industry sector?
- Decision making on IT acquisitions went from the IT people to the accountants:
  - Accountants who have now been liberated from their IT departments are now calling the shots on high value IT decisions.
  - There is a growing atmosphere of suspicion about the justifiability of the huge Y2K expenditures [32].
- The Windows GUI is the only practical user interface to a computer.
- The Internet (and TCP/IP) is now the accepted data communications orthodoxy.
- Safety in numbers:
  - In the older days, nobody was ever sacked for buying IBM.
  - Corporate leaders meet and compare notes about their respective IT solutions.
- Universities control the corporate thinking:
  - Thousands of graduates have only ever been exposed to the Codd relational model and cannot conceive of other models.
  - Students are increasingly educated using artificial business models that neatly fit their relational toolkits and avoid real-world complexity.

The Pick community was certainly not united. Even the local Victorian IPUA attempted to assuage their vendors' sensibilities by renaming themselves to the 'Multi-Dimensional Database Forum' to remove the word 'Pick' from the association name. Few licensees embraced TCP/IP and fewer still acknowledged SQL and interoperability, and those that did make such an investment ensured that it remained proprietary and certainly not portable to other Pick vendors.

Perhaps the most important, if not subtle change was how corporate ICT decision making had evolved. The vendors, both ERP and their associated database suppliers no longer dealt with anyone lower than the absolute top level decision makers. After all, why negotiate with an IT manager when they would have to put a submission to a higher authority.

Associated with this exclusion of the IT personnel is the windfall consequence of financial decisions being made with only the highest level decision-maker. The vendors are then only beholding to the boards of directors who have already been marketed at their associations such as the Australian Institute of Company Directors.

4 Conclusions

The Pick community's inability to work together on keeping up with technology will be one of the reasons that many organisations will be electing to drop Pick as their preferred business platform and select more expensive mainstream solutions.

It is my opinion that industry had become so used to the two dimensional database model that interfaces tended to mirror the spreadsheet in concept. Pick's data model being inherently three (or even four) dimensional was difficult to effectively represent with the tools of the day and it never made an effort to effectively market the innate advantage that such structures offered. It is of passing interest that today's advanced HTML is now capable of such a representation without too much difficulty.

It is also curious to note that no popular operating system has been named after its inventor. Had PickOS been named something like OSMV (OS Multi-value) then a lot of criticism might have been avoided, and many days in the California Superior Court likewise avoided. And had Dick Pick named his programming language similarly after the famous courtesans like Pascal and Ada, perhaps something like 'Zion' might have reduced the criticisms of the language name.

But it is perception that matters today and perception is reality. Perception is a controlled substance and today the accepted orthodoxies are Windows point and click, Internet, Unix, rigorous Codd relational model and it just has to be 'world's best practice', whatever that means. With senior executives independently making the key decisions today, businesses are happy to change their business models and processes to align with some 'best' standard from Europe or America as manifested in a small selection of 'Enterprise' systems. And thanks to the largest software manufacturer in the world, it doesn't necessarily have to work entirely properly either.

References

1. Gadamer, H.-G. (ed.): The Historicity of Understanding. Critical Sociology, Selected Readings, ed. P. Connerton. Penguin Books Ltd., Harmondsworth (1976)
2. Lukaitis, S.: Investigating Business and IT Alignment. Heidelberg Press, Heidelberg (2012)
3. Gadamer, H.-G.: Truth and Method, 3rd edn. Continuum, London (2004)
4. Forrester, C.: Interview (2010)
5. Hewlett-Packard: MPE-IV Software Pocket Guide. Hewlett Packard, Cupertino (1981)
6. Lukaitis, S.A.: A Short History of the Pick Environment in Australia. In: Tatnall, A. (ed.) History of Computing: Learning from the Past, pp. 146–158. Springer, Berlin (2010)
7. Brooks, F.P.: Mythical Man Month. Addison-Wesley, Reading (1975)

8. Denning, P.J.: The working set model for program behavior. Communications of the ACM 11(5), 323–333 (1968)
9. Rodstein, H.E.: Pick For Professionals - Advanced Methods and Techniques. In: Sisk, J.E. (ed.) The Pick Library. TAB Professional and Reference Books, Blue Ridge (1990)
10. Martin, J.: Computer Database organization, 1st edn. Prentice-Hall Inc., Englewood Cliffs (1975)
11. Volokh, E.: Relational Databases vs IMAGE: What the Fuss is all About. In: Interex 1986, Detroit (1986)
12. Codd, E.F.: A relational model of data for large shared data banks. Commun. ACM 13(6), 377–387 (1970)
13. Codd, E.F.: Extending the database relational model to capture more meaning. ACM Trans. Database Syst. 4(4), 397–434 (1979)
14. Codd, E.F.: The relational model for database management: version 2, p. 567. Addison-Wesley Longman Publishing Co., Inc. (1990)
15. RMIT, Advanced College Handbook. RMIT Advanced College, Melbourne (1985)
16. Tsichritzis, D.C., Lochovsky, F.H.: Data Models. Prentice-Hall, Inc., Englewood Cliffs (1982)
17. Brinton, J.B.: New minis push into power era. In: Electronics 1979 (1979)
18. Wade, L.: Superminis: Evolution or quantum jump. In: Digital Design 1979 (1979)
19. Fenwick, P.: New Directions in Information Technology. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
20. Coulson, R.: The Future of PIK in Australia. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
21. Leister, T.: Where is Pick Going? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
22. Maggi, A.D.: Pick - From Proprietary to open Systems. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
23. Cave, C.: Where is Pick Heading? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
24. Couvret, T.: Pick - the Next 10 Years. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
25. Highland, B.: Whither Pick in the 90s? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
26. Cianchi, T.: Will Pick (and the software business as we know it) Survive? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
27. Ferris, M.: Where is Pick Headed in the Next 5-10 Years? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
28. Buchanan, J.: Whither Pick or Wither Pick? In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
29. Churchill, B.: The future of Pick - a User's View. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
30. Glassman, A.: Predictions for 1992. In: Pick-Up 1992. International Pick Users Association (NSW), Neutral Bay (1992)
31. IDBMA, Industry Impact Study - the Pick Marketplace. International Database Management Association Inc., San Diego, California (1989)
32. Lukaitis, S.: The key issues that impact on the successful alignment of business and its IT function. In: Department of Information Systems 2010. Deakin, Melbourne (2010)

Evolution of Computer Science Education in the Purview of Free Education

S.T. Nandasara

University of Colombo School of Computing (UCSC), Colombo, Sri Lanka
stn@ucsc.lk, nandasara@yahoo.com

Abstract. Sri Lanka was one of the first developing nations to understand the importance of investing in human resources and promoting gender equality. Advances made by the country in health and education are at par with those of the advanced countries. Near universal literacy and a well-developed system of school education, places Sri Lanka as a leader in education in South Asia and amongst the top-performing countries in the entire world. High priority was given to education for over six decades whereby free education has resulted in an increase of school enrolment from 1 million in 1947 to a peak of 4.2 million in the mid-nineties, achieving universal primary education and a high level of participation in secondary education. Net enrolment ratio (in 2004) of 97.9% at primary level, a completion ratio of 95%, and a gender parity of 96% are laudable achievements.

In terms of multi-disciplinary e-readiness criteria, Sri Lanka is also ranked higher than the neighbouring Asian countries in spite of the relatively slow penetration of computers, internet and other telecommunication media. According to the United Nations Report on e-Government Survey of 2008, Sri Lanka is ranked at eighth among twenty other countries in the Southeast/South Asian region.

Keywords: Computer science education, university, Sri Lanka, free education, historical perspective.

1 The Land of Free Education

The Democratic Socialist Republic of Sri Lanka, known in short as Sri Lanka is a free, sovereign, and independent democratic socialist republic. It is an island in the Indian Ocean located close and to the north of the equator. Sri Lanka has a total land area of 65,610 sq. km, encompassing beaches, green vegetation, and a mountainous mass somewhat south of centre with heights reaching about 2,500 meters, surrounded by broad plains.

In 1805, after the British invasion, the education system in Sri Lanka was developed based on the British System. Sri Lanka received independence from the British on the 4th February 1948. Hon. C. W. W. Kannangara, the then Minister of Education in the State Council (1931-1947) introduced several far reaching policies in education, a few years before receiving independence from the British. They included

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 127–142, 2012.

primary education in the mother tongue, free education for primary, secondary and higher education, establishment of a number of quality schools in all regions of the country and the provision of a free mid-day meal in schools. The striding measures earned him the name 'The Father of Free Education' and have contributed immensely towards opening up higher education to the masses as opposed to the elite that benefited till then.

In Sri Lanka, though the government plays the primary role in education, it does not claim monopoly over it. There are many Buddhist *pansala* and *pirivena*, Muslim schools, and Christian schools in the country. Roman Catholic Churches alone operates several hundred schools enrolling over 80,000 children.

2 Origin, Size and Growth of Higher Education

The roots of the higher education system in Sri Lanka can be traced back to the Ceylon Medical College established in 1870 by the then Government under the leadership of the Governor, Sir Hercules Robinson and setting up of the Ceylon University College in 1921. This College affiliated to the University of London to prepared students for external examinations conducted by it. When the University of Ceylon (later renamed as University of Sri Lanka) was established in 1st July 1942, by proclamation under the Ceylon University Ordinance, No. 20 of 1942, both these Colleges were amalgamated with it. The Ceylon Medical Collage became the Faculty of Medicine, and the Ceylon University Collage was divided into Faculties of Oriental Studies, and Arts and Science. The Faculty of Engineering was established in Colombo in 1956. Since the establishment of Ceylon Medical College in 1870, it could claim to have been associated with higher education for over 140 years.

The University of Ceylon was established in Colombo, in 1942, and on the 6th October 1952 a new campus in Peradeniya, Kandy was inaugurated in the tea and rubber plantation near the banks of Mahaweli River, the longest river in the country. Sir Ivor Jennings was the first Vice-Chancellor to the Peradeniya Campus. Dentistry, which was confined initially to one department in the Faculty of Medicine in Colombo, was moved to Peradeniya in 1954, and the department was renamed as the Dental School, with one unitary department of Dental Surgery. The Faculty of Engineering which was initially established in Colombo in 1956 was transferred to Peradeniya in 1965. In 1967, it was made a separate university with the name 'University of Ceylon - Peradeniya' and it became the independent 'University of Peradeniya' in 1979. Today, the Peradeniya University stands as a prominent educational institution in the country, catering to the needs of students from a variety of academic streams and ethnic groups that have been at Peradeniya for over the last 60 years.

Likewise, for technical education, the Government Technical School (GTS), established in 1893 at Maradana, Colombo (its name was changed to the Ceylon Technical College-CTC in 1906) was the first institution for formal technical education in Sri Lanka. The CTC provided technical education in Civil, Electrical and Mechanical Engineering fields and prepared candidates for the external degrees in Engineering of the University of London in England. In 1960, the technician courses at the CTC were transferred to a new establishment called the Institute of Practical Technology (IPT) at the present premises of the University of Moratuwa. With the recommendation of the Government appointed Commission of Inquiry on Technical Education report of 1963, IPT earned university status, as the Katubedda Campus of the single University of Ceylon, on 15th February 1972.

The university system in Sri Lanka operates within the framework laid down in the Universities Act 16 of 1978. This reintroduced the system of separate universities functioning under the direction of the University Grants Commission (UGC). Thus the universities, namely, Colombo, Peradeniya, Moratuwa, Jaffna, Kelaniya and Jayawardenapura became separate universities. The UGC, which functions as the apex body in the university system, allocates funds to the universities and institutes, serves as the central admission agency for undergraduate degree courses, monitors and reviews the working of the university system with a view to maintaining standards, and implements national polices in respect of university education. Earlier higher education was under the Ministry of Education. In 2007, a separate Ministry of Higher Education was created to give greater thrust to higher education development. UGC statistics show that there were 15 public universities, 7 postgraduate institutes, and 9 other higher education institutions by 2010. In addition, there are 2 universities for Buddhists studies established under the Ministry of Higher Education.

As per 2010 UGC data, 73,398 undergraduate students were enrolled in the public universities in 2010. In addition, 8,984 students were enrolled in universities for postgraduate degrees and 14,187 students were enrolled in higher educational institutes for certificate, diploma and postgraduate programs. 29,222 students are enrolled in the Open University of Sri Lanka in the year 2010. While seven universities are large and have enrolment in a range of subjects, the remaining are mainly regional universities with small enrolments in science based courses. Another about 150,000 students (with two-third of them women) take exams without attending classes and are ‘external students’.

After 13-year of schooling, about 125,284 qualified for admission to the universities in the academic year 2009/2010. Out of this, only 21,547, which is about 5% of the average age cohort, were selected for university admissions. Others either found entry into the labour market or appeared for university exams as external students.

3 Computing Education in Sri Lanka

More than 50% of the graduates in the university system are enrolled in arts, science and commerce subjects with low employability. There are few higher educational institutes offering degree and postgraduate degrees in the field of computer science and information technology, where the demanding job opportunities exist in the market. In 2010 only about 2,756 students were enrolled in the field of Computer Science, Information Technology and Information & Communication Technology (for undergraduate studies) fields by the Eastern University (24), University of Ruhuna (42), University of Kelaniya (168), Sabaragamuwa University (191), University of Jaffna (197), South Eastern University (199), University of Peradeniya (201), Rajarata University (226), Uva Wellassa University (322), University of Moratuwa (449) and the University of Colombo School of Computing (737). In addition the University of Colombo School of Computing registered 2,514 external students for Bachelor of

Information Technology (BIT), external degree in 2010 and registered 2,560 students in the year 2011. It is the only external degree in IT offered by the university systems. Their total student enrolment for the first year external degree program was 4,181 in 2011.

4 'Computer Revolution' in Sri Lanka

As mentioned earlier, the University of Colombo established in the year 1870, could now claim to have been associated with higher education for nearly 140 years, and has become a centre of excellence of international repute that contributes significantly towards national development and human resource development in the field of computer science and information and communication technology through the Department of Mathematics, Statistical Unit, Computer Centre, Institute of Computer Technology and the University of Colombo School of Computing (UCSC). UCSC as a institute celebrated its 25^{th} Anniversary on the 27^{th} October, 2010, and has been in the forefront of the 'Computer Revolution' in Sri Lanka, having introduced the teaching of computer programming and applications as early as in 1967, more than a decade before other educational institutions, thereby producing, over the years, a large number of pioneer computer scientists and IT graduates out of students entering the university from a variety of disciplines. Students who earned knowledge from the University of Colombo are presently employed as researchers, educators, data processing managers, analyst programmers, software engineers and in many other areas in the professional field of information technology, not only in Sri Lanka but also in South and South East Asian as well as the African and Middle Eastern Regions.

5 Teaching FORTRAN without Computers

Teaching of computer programming and the use of computer applications for research and teaching at the University of Colombo commenced in 1967. The ICL 1901 mini computer at the State Engineering Corporation was used free of charge thanks to the encouragement given by their management towards the introduction of computing at the University of Colombo in Sri Lanka. A few years later in 1971, the Department of Census and Statistics allowed the University of Colombo free computer time on their IBM 360/25. The fact that these installations were close to the University of Colombo and the interest of both organizations in statistical and scientific applications helped the university researchers to make very good use of this invaluable gesture.

According to the 'Development Plan of the Faculty of Science', that the University of Colombo introduced in 1975, the first teaching was restricted to FORTRAN programming for staff and students as an extra-curricular activity. However, within a few years a combined course in Computer Programming and Numerical Methods was introduced as a paper in Applied Mathematics for the general degree. A single course unit in Computer Programming was introduced in the late seventies. The number of courses offered increased in the late 1970s and the course unit system operating at the University of Colombo made it possible to offer a range of third year degree options. After the study: 'The Introduction of Computer Science into Degree Studies in

Sri Lanka' submitted to the Ministry of Higher Education in 1983 by Professor C. M. Reeves and the Computer Policy Report of NARESA (1984) Computer Programming and Applications became a part of most postgraduate and undergraduate courses in the country. Computer Applications was introduced in some of the courses meant for those who were not scientists. During the initial stage of this activity, practical sessions were not available for the students due to the non-availability of computers at the University of Colombo. However, the students who were keen to take the course unit in FORTRAN and COBOL programming had to imagine the machine's internal functionality and code accordingly to solve problems only on the paper which then had to be transferred to punched cards and sent to a computer elsewhere.

The above programs were initiated by the Mathematics Department of the University of Colombo which was at that time developing the field of Applied Statistics. A Statistical Unit that was established in 1968 at the request of the Department of Mathematics, the Department of Geography and the Faculty of Medicine did not survive long, due mainly to the loss of several key staff members to overseas universities. However, thanks to the support received from the staff of the Department of Applied Statistics of the University of Reading, UK with British funding, the Statistical Unit was revived. The University of Reading was involved in a link arrangement from 1974 and helped the Statistical Unit to progress steadily to become a National Centre for Statistical Research, Teaching and Consultancy. Dr Roger Stern from the University of Reading, UK who had arrived in 1974 on a two year assignment to help develop Statistics at Colombo was very knowledgeable in the use of computers in Applied Statistics and helped establish the Statistical Consultancy and Data Processing Services (SCADPS), the forerunner of many of the later developments.

Dr Roger Stern was instrumental in the establishment of an academic link between the Statistical Unit and the Department of Applied Statistics of the University of Reading, UK with British Government assistance. This link, one of the earliest and most successful of the inter-University links in Sri Lanka, continued until the early nineties and helped establish the M.Sc. Courses in Applied Statistics and also developed Statistical consultancy. Ian Wilson also from Reading who replaced Roger Stern saw those developments through.

6 Computer Programming Course Units for Non-Science Students

Under the Higher Education Reforms that took place in 1972 (Jayarthne Report), the Department of Mathematics and the Statistical Unit of the University of Colombo made a remarkable attempt to initiate new course units in Mathematics, Statistics and FORTRAN Programming to the Statistical Services job stream for Faculty of Art degree students following the newly introduced Special Degree in Development Studies. 30 students were selected from 210 reading for the above degree and were given the opportunity to follow a special degree stream which was specially designed and managed by the Department of Mathematics of the University of Colombo.

7 First Computer of the University of Colombo

The requirements of the Statistical Unit of the University of Colombo in the form of computing support for research, consultancy and teaching resulted in a substantial increase on the computing field. Soon thereafter it was felt that the Statistical Unit should have its own computer for teaching and consultancy in addition to using the free computer time available for research at the Department of Census and Statistics as indicated earlier. An HP9825 desktop microcomputer (HP claimed that this is a Desk Calculator) was obtained under the link arrangement with British Government Assistance in 1977. This introduced in-house computing at the University of Colombo and a small Computer Unit was established in 1977. (The Computer Unit was later termed as the Computer Centre in 1981.) Unfortunately, difficulties arising out of the non-availability of local servicing facilities made this excellent machine (at that time) rather unpredictable. In 1978, the HP9825 microcomputer was sent back to the UK for repairs, and it was lost in the process and was never seen again. In 1980, a serious attempt was made to fulfil the need for computing power with a modern computer (-proposal for the purchase of a mini computer for the University of Colombo) capable of statistical work and therefore having a configuration allowing the implementation of some of the well-known statistical packages. The result was to have Data General NOVA/4 minicomputer on loan in 1980 and the purchase of a Data General Eclipse S/140 minicomputer in 1981, with time sharing, multi-user interactive capabilities together with adequate disc storage (20 MB), RAM (128 KB) and a magnetic tape drive with eight terminals. The purchase was made possible by the pooling of funds from Netherlands Universities Foundation for International Cooperation (NUFFIC), University Grants Commission (UGC) and the Equipment Vote of the Faculty of Science of the University of Colombo. We wish to record our appreciation of this timely assistance to introduce a multi-user, multitasking computer to the University of Colombo. One major objective in selecting the particular Computer Configuration was the implementation of third party software packages used worldwide such as GLIM, MINITAB and SPSS which have been implemented on this machine, and thus proved to be an invaluable asset, especially in the consultancy service segment of the Statistical Unit.

Up to the early seventies, teaching was confined to one member of staff. In the late seventies, more were available and a post of Computer Programmer was also created. The Statistical Officers and support staff recruited for the Statistical Unit were able to double up as teachers of computing. Some of the academic staff who had obtained their postgraduate degree in statistics overseas also devoted a considerable amount of their efforts towards matters relating to computer applications.

The Computer Centre was formed in 1981 and expanded the curriculum on Computing by introducing a new course in *Numerical Analysis* to the undergraduate curriculum. At the same time, it also launched courses in *Computer Application* for non-university students. These courses proved to be very popular amongst both IT professionals and University staff, who up to that time had lacked exposure to formal techniques in Computer Programming.

8 Computers Go Public

One of the main demarcation points of the history of computing in Sri Lanka was the introduction of computers for assisting the Commissioner of Elections to process the results of the National Presidential Election in November, 1982. In late 1981, thanks to the Reading-Colombo Link programme, Colombo made a request to Overseas Development Assistance (ODA), UK for assistance to purchase a few BBC microcomputers, due to be released to the market in 1982. When they did arrive, in October 1982, this remarkable microcomputer was an immediate success and the Computer Centre received much publicity among the public by their computer display which enabled the telecast of the Presidential Elections of 1982 over National Television (*Rupavahini*). This saw the use of computers to process election results and the release of results as graphics displays for telecasting. For this purpose floppy disk drives for the BBC microcomputer were used even before they were introduced to the UK market. This process of release of results of every national election has continued without a single break thereafter, with technological improvement at every stage.

The BBC microcomputer was exploited to be used as a tool for teaching statistics, programming and also for research in addition to its value as an aid for promoting computer literacy. In 1983, ODA, UK granted GBP 10,000 in addition to the GBP 3,000 given in 1982 and the UGC grant for microcomputers was used together with this grant to establish a networked laboratory of 13 BBC microcomputers and also purchased several stand-alone BBC microcomputers, Amiga Computers, Acorn computers, WANG-PC computers, A Kaypro-2 and RadioShack TRS80-16 computers with disc drives, second processors and other peripherals and software.

9 First Ever Computer Courses for Public and Government Employees

With high publicity received by the University of Colombo due to the release of the Computer Assisted Presidential Election results, the Computer Centre was able to inaugurate a Certificate Course in Computer Applications for the general public, to be held during weekends, not interfering with the undergraduate courses. These courses were primarily meant for the employed to gain knowledge of computer applications rather than to those wanting to learn computer programming for employment. Furthermore, preference was given to those in the scientific and educational sectors.

As highlighted by the National Computer Policy for Sri Lanka – Report of the special working committee of the Natural Resources, Energy and Science Authority of Sri Lanka (1983) – the University of Colombo realized an important aspect of the its Computer Centre's extension work through a *Computer Literacy programme* in Schools. In early 1983, *Computer Assisted Education* was introduced by the Ministry of Education to three schools as a pilot project and the University of Colombo helped the Ministry in launching its Computer Education Programme through teacher training. Literacy courses were conducted for students of the Institute of Workers' Education of the University of Colombo and proved to be a success.

While developing its computing facilities and skills of its staff in computing, the University of Colombo also took steps to actively encourage the use of computers in scientific research. A computer exhibition on 'Computers for Scientific Research' was held by the Computer Centre during Annual sessions of the Sri Lanka Association for the Advancement of Science (SLAAS) in December 1982 and several seminars, training sessions etc. have been held since then. The large number of computers and applications exhibited at the Natural Resources, Energy and Science Authority of Sri Lanka (NARESA) sponsored the 'National Exhibition on Science and Technology' held in 1985 clearly indicated the advances made by then.

Due to increased demand the Data General Mini Computer was upgraded to have 15 terminals and later facilities at the Computer Centre were upgraded with an additional Data General MV2000 with 1MB RAM and AOS/VS operating system. There were by then three microcomputer laboratories of BBC microcomputers and accessories at Colombo University. The IBM-PC and WANG-PC computers were in heavy demand. A Kaypro-2 received as a gift, and the RadioShack TRS80-16 were used for student projects as well as consultancy work.

The staff of the Computer Centre succeeded in making the BBC micro a terminal for the Data General minicomputer thus making available a versatile and low cost terminal as well as a device for data transfer between two computer systems.

10 Computing for National Interest

Another important contribution by the staff has been the part played in the development of a Statistical Package for the BBC microcomputer and an IBM PC Compatible computer called 'INSTAT', developed with the collaboration of the University of Reading, UK and released to the overseas market (1984). This software has been the basis for many a course in Statistics both in Sri Lanka and overseas.

In technical research in the field of computer technology and the involved in national interest in information technology education since early eighties, the University of Colombo has been actively engaged in the development of local language (Sinhala/Tamil) interfaces for personal computers in applications areas such as word processing, database management systems and spreadsheet applications; a research area nobody has hitherto undertaken in Sri Lanka. This work mainly involves developing software and hardware based solutions for the IBM compatible computers. Development of a very early tri-lingual (Sinhala/Tamil/English) Sinhala Basic Input and Output System – SBIOS and Tamil Basic Input and Output System – TBIOS for IBM PC-DOS with a very popular multilingual (Sinhala/English) word processor '*WadanTharuwa*' (වදන්තරුව), Tri-lingual (Sinhala/Tamil/English) Desktop Publishing System for Xerox Ventura – *Athwela* (අත්වැල), Tri-lingual input method for Windows 95/98 '*Sarasavi*' (සරසවි) packages and a UNICODE version '*WinMASS Sarasavi*' to the Sri Lankan market. Thus was an invaluable contribution to the nation as a result. The initial product received wide publicity and recognition.

These developments have been part of the pioneering work concerned with the introduction of local language computing and providing a world first ever Indic origin tri-lingual National Web Site (www.lk) for the Government Information Department of Sri Lanka, inaugurated by the late Honorable Minister Mr. Dharmasiry Bandaranayake at a public ceremony held in Bandaranayake Memorial International Conference Hall (BMICH) on 15th September, 1996. The Tamil language related research and development activities were continued thereafter.

One of the other major roles played by UCSC and the ICT was to coordinate research and recommend the draft Sinhala Code for ISO/IEC JTC 1/SC 2/WG 2 to include in UNICODE. Based upon the recommendation made by the Computer and Information Technology Council of Sri Lanka (CINTEC) and the subsequent approval of Sri Lanka Standard Institute of Sri Lanka, the ISO/IEC JTC 1/SC 2/WG 2 Meeting #32 was held in 1997-04-01, in June, 1997 and ISO/IEC JTC 1/SC 2/WG 2 Meeting #33 held in Crete, Greece where the draft Sinhala Code was discussed intensively. After a few ad-hoc committee meetings with National delegates and other nominated country delegates concluded the repertoire, names, and arrangements for Sinhala script based on Sri Lankan proposal with slight modification with the support of the majority of delegates from Canada, Netherlands, Greece, UK, USA and forwarded to WG 2 for adoption and processing at a pDAM stage. This was ratified at the WG 2 meeting #34, Redmond, WA; 1998-03-16 held at Seattle, USA and the Sinhala Code Chart was included in Unicode Version 3.0.

Other software developed in terms of national interest includes the graphical display for the media such as the TV Programme Parade, the Cricket Scoreboard, SAAF Games Display and National Quiz programs organized, managed and conducted by University of Colombo in early 1980s.

The computing staff at Colombo has been engaged in dissemination of knowledge to the nation's citizens through public and private media channels for more than two decades. These television and radio programs received the highest ratings and were most popular programs all the time. These programs, namely, the 'Computing for Schools' (5 episodes, 1995), IT programs directed by Daya Liyanage from MTV TV channel (5 episodes, 1996), 'අන්තර්ජාලය ඔබේ නිවසට' – weekly one hour LIVE program for IT related technical discussion on Sri Lanka Rupavahini Corporation (SLRC) (National Television), Sundays 7.00pm – 8.00pm (- more than 150 episodes, 1997-2001), 'Internet and You' a weekly one hour LIVE program for IT and internet related technical discussion on Sri Lanka Broadcasting Corporation (SLBC) Wednesdays 9.00am to 10.00am (- more than 120 Programs, 1997 – 2001), 'e@ශනිදා' - weekly one hour LIVE program for IT related technical discussion on SLRC Sundays 7.00am - 8.00am, (5 Programs, 2003), 'IT-Quiz 2000' nation-wide IT quiz competition telecasted on National Television, 'FORUM for BIT' – a weekly 30 minutes educational TV program for Bachelor Degree in Information Technology (BIT) telecasted on SLRC and TVLanka, (132 Programs from Oct. 2003 to Sep. 2005), 'සියබස් අලංකාරය' - weekly television program on Independent Television Network (ITN) (22 Episodes, 2005), 'පරිගණක පරාදීසය' - weekly 30 minutes children program on SLRC (Thursday 5.00pm - 5.30pm, 2008), 'e-මිතුරෝ' - weekly 30 minutes IT program for children on SLRC (Thursday from 5.00pm to 5.30pm, 2009), 'ආසයි ලේසියි IT' - weekly 30 minutes program in Information and Communication Technology for students in schools and universities, and general public on SLRC,) Saturdays from

6.00pm to 6.30pm, (102 episodes, from May 8th, 2010 and onward), ‘ජාතික පාසල’ - 24 programs for ‘Information and Communication Technology’ for A/L syllabus were recorded and is being telecasting from 1st October, 2011 to 31st of July, 2012 on Rupavahihi and many other programs on National and other Television channels for IT related discussions.

11 Early Regional Collaboration

The University of Colombo was involved with the organisation of the First Asian Regional College on Microprocessors in June 1984 supported by the International Centre for Theoretical Physics, Triest and co-sponsored by the UGC and the Council for Information Technology (CINTEC). This benefited 40 foreign and 32 Sri Lankan participants. A sum of US$ 120,000 was raised for the Collage. Of this US$10,000 was funds earmarked for the University of Colombo by UNESCO. This resulted in a valuable set of books being made available for the Library. Another international course was supported by the University of Reading and cosponsored by the Computer Centre and the Statistical Unit in December, 1984, on ‘Statistics in Agriculture’, with a heavy bias towards computer use in agriculture. 14 Foreign and 10 Local participants took part and they were supported by several international and national organizations including the Agrarian Research and Training Centre, Sri Lanka to mark the tenth Anniversary of the Colombo-Reading Link. This was repeated in 1985 for 30 local participants. It was during this course that the Statistical Unit of the Mathematics Department was elevated to the status of a Department – that of Statistics and Computer Science (DSCS).

In 1984, the Computer Centre helped the British Council and the Ministry of Education to conduct a two week course on ‘*Computer Education*’ for school teachers and curriculum developers. All these activities have provided valuable experience to the staff and have also contributed significantly towards promoting the development of computer applications in the country.

12 Staff Exchange Programs and Postgraduate Training

The British Assistance for Statistics provided for staff training in Computing too through a link with the University of Reading, UK. The UGC initiative following the Reeves Report ‘On the Introduction of Computer Science in to Degree Studies in Sri Lanka – A report to the Ministry of Higher Education’ (1983) also provided training in Computing at the University of Wales, UK. Subsequent support from the UNDP helped continue this trend. Many staff members returned with a Diploma, M.Sc., M.Phil., and PhD. in Computer Science during 80s and 90s as a result of these initiatives. Many received training in Japan too. Although there was some staff loss to the private sector or overseas, new blood was pumped into the Centre and incentives in the form of job satisfaction, additional remuneration received from extension courses and consultancy work, and an ever improving range of available hardware and software and good work environment, together with challenging projects made many stay, in spite of much better job prospects elsewhere.

As a result of the staff exchange programs, the University of Colombo actively engaged in developing study guide books for students and lecturers of the universities to introduce 'Software Design and Implementation: A PASCAL Based Course' with collaborative assistance received from the University of Keele, UK and The British Council, for the purpose of course material preparation. These materials were used by undergraduates in all the universities of Sri Lanka.

13 Collaborative Research Activities

A collaborative research project on crop and climate data between the University of Colombo and the University Reading resulted in a database being developed at Colombo University and many local and overseas groups and Research Institutes were able to obtain data for their research. This Crop and Climate Database project computerized the range of daily climate data, including temperature, rain, wind etc. collected from 100 metrological stations established around the island for the past 100 years. This, together with the consultancy work done in the areas of Agriculture, Health and Education made the Computer Centre a very useful resource in Sri Lanka. It developed expertise in the packages INFORMIX, SAS and SPSS and became the only expert group for such packages in the county at that time.

Collaborative work with Research Institutes such as the Rubber Research Institute (RRI), Tea Research Institute (TRI), Coconut Research Institute (CRI) and Meteorology Department grew during early 1980s with advice given on computing including evaluation of needs and assistance in purchasing of computer equipment and also training. The Centre was also able to give five microcomputers on loan to the Research Institute, complete with the Statistical Package INSTAT.

14 Establishment of a Fully Fledged Academic Department for Computing

In 1984, the Department of Mathematics made a proposal for the Establishment of a Computer Centre, a Department of Computer Science and Statistics and the commencement of a Postgraduate Diploma Courses in Computer Science submitted to the UGC. In January 1985, the existing Department of Mathematics split into two departments; one remained as the Mathematics Department, and the Department of Statistics and Computer Science (DSCS) was formed by merging the Statistical Unit and Computer Centre. It was only in 1986 that the formal separation took place as the separation of activities was not a simple exercise. The new department thus inherited skills in both disciplines and consequently occupies a unique position in the University system in Sri Lanka as a centre of excellence in both these fields. The new department's major objectives at the time of formation were to run specialized degree programs in both Statistics and Computer Science. To this end, further staff training was undertaken and M.Sc. qualified staff enrolled for Ph.D. research programs at British Universities with the help of more British Government aid. Immediately after the establishment of DSCS, a Postgraduate Diploma in Computer Applications was launched in 1986.

In 1987, Computing took a quantum leap forward with the establishment of a link between the DSCS and the Department of Computing Mathematics, University of Wales Collage of Cardiff, UK. This was made possible through a UNDP sponsored project which was set up to improve the Computing teaching infrastructure in the Department. Both B.Sc. and M.Sc. degree programs in Computer Science that resulted from this have proved to be very popular, with demand far exceeding the number of places available. The M.Sc. programs in Applied Statistics and Computer Science at the DSCS provided very cost-effective alternatives for the training of academic staff from local universities, and a number of such staff have availed themselves of this opportunity. The local IT industry has also benefited from this program with the availability of graduates with the right mix of theoretical and practical skills.

As of 1995, the DSCS further improved its Statistical Education initiative by introducing a special degree course in Statistics at undergraduate level and by restructuring its M.Sc. course in Applied Statistics to a more streamlined form. The special degree course in particular benefits from students undertaking placements in industry in the final year, thus enabling them to do projects which are relevant to the country at large.

The Department of Computer Science (DCS) of the University of Colombo was established in 2001 by splitting the Department of Statistics and Computer Science (DSCS), which functioned since 1985 as part of the Faculty of Science of the University of Colombo. While the DCS was responsible for undergraduate and postgraduate training in Computer Science, the Department of Statistics (DS) was responsible for statistics education in both undergraduate and postgraduate education.

Although the University of Colombo produced graduates with computer science as a subject for some years, its first batch of students specializing in Computer Science and obtaining the B.Sc. special degree graduated in 1992, the first such group in Sri Lanka.

15 Establishment of Institute of Computer Technology (ICT)

In early 1984, the University of Colombo Computer Centre, while consolidating its position as a leading computer installation and consultancy service, worked on plans for the establishment of an Institute of Computer Technology (ICT) with the assistance of the Japanese Government. Final approval was granted by the mission that visited Sri Lanka in March 1987 making available funding – amounting to 490 million Japanese Yen – from the Japanese fiscal year 1987/88. The agreement signed by the two governments required the Sri Lankan Government to provide building space, staff and their salaries and recurrent expenditure for the ICT.

An initiative of CINTEC, the University of Colombo and the UGC resulted in the establishment of the ICT at the University of Colombo in 1987 as an Institute established under the Universities Act. The ICT was provided with the largest mainframe computer system in the country then with other peripherals and staff training under Project Type Technical Co-operation of the Japan International Cooperation Agency (JICA), Government of Japan. The ICT was to conduct Postgraduate training programs to produce Analyst Programmers for the country. This was a result of the Japanese Mission's visits to the Computer Centre of the University of Colombo in April 1984 and in February 1986 in relation to the proposed Institute and very hard negotiations to win from among several proposals from other countries. Finally the University of Colombo was able to convince the Japanese Mission as well as the Government of Japan, of the need for such an institution. This was a milestone of success of computer education in Sri Lanka.

16 Third Country Training Programs

The Japanese assistance provided to the Institute of Computer Technology resulted in building sufficient capacity both in human resources and in facilities. The donor having seen the satisfactory completion of this phase moved on to the next where the ICT was expected to use these resources to provide training for those from other (third) countries through the Third Country Training Program (TCTP). Accordingly, from 1993 to 1998 a TCTP in Structured Systems Analysis and Design was held annually from fifteen Asian countries. On successful completion of this program, a second TCTP in Information Systems Engineering was conducted from 1998 to 2002 for twenty participants each from eighteen Asian, Far East and African countries. In 1998 the ICT received the JICA President's Award for the Best Regional Training Centre among its 60 JICA assisted countries. This excellent concept of south-south cooperation was later introduced to Sida, the Swedish international development agency who sponsored a TCTP in the Design, Installation, Management and Maintenance of Network Systems for twenty participants from Asia, Africa and even Latin American Countries for a few years.

17 First Decade of the 21st Century

The developments in Computing indicated above resulted in the University of Colombo becoming a Centre of Excellence by the dawn of the new millennium. Several landmark events took place thereafter enabling the consolidation of the status it had built up in the last four decades of the 20th Century.

The ever increasing demand for IT graduates both globally and locally combined with the inadequacy of the state sponsored free education system prompted the staff of the ICT together with DCS to launch a very innovative external degree program for the Bachelor of Information Technology Degree (BIT). This was an instant success with over 5,000 students registering for year one in 2000. The ICT and DCS developed the curriculum and were to hold the examinations while the University of Colombo was to award the degree. The private sector was to provide the training as the students were registered as external students. This was supplemented by web based course details, on-line course materials, quizzes, model papers and answers and also by a weekly IT program. This was an excellent example of Public-Private Partnership with over 50 private educational institutions preparing students for the BIT. Under the able leadership of Professor V. K. Samaranayake, the DCS and the ICT were able to negotiate substantial donor funding for human resource development and infrastructure from the Swedish government under the Sida/IT program in 2000. In the meantime the DCS and the ICT also received substantial equipment and postgraduate funding from ADB and JICA under Government of Sri Lanka auspices.

In the September of 2002, the Institute of Computer Technology and the Department of Computer Science of the Faculty of Science merged to form the University Of Colombo School of Computing, or UCSC, as a centre of Higher Learning affiliated to the University of Colombo with a fair amount of financial and administrative autonomy. This merger helped to bring together over 20 academic staff with 15 PhDs and a large number of postgraduate qualified IT specialists and also all the resources of the two institutions under a single entity. The UCSC has three academic departments and five centres. It is now enjoying the status of being the best IT centre for higher education in the country**.** UCSC commenced its undergraduate teaching in earnest with a three year Bachelor of Computer Science degree, a four year Bachelor of Science in Computer Science degree with intake of 240 students annually, in addition to continuing its ongoing special degree in Computer Science with the Faculty of Science. Under the MOU signed with the Faculty of Science a very close collaboration was envisaged with UCSC staff involved in teaching faculty students for Joint Honours degree programs with a substantial computing component. The BIT degree continued as usual with an annual intake of around 2500 (http://www.bit.lk) and three new Masters programs namely in, Computer Science, Information Technology and Advanced Computing commenced in October 2002 catering to around 200 part time students. In 2002, UCSC was successful in obtaining a large JICA grant – amounting to 390 million Japanese Yen – to establish ad Advanced Digital Media Technology Centre (ADMTC) equipped with multimedia laboratories, a professional Digital Recording Studio and funding for collaborative research project with leading Japanese universities to build the capacity in the field of web based teaching. The established tradition of third country training programs continued with JICA support and the last multimedia development course under the current funding phase concluded this entry.

The Sida/IT project which commenced in 2000 was extended to two phases and concluded in August 2010. The Sida/IT project was unique in the way it opened PhD opportunities to computing academic staff from all state universities in Sri Lanka. Staff of universities from Jaffna, Peradeniya, Ruhuna and UCSC obtained PhD and M.Phil qualifications from Uppsala, Stockholm, KTH, Mid Sweden, Gottenburg and Halmstad universities in Sweden enriching the local university human resources. This also spurred staff exchanges between local counterpart universities and those in Sweden. Under the infrastructure development, the LEARN network connecting all universities in Sri Lanka was strengthened with campus-wide fibre networks and wide band access to internet. In all this, UCSC played a key role in coordinating all activities and providing the necessary technical support.

Asia eBIT was a project that was started in the first quarter of 2005 to improve the quality of the BIT program through e-learning services and it was funded by the European Union. It was a three year project which had two foreign partners, the Department of Computer and Systems Science (DSV), the Royal Institute of Technology (KTH - Kungliga Tekniska högskolan), Stockholm University in Sweden and Delft

Technology University in Netherland and which also provided for staff exchange and PhD opportunities for local staff. Highly motivated staff were also able to secure many foreign grants for specialized research in the areas of grid computing (SPIDER), information security (SPIDER, ISIF) and language processing (IDRC).

In 2002, UCSC commenced another undergraduate degree program carefully taking into consideration the needs of the industry; this was called the Bachelor of Science in ICT, both a three year and a four year degree – in effect an Information Systems degree. The UCSC continued to evolve its postgraduate studies and research capabilities, redesigning the masters programs into Masters in Computer Science, Masters in IT and Master of Science in Information Systems Security, and registering a number of M.Phil and PhD students while utilizing the generated funds to enhance staff research capabilities. UCSC commenced its own journal publication in 2008, called ICTER, the International Journal on ICT for Emerging Regions, which is widely circulated around the world, and is moderated by an international advisory panel. At the same time, UCSC in its pioneering spirit formed a company called Theekshana to work towards national development goals in e-governance and other areas, by utilizing the in house expertise of is academic staff in strengthening industry-university collaboration.

UCSC was proved and announced at this stage that responsibility was taken to conduct the fully automated on-line training programs as part of the pre-orientation program, which was a tailor made program for 22,500 New Entrants to Universities in order to improve English Language and Information Technology skills. This has been conducted for a period of three months from July to September 2011 before they enter the University in October. The courses and the on-line tests, (both pre and post) was developed by UCSC and conducted through the NODES – The National Online Distance Education Service of the Ministry of Higher Education. NODES have access to a fibre network of 26 state-of-the-art NODES Access Centres island-wide to complete the above tasks. These centres includes 660 high end multimedia PCs for individual use, video conferencing facilities to connect 20 centres simultaneously, scanning and printing services and are staffed with knowledgeable operators for onsite assistance to students. These centres are networked with high speed fibre communication line bandwidth to have minimum of 2 Mbps dedicated to centres, 45 Mbps dedicated access to Network Operating Centre and 10 Mbps dedicated access to Internet. As noted elsewhere, apart from the academic departments, UCSC also has a number of centres, ranging from professional development, external degrees, e-learning and network operations (serving the whole of the University of Colombo) to forensics.

18 Conclusion

Sri Lanka has throughout its recorded history given priority to human development and in particular towards education. This has resulted in a high quality of life even though pure economic indicators make the country one that is still developing. In the field of computing, which was recognized as important even in the late sixties, a strategy of sharing whatever knowledge one had without awaiting expensive resources has shown results. Another aspect was the policy of computers for all ages, professions and for the society at large. The development of Computer Education has been well planned taking into account not only the currently available technology but also future trends as envisaged by the policy makers. This included the provision of resources, both human and material, and the strategic and optimum utilization of limited donor assistance. These initial steps have quite rightly led to international recognition and regional collaboration.

Today, the UCSC continue to grow from strength to strength, providing beyond doubt, the benefit of the initial planning and positive approach of the successive development phases of its predecessor institutions and their members. Its stature as a centre of international repute and success is also a strength to the many donors who have assisted in the early development efforts who can now see positive results from their investment in development.

With the demise of its founding director and mentor, Professor V. K. Samaranayake in 2007, an era came to an end confirming one of the nature's certainties. However, in the true spirit of this giant man in the national Information Technology landscape, and the institutions and the human resources base he was so dedicated in building up, the UCSC and of its predecessor institutions and their members shall continue its ground breaking activities beyond its 45 years of hard work and looks forward with pride and eagerness to continue its mission for the nation and its citizens.

Evolution of Computer Education in Spain: From Early Times to the Implementation of the Bologna Agreement

Ramon Puigjaner

Universitat de les Illes Balears
putxi@uib.cat

Abstract. This paper intends to present a short overview of the evolution of computer education in Spain since the initial teaching in this domain to the current works to adapt it to the European Higher Education Space (EHES) from the point of view of somebody that has been involved and has directly participated in most of the steps of this evolution.

Keywords: Computer science education, Spain, Bologna agreement.

1 Remote Precursors

It is not here the correct place to talk about the precursors of computing machines but at least it is convenient to mention Ramon Llull (Raimundus Lullius) who in the XIII century invented several logical machines oriented to convert the Muslims to the, for him, truth religion, and Leonardo Torres Quevedo and Esteve Terrades who in late XIX and early XX centuries built several analog machines to solve complicated analytical calculations.

2 Early University Courses: Decade of 1960s

At university level, just the *Universidad Complutense de Madrid* had an Automatic Computation speciality with some courses on basic computer architecture and on programming, common to the curricula of Mathematics and Physics. The Industrial Engineering Schools had a course on Computers which mainly explained the basic Von Neumann architecture and the FORTRAN language.

In March 1969, the Ministry of Education created in Madrid the *Instituto de Informatica*, [5] a strange organization, without any contact with the university and following a strange curriculum: the students earned a different title after each one of the five years of studies after secondary education. With these degrees it was intended that people who earned them were ready to develop professional tasks in industry and companies. It is easy to understand the difficulties of simultaneously giving a solid background and the practical knowledge associated to each degree.

In October 1967 was created in Barcelona the *Asociación de Técnicos de Informática* that developed an important educational task (mainly courses on data

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 143–165, 2012.

structures, basic computer organization, programming languages, operating systems, etc.) oriented to give a computer science background to people working with computers at that time without any formal education in computer science.

3 Computer Science Arrives to the University: Decade of 1970

The *Instituto de Informática* started its regular courses in 1970 according the above commented curriculum. It created a delegation in Donostia in 1971 and in 1972 the *Universitat Autònoma de Barcelona* created a Department of Informatics in its Faculty of Sciences. This Faculty was obliged to follow the same curriculum of the *Instituto de Informática*. The team that started to teach Computer Science in this University (I belonged to it) was a mixture of good professionals and people having followed some university computer courses, mainly in France (Paris and Grenoble); they kept the titles of the official curriculum but they tried to transform the contents into a more reasonable structure according to the university spirit.

In 1974, the Spanish Ministry of Education considered that Informatics had to be included in the regular university structure. A commission was created to study how to pass the *Instituto de Informática* and its satellites to the university. Several universities fought in this commission to get computer studies. Finally by the end of 1975 it was decided that three Faculties of Informatics had to be created: Barcelona (in the *Universitat Politècnica de Catalunya*), Donostia (in the *Euskal Herriko Unibertsitatea*) and Madrid (in the *Universidad Politécnica de Madrid*), and that the previous institutions giving informatics studies had to stop to teach informatics [6]. This was true in Madrid and Donostia because the *Instituto de Informática* and its delegation were incorporated in the corresponding universities and their denomination changed. In Barcelona the situation was more complicated because it was necessary to passing studies from one university to another (unbelievable in Spain at that time). Finally, both universities kept their studies. The new Faculties started to work in October 1977 with a five year curriculum that, for the first time in Spain, was different for each university. In addition, in the Faculty of Barcelona the classical curriculum structure of a set of courses per academic year was broken and the curriculum was organized by courses with their corresponding pre-requisites in such a way that the student was able to organize his/her own curriculum choosing courses among those offered by the Faculty but respecting some compulsory courses [7]. A draft version of this curricula can be found in [13]. In all cases curricula were planned for five years of studies. Curricula of the Madrid and Donostia Faculties followed the traditional structure. In all cases, but especially in the Faculty of Barcelona, curricula were inspired by the ACM Computer Curriculum 1968 [1]. The students that successfully followed the studies in one of these universities obtained the title of *Licenciado en Informática* (Licensed in Informatics).

For its novelty at that moment, the structure of the Faculty of Barcelona and its curriculum follows. The Faculty was organized around 8 departments: Mathematics, Theoretical Computer Science, Computer Programming, Computer Architecture, Physical Systems, Automatic and Hybrid Systems, Statistics, and Information Systems.

Students arriving to the Faculty had a first year with the following courses:

- Computers and programming
- Algebra
- Mathematical Analysis I (Infinitesimal calculus)
- Representation techniques
- Physics

To get the first cycle of this License it was compulsory to succeed in the following courses:

- Mathematical analysis II
- Computer structure
- Information structure
- Programming technology
- Statistics
- Operating systems

Each course was assigned a number of credits (1 credit was 1 hour of course per week during an academic year) and to earn the first cycle a student should get courses for an amount of 75 credits and to earn the second cycle (*Licenciado en informática*) a student should get courses for an amount of 50 supplementary credits.

Each course had some pre-requisites, could be compulsory (C) or optional (O) and be valid for just the first cycle (F) or for both (B). A (C) beside some pre-requisite means that both courses can be followed in parallel.

Department of Mathematics

Course	Acronym	Credits	Pre-req.	Class	Validity
Algebra	AL	5	None	C	F
Mathematical analysis I	AM-1	5	None	C	F
Mathematical analysis II	AM-2	6	AN-1	C	F
Numerical calculus	CN	4	AL, AN-2, TR	O	F
Representation techniques	TR	3	None	C	F
Numerical analysis	AN	4	CN	O	B
Information and coding theory	TIC	4	AF, E	O	B

Department of Theoretical Computer Science

Course	Acronym	Credits	Pre-req.	Class	Validity
Finite automata	AF	6	AL, E	O	B
Computability theory	TC	4	AF	O	B
Language theory	TL	4	C, AF	O	B
Mathematical logic	LM	4	AL, CP	O	B
Artificial intelligence	AI	4	LM, TL, AD	O	B

Department of Computer Programming

Course	Acronym	Credits	Pre-req.	Class	Validity
Computers and programming	CP	7	None	C	F
Information structure	EI	4	CP	C	B
Programming languages	LP	4	AL, CP	O	B
Programming technology	TP	4	AL, CP	C	B
Compilers	C	4	EI, LP	O	B
Files and data bases	FBD	5	EI, LP	O	B

Department of Computer Architecture

Course	Acronym	Credits	Pre-req.	Class	Validity
Computer structure	EC	4	CP, AL	C	F
Operating systems	SO	4	EC, EI, TP	C	B
Computer architecture	AC	4	EC, AF	O	B
Design and evaluation of configurations	DAC	4	AC, SI	O	B
Communications and computer networks	CRC	4	SO, AC, SI	O	B
Operating systems design	DSO	4	SO, AC	O	B
Diagnostic and reliability	DF	4	AC, TIC	O	B

Department of Physical Systems

Course	Acronym	Credits	Pre-req.	Class	Validity
Physics	F	5	None	C	F
Electronics	EL	4	F, AN-1	O	F
Digital circuits	CD	4	EL, AF	O	B
Design of computers	DC	4	CD, AC, DAH	O	B
Peripheral equipments	EP	4	CD	O	B
Analogical and hybrid design	DAH	4	EL	O	B

Department of Statistics

Course	Acronym	Credits	Pre-req.	Class	Validity
Statistics	E	5	AL, AN-2	C	F
Simulation	SI	4	E	O	B
Data analysis	AD	3	E	O	B
Stochastic processes	PE	4	E	O	B
Optimization	O	5	AN-2, AL, E (c)	O	B
Optimization algorithms	AO	4	E, O	O	B
Operational research	OR	4	E, O	O	B

Department of Automatic and Hybrid Systems

Course	Acronym	Credits	Pre-req.	Class	Validity
Systems and signals	SS	4	AN-2, ES (c)	O	B
System dynamics	DS	4	SS	O	B
Optimal control and filtering	COF	4	DS, PE	O	B
Analogical and hybrid calculus	CAH	4	AN-2, LP, EL	O	B
Architecture and design of control systems	ADSM	4	SS, CAH	O	B
Real-time operating systems	SOTR	4	SO, SS	O	B

Department of Information Systems

Course	Acronym	Credits	Pre-req.	Class	Validity
Economy	ECO	3	None	O	F
Design and utilization of files and data bases	DUABD	3	EI, DT (c)	O	B
Technological design	TP	4	TP DUABD (c), SO (c)	O	B
Logical design of information and decision systems	DL	5	DT, DUABD, EO	O	B
Project methodology	MP	5	DL, TO	O	B
Organization structures	EO	3	ECO	O	B
Organization techniques	TO	3	EO	O	B
Organization administration	AO	4	EO	O	B
Computer centre management	GSI	2	DL, TO	O	B
Group dynamics	DG	2	TO, AO	O	B
Law	DE	2	AO	O	B

4 General Restructuring of University Studies: Decade of 1980s

Around 1980 a new three years study in Informatics was created and started in Madrid (*Universidad Politécnica de Madrid*) and Valencia (*Universitat Politècnica de València*). The students that successfully followed these studies obtained the title of *Diplomado en Informática* (Diplomate in Informatics). Some after the *Universidad de Las Palmas de Gran Canaria* and the *Univeristat de les Illes Balears* (Palma de Mallorca) created three year studies in Informatics. When these studies in Valencia, Las Palmas and Palma de Mallorca reached the third year, extensions to five years were implemented in Valencia, Las Palmas and Palma de Mallorca. These three years studies had two orientations or intensifications:

- Computer systems, mainly devoted to a vision of the computer under the user interface
- Business management applications

In the last years of this decade the Spanish Ministry of Education started a general reorganization of all university studies creating a catalogue of official titles (those delivered by the Ministry itself based on the studies done in some university) reorganizing the existing titles and creating new ones. To get official acceptance by the Ministry each one of these titles had to respect a set of contents established by the Ministry (main topics). To describe the relative importance of each of these topics a measure was invented: the credit equivalent to ten hours of teaching (including all kind of activities driven by the university teaching staff: theoretical classes, practical classes, etc.) received by the students. An academic year was estimated to have 30 weeks.

In the case of informatics three new titles were created and those existing till that moment disappeared:

- *Ingeniero en Informática* (Informatics Engineer): five years divided in two cycles and between 300 and 400 credits [8].
- *Ingeniero Técnico en Informática de Sistemas* (Technical Engineer in Informatics: Computer systems orientation): three years and between 180 and 225 credits [9].
- *Ingeniero Técnico en Informática de Gestión* (Technical Engineer in Informatics: Computer business management orientation): three years and between 180 and 225 credits [10].

The main contents of these new careers were:

Ingeniero en Informática

First cycle

Main topic and description	Credits
Statistics Descriptive statistics. Probabilities. Applied statistical methods.	6
Structure Data and Information Abstract data types. Data structures and manipulation Algorithms. Information structure: Files, Data bases.	12

Computer structure and technology Functional Units: Memory, Processor, Periphery, Machine and Assembly Languages, Functional schema. Electronics. Digital Systems. Peripheral devices.	15
Informatics physical fundamentals Electromagnetism. Solid state. Circuits.	6
Informatics mathematical fundamentals Algebra. Mathematical Analysis. Discrete Mathematics. Numerical Methods.	18
Programming Methodology and Technology Algorithms design. Algorithms analysis. Programming Languages. Programmes design: Modular decomposition and documentation. Programmes verification and testing techniques.	15
Operating Systems Operating systems organization, structure and services. Memory and processes management and administration. Input/output management. File systems.	6
Automata and Formal Languages Theory Sequential machines and finite automata. Turing machines. Recursive Functions. Formals grammars and Languages. Neuronal networks.	9

Second cycle

Main topic and description	Credits
Computer architecture and engineering Parallel architectures. Architectures oriented to applications and languages.	9
Software engineering Requirements analysis and definition. Software design, properties and maintenance. Configuration management. Planning and management of informatics projects. Applications analysis.	18
Artificial Intelligence and knowledge engineering Heuristics. Knowledge based systems. Learning. Perception.	9
Language Processors Compilers, translators and Interpreters. Compiling phases. Code optimization. Macroprocessors.	9
Networks Networks Architecture. Communications.	9
Informatics Systems Analysis methodology. Informatics systems configuration, design, management and evaluation. Informatics systems environments. Advanced technologies of information systems, data bases and operating systems. Projects of informatics systems.	15

Ingeniero Técnico en Informática de Sistemas

Main topic and description	Credits
Statistics Descriptive statistics. Probabilities. Applied statistical methods.	6
Structure Data and Information Abstract data types. Data structures and manipulation algorithms. Information structure: Files, data bases.	12
Computer structure and technology Functional units: Memory, processor, periphery, machine and assembly languages, Functional schema. Electronics. Digital systems. Peripheral devices.	15
Informatics physical fundamentals Electromagnetism. Solid state. Circuits.	6
Informatics mathematical fundamentals Algebra. Mathematical analysis. Discrete mathematics. Numerical methods.	18
Programming Methodology and Technology Algorithms design. Algorithms analysis. Programming languages. Programmes design: Modular decomposition and documentation. Programmes verification and testing techniques.	12
Networks Networks Architecture. Communications.	6

Operating Systems Operating systems organization, structure and services. Memory and processes management and administration. Input/output management. File systems.	6
Automata and Formal Language Theory Sequential machines and finite automata. Turing machines. Recursive functions. Formal grammars and languages. Neuronal networks.	9

Ingeniero Técnico en Informática de Gestión

Main topic and description	Credits
Statistics Descriptive statistics. Probabilities. Applied statistical methods.	9
Structure Data and Information Abstract data types. Data structures and manipulation algorithms. Information structure: Files, data bases.	12
Computer structure and technology Functional units: Memory, processor, periphery, machine and assembly languages, Functional schema. Electronics. Digital systems. Peripheral devices.	9
Business management software engineering Business management software design, properties and management. Planning and management of informatics projects. Analysis of management application.	12
Informatics mathematical fundamentals Algebra. Mathematical analysis. Discrete mathematics. Numerical methods.	18
Programming Methodology and Technology Algorithms design. Algorithms analysis. Programming languages. Programmes design: Modular decomposition and documentation. Programmes verification and testing techniques.	15
Operating Systems Operating systems organization, structure and services. Memory and processes management and administration. Input/output management. File systems.	6
Organization techniques and business management Economic system and business. Administration and accounting techniques.	12

Also it was stated that the students having earned one of these three years degrees were allowed to follow the second cycle of *Ingeniero en Informática*.

5 Set Up of the New Careers: Decade of 1990s

In the early years of this decade all universities giving the old degrees in informatics adapted their curriculum to the new characteristics. This adaptation took different solutions:

- Universities that delivered the three degrees separately: the five years degree in a faculty or school and the three year degree in a different school.
- Universities that delivered the three degrees in the same faculty or school with a complete implementation of the three degrees.
- Universities that delivered the three degrees in the same faculty or school but without the implementation of the first cycle of *Ingeniero en Informática* and using both three years degrees as the first cycle.
- Universities that had just the five years degree.
- Universities that had one or both three years degrees.

However, soon several problems appeared:

- The fact that three different first cycles gave access to the second cycle introduced difficulties in different topics like networks, computer architecture and software engineering. The reasons were that sometimes the same main topic with the same descriptors had assigned a different number of credits or that the student coming from some first cycle had previous knowledge of some topic not known for the students coming from other first cycles.
- The low number of credits assigned to operating systems obliged most universities to create supplementary courses in this topic.
- The growing importance of networking. It was remarked that it was possible that an *Ingeniero Técnico en Informática de Gestión* could earn his/her title with no knowledge of networking.
- The inconvenience of having Automata theory as a compulsory topic in the first cycle of *Ingeniero en Informática* (too theoretical for beginners) and in *Ingeniero Técnico en Informática de Sistemas* (too theoretical for the applied orientation of the three years studies).

During this period the number of faculties and schools delivering these degrees was continuously increasing (and currently there are approximately 80 in Spain). This fact and the need for exchanging information about experiences and discussing the difficulties in the implementation of their curricula provoked the need of discussion meetings with the participation of all faculties and schools teaching informatics careers. These annual meetings started in 1995. However, as the number of schools and faculties was continuously increasing, in 1998 it was decided to set up a minimal organization with a president and a secretary and a title: *Conferencia de Decanos y Directores de Centros Universitarios de Informática*, CODDI (Conference of Deans and Directors of Informatics University Centres). The first task assigned just after the appointment of a president was the review of the main topics of the informatics careers in order to correct the detected inconveniences. In 1999 this task was completed and the result was [4]:

Ingeniero en Informática

First cycle

Main topic and description	Credits
Algebra and discrete mathematics Basic algebraic structures. Lineal algebra. Combinatory. Discrete structures: graphs, trees. Logic. Coding. Numerical applications.	12
Mathematical analysis Successions and series. Integration. Differential equations. Numerical applications.	6
Data bases Data models. Data base management systems.	6
Statistics Probabilities. Applied statistical methods. Statistical inference.	6
Computer structure Functional units: memory, processor, input/output. Machine and assembly languages. Running schema. Microprogramming.	12

Informatics physical fundamentals Electromagnetism. Electronics. Circuits	6
Software engineering fundamentals Software systems analysis and design. Software properties and maintenance. User interfaces.	6
Programming and data structure Algorithms design and analysis. Programming paradigms and languages. Basic techniques of programmes design, verification and testing. Object oriented programming. Abstract data types. Data structures and manipulation algorithms.	21
Computer networks Communication elements and systems. Hierarchical structure of networks. Usual types of networks: local area networks and wide area networks. Network interconnection. Security.	6
Operating systems Operating systems organization, structure and service. Memory, processes and resources management and administration. Input/output management. File systems.	9
Computer technology Electronic components and systems of computers. Digital Systems. Microprocessors. Peripherals structure and functioning.	6

Second cycle

Main topic and description	Credits
Computer architecture Speed increasing techniques. Parallel architectures.	9
Automata theory, formal languages and language processors Sequential machines and finite automata. Turing machines. Complexity theory. Recursive functions. Formal grammars and languages. Compilers. Translators and interpreters. Compilation phases. Macroprocessors.	15
Software Engineering Requirements analysis and definition. Software properties and maintenance. Software quality assurance. Software projects planning and management. Methodologies. Human-machine interfaces.	15
Artificial intelligence Heuristics. Knowledge representation techniques. Knowledge based systems. Perception.	6
Networks and distributed systems Network configuration, administration and management. Interconnection. High performance networks. Quality of service. Security. Information compressing. Distributed systems.	9
Informatics systems Analysis methodology. Informatics systems configuration, design, management and evaluation. Informatics systems environments. Advanced technologies of information systems, data bases and operating systems. Projects of informatics systems. Audit. Security.	12

Ingeniero Técnico en Informática de Sistemas

Main topic and description	Credits
Algebra and discrete mathematics Basic algebraic structures. Lineal algebra. Combinatory. Discrete structures: graphs, trees. Logic. Coding. Numerical applications.	12
Mathematical analysis Successions and series. Integration. Differential equations. Numerical applications.	6
Data bases Data models. Data base management systems.	6
Statistics Probabilities. Applied statistical methods. Statistical inference.	6
Computer structure Functional units: memory, processor, input/output. Machine and assembly languages. Running schema. Microprogramming.	12

Informatics physical fundamentals Electromagnetism. Electronics. Circuits	6
Software engineering fundamentals Software systems analysis and design. Software properties and maintenance. User interfaces.	6
Programming and data structure Algorithms design and analysis. Programming paradigms and languages. Basic techniques of programmes design, verification and testing. Object oriented programming. Abstract data types. Data structures and manipulation algorithms.	21
Computer networks Communication elements and systems. Hierarchical structure of networks. Usual types of networks: local area networks and wide area networks. Network interconnection. Security.	6
Operating systems Operating systems organization, structure and service. Memory, processes and resources management and administration. Input/output management. File systems.	9
Computer technology Electronic components and systems of computers. Digital Systems. Microprocessors. Peripherals structure and functioning.	6
Informatics systems Management, planning and development of computer systems projects.	6

Ingeniero Técnico en Informática de Gestión

Main topic and description	Credits
Algebra and discrete mathematics Basic algebraic structures. Lineal algebra. Combinatory. Discrete structures: graphs, trees. Logic. Coding. Numerical applications.	12
Mathematical analysis Successions and series. Integration. Differential equations. Numerical applications.	6
Data bases Data models. Data base management systems.	6
Statistics Probabilities. Applied statistical methods. Statistical inference.	6
Computer structure Functional units: memory, processor, input/output. Machine and assembly languages. Running schema. Microprogramming.	12
Software engineering fundamentals Software systems analysis and design. Software properties and maintenance. User interfaces.	6
Programming and data structure Algorithms design and analysis. Programming paradigms and languages. Basic techniques of programmes design, verification and testing. Object oriented programming. Abstract data types. Data structures and manipulation algorithms.	21
Computer networks Communication elements and systems. Hierarchical structure of networks. Usual types of networks: local area networks and wide area networks. Network interconnection. Security.	6
Operating systems Operating systems organization, structure and service. Memory, processes and resources management and administration. Input/output management. File systems.	9
Informatics systems Management, planning and development of computer business application projects.	6
Business structure and functions Business as a system. Management and administration techniques. Organization structures.	6
Information systems Evaluation and management of information systems development. Strategic planning of information technologies and systems de. Applications.	6

This new definition of the main topics for the three degrees corrected the main defaults of the previous one:

- The same topic had the same description and the same number of credits.
- Networking was compulsory in the three first cycles.
- Two theoretical topics disappeared from the first cycles.
- Mathematics was split between Algebra and Analysis.
- Operating systems had a greater number of compulsory credits.

CODDI submitted this proposal to the Ministry of Education. However it was not accepted because its acceptance would have allowed other careers also to request the modification of their compulsory main topics. And this would have introduced a high degree of discussion between universities and between these and the Ministry. Nevertheless it was accepted as guidelines for the analysis and acceptation of future curricula submitted to the Ministry by the universities.

6 Towards the European Higher Education Space (EHES): Decade of 2000s

Coincidental with the change of millennium, the European Union decided to ask the member states to reorganize their university systems in such a way that a convergence was reached around 2010 in two main aspects:

- University studies should be organized in three levels: bachelor, master and doctorate.
- University studies should define for each course the effort required of the student. The European Credit Transfer System (ECTS), equivalent to approximately 25 to 30 hours of work for the student including all his/her activities (theoretical courses, practical courses, seminars, personal study, etc.).

This convergence was named as the Bologna process because the agreement of all the member states was reached in a meeting held in the city where the first European university was created.

In 2001 CODDI started to work on how to adapt informatics studies to this convergence process. Initially a set of considerations showing mismatches in either the university studies structure or in the consideration by society of the degrees delivered by the university [2]. These considerations were:

- It was observed that neither the market nor the universities had succeeded to clearly discern the professional and educational differences between the *Ingenierías Técnicas en Informática* and the *Ingeniería en Informática* due to the constant evolution of informatics and professional changes.
- The difficulties found at the second cycle of the *Ingeniería en Informática* due to coexistence of students coming from three different first cycles.
- The fact that the *Ingenieros Técnicos* did not have professional acknowledgement at European level as university graduates.
- The difficulty for defining competencies and responsibilities of the informatics professionals.

- The great number of new activities with a fuzzy limit with other engineering branches (telecommunication engineers, industrial engineers, etc.).
- The consequences of the effort done to offer a higher non-university education, public as well as private, obliged to reconsider the structure, contents and level of informatics university studies.

The main conclusions of these discussions were [3]:

- An initial premise said that the market would experience a strong growth of demand in a near future. The strategic presence of informatics suggested the need of a set of solutions considering all education levels (primary education, secondary education, vocational education) and not only the university level.
- In this sense deep consideration about the University planning had to be done to take into account the geographical distribution and the resources needed to attain the planned objectives.
- The university structure would have to be organized in two cycles: *Grado* (degree) and Master.
- There had to be just a unique title of *Grado* whose name would be *Ingeniería en Informática* (Informatics Engineering).
- The title of *Ingeniero en Informática* would furnish full professional competencies for the exercise of the profession.
- The education furnished at the *Grado* level would be general in the informatics domain.
- The *Grado* studies would have 240 ECTS credits organized en 4 years.
- Among the fundamental educational contents of the *Grado*, there had to be an End of Studies Project, that would integrate the knowledge acquired by the student during his/her studies and that would make an approximation to real professional cases as well as to transversal contents that would put in evidence his/her abilities for the exercise of engineering activities.
- The common educational contents of the *Grado* would represent about 60% of the total study load, including the End of Studies Project, leaving 40% for topics to be freely decided by each university.
- Among the courses to be freely decided by the universities, it was recommended to have a large enough offer of courses oriented to give the students a solid knowledge of current informatics technologies as well as application domains.
- The Master would have as objective the professional specialization of the *Ingeniero en Informática* or his/her preparation for the research.
- The number of Master degrees would be large enough to cover the demand of specialized professionals at every moment.
- Master studies would have between 60 and 120 ECTS credits, depending on the previous degree earned and would include some effort allocated to a Master Thesis
- The Master degree would allow access to the preparation of a doctoral thesis to obtain the Doctor degree.

The structure of the studies is represented in the following table.

Contents			Min.	Max	
		Scientific base	10%	15%	Informatics mathematical fundamentals. Informatics physical fundamentals.
Common educational contents	60%	Informatics engineering specific contents	35%	40%	Programming. Software engineering. Information systems engineering. Intelligent systems engineering. Operating systems. Distributed systems and networking. Computer engineering.
		General contents of engineering	5%	10%	Business management. Ethical, legal and professional aspects. Professional abilities.
		End of studies project	6%	6%	
Contents freely decided by the university	40%				
Total effort		240 ECTS credits			

The students that earn the *Grado en Ingeniería en Informática* would be characterized by the following:

- To be prepared to exercise his/her profession, having a clear knowledge of the human, economic, social, legal and ethical dimensions.
- To be prepared to assume responsible tasks in any kind of organization along his/her professional life, in technical as well as in managerial positions, and to contribute in information and knowledge management.
- To have the required abilities in the professional practice of engineering: to be able to manage projects, to communicate in a clear and effective way, to work in a multidisciplinary team as well as to manage it, to adapt himself/herself to the changes and to autonomously learn along with his/her life.
- To be prepared to learn and to use in an effective way techniques and tools that could appear in the future. This versatility had to be especially valuable in organizations in which permanent innovation was needed.
- To be able to specify, design, build, verify, audit, evaluate and maintain informatics systems giving answers to user needs.
- To have the basic education to be able to continue his/her studies of Master and Doctorate in Spain or elsewhere[1].

[1] Up to this point, with very small differences this article is a copy of the paper presented at the conference on History of Computer and Education, HCE3, in the frame of the IFIP World Computer Congress WCC 2008 [14].

7 Implementation and Follow Up of the EHES: Decade of 2010

7.1 Implementation

Several changes in the Spanish Ministry of Education delayed the implementation of the EHES. Finally in 2007 the framework for the implementation of the EHES was set up [11]. Some points were clear but the framework was not yet complete:

- There would not be a catalogue of official titles; each university had to propose its own titles that would be validated by an independent agency (*Agencia Nacional de Evaluación de la Calidad del Sistema Universitario* – ANECA, National Agency for the Evaluation of the Quality of the University System) that would evaluate the appropriateness of the proposed title (specially avoiding confusion to society), the quality of the proposal and the existence of a sufficient amount of human and material resources allocated to the correct implementation.
- The *Grado*, in our case of *Ingeniero en Informática*, would have 240 ECTS that would include the end of studies project.
- The Master degree would have 60 or 120 ECTS depending on the coherence between the grade earned and the intended Master.

From this information it was easy to see that the proposal of the CODDI, several years before the decisions of the Ministry, was fully in line with the framework in which the universities would work in the near future.

Early in 2008 five commissions were set up to analyse the new curricula on Science, Engineering, Health, Law and Economics, and Letters and Humanities. Informatics was included in the engineering domain.

In order to understand the environment, it is necessary to know that all classical engineering studies (civil engineers, mining engineers, naval engineers, industrial engineers, agronomical engineers, forest engineers, aeronautical engineers and telecommunication engineers) have exclusive professional capabilities and are differentiated by branch and level (3 years or 5 years of studies) and associated to the academic diploma. This organization causes serious conflicts of interests in which are involved the corresponding professional associations. The informatics engineers have no defined professional capabilities and their professional space is frequently invaded by other engineering branches, mainly the industrial and telecommunication engineers.

Traditional engineering associations were against the decision of the government of associating professional capabilities to the *Grado* diploma when there were different capabilities for diplomas of 3 and 5 years. Before solving this fight, the verification process started in March 2008. This process was theoretically very well-conceived but looking more to how to teach than what should be taught. However, there were terms not clearly defined, related between them and critical for the correctness of this process, like: objectives, competencies, modules, matters, courses, contents, learning outcomes, etc.

Points to verify were:

- Title description
- Justification
  - Academic, scientific or professional interest
  - External references of the title
  - Consultation procedures
- Objectives
  - General and specific competencies
- Students access and admission
  - Information systems prior to registration and reception procedures and guidance for students newly admitted
  - Conditions or special admission tests
  - Support and guidance to students once enrolled
  - ECTS recognition and transfer
  - Course of adaptation
- Teaching planning
  - Knowledge structure
  - Student mobility
  - Description of modules, matters and courses
- Academic staff
  - Teaching staff and other human resources needed and available
  - Adequacy of faculty staff to support the curriculum
- Material resources and services
  - Available material resources
  - Planned material resources
- Planned results
  - Estimation of indicators
  - How to assess progress and learning outcomes
- Quality assurance system
  - Responsible for system quality assurance of curriculum
  - Procedures for assessing and improving the quality of teaching and faculty staff
  - Procedures to ensure the quality of internships and mobility programs
  - Procedures for analyzing the employment of graduates and satisfaction with the obtained training
  - Procedures for the analysis of the satisfaction of the different stakeholders and attention to suggestions and complaints. Criteria for the extinction of the title
- Implementation schedule
  - Implementation chronogram
  - Adaptation of the existing curricula students
  - Careers to be extinct

The verification commissions started to work with a formal description of its work but with an important lack of definitions (as explained before) and fundamental and logistic problems. Among the last ones an important difficulty was the lack of a good management system for the proposals. Due to the lack a good definition of its task, commission members used their better knowledge to the proposal analyses. Two points were taken especially into account:

- The name of the title should represent the delivered contents.
- The title description should not induce into error the future students reading the document with the intention of following these studies.

However, as classical engineering studies had associated professional capabilities supported by the corresponding professional associations, the need soon appeared to define the objectives and competencies of these studies. To solve this problem the Spanish Ministry of Education published in February 2009 these characteristics for all classical engineering studies. Nevertheless these definitions were not an example of well doing because:

- The objectives were defined in terms of competencies.
- The competencies were defined in terms of teaching contents.

This decision was against the previous definition in which the *Grado* gave the professional capabilities and the Master gave the specialization. With this decision the *Grado* gave the professional capabilities of the 3-years engineers in its concrete specialties and after obtaining the Master they got all the professional capabilities of 5-years engineers. This approach has the above mentioned inconvenience of passing from a specialized knowledge to a general one.

In the case of Informatics Engineering and some other newly created they had no professional capabilities. This fact represented to place them in a lower level than the classical engineers and the CODDI fight to have at least a definition of its studies similar to the other engineering studies even if they had not professional capabilities. This was obtained in August 2009 when the Ministry of Education published the definition of objectives and competencies for the *Grado* and Master in Informatics Engineering [12] with a better conception avoiding the definition of objectives in terms of competencies and the competencies in terms of teaching contents. This definition included the five specific technologies defined in the ACM/IEEE-CS Computing Curricula [14] in a similar way to the other classical engineers all of them had several specialties:

- Computer Engineering
- Software Engineering
- Computing
- Information Systems
- Information Technology

These specific technologies are different in their competencies but have the same objectives. The objectives are:

1. Ability to design, write, organize, plan, develop and sign projects in informatics engineering whose goal, according the knowledge acquired in agreement with the competencies, is the conception, development or exploitation of informatics systems, services and applications.
2. Ability to lead informatics activities projects in the field of computing according to the knowledge acquired in agreement with the competencies.

3. Ability to design, develop, evaluate and ensure accessibility, ergonomics, usability and safety of the systems, services and applications, as well as the management of information.
4. Ability to define, evaluate and select hardware and software platforms for the development and implementation of systems, services and applications, according to the knowledge acquired in agreement with the competencies.
5. Ability to design, develop and maintain computer systems, services and applications using the methods of software engineering as a tool for quality assurance according to the knowledge acquired in agreement with the competencies.
6. Ability to design and develop systems or centralized or distributed architectures integrating hardware, software and networks according to the knowledge acquired in agreement with the competencies.
7. Ability to recognize, understand and implement the necessary legislation during the development of the profession *Ingeniero Técnico en Informática* and use specifications, regulations and mandatory standards.
8. Knowledge of basic materials and technologies that enable learning and development of new methods and technologies as well as to equip them with great versatility to adapt to new situations.
9. Ability to solve problems with initiative, decision making, autonomy and creativity. Ability to communicate and transmit the knowledges, skills and abilities of the profession of *Ingeniero Técnico en Informática*.
10. Knowledge to perform measurements, calculations, assessments, appraisals, surveys, studies, reports, scheduling and similar computer works, according to the knowledge acquired in agreement with the competencies.
11. Ability to analyse and assess the social and environmental impact of technical solutions, understanding the ethical and professional responsibility of the activity of *Ingeniero Técnico en Informática*.
12. Knowledge and application of basic economic and human resource management, organization and planning of projects as well as legislation, regulation and standardization in the field of ITC projects, according to the knowledge acquired in agreement with the competencies.

The competencies structure is the following:

- Module of basic knowledge: 60 ECTS
  - Ability to solve mathematical problems that may arise in engineering. Ability to apply knowledge of: linear algebra, differential and integral calculus, numerical methods, numerical algorithms, statistics and optimization.
  - Understanding and mastery of basic concepts of fields and waves and electromagnetism, theory of electrical circuits, electronic circuits, semiconductor physical principle and logic families, electronic and photonic devices and their application to solving engineering problems.

  - Ability to understand and master the basic concepts of discrete mathematics, logics, algorithmics and computational complexity, and their application to solving engineering problems.
  - Basic knowledge of use and programming of computers, operating systems, databases and computer programs with applications in engineering.
  - Knowledge of the structure, organization, operation and interconnection of computer systems, the fundamentals of programming and its application to solving engineering problems.
  - Adequate knowledge of the company concept, institutional and legal framework of the company. Business organization and management.
- Module common to the informatics branch: 60 ECTS
  - Ability to solve mathematical problems that may arise in engineering. Ability to apply knowledge of: linear algebra, differential and integral calculus, numerical methods, numerical algorithms, statistics and optimization.
  - Ability to design, develop, select and evaluate computer applications and systems, ensuring their reliability, safety and quality, according to ethical principles and the current laws and regulations.
  - Ability to plan, design, deploy and manage computer projects, services and systems at all domains, leading its implementation and continuous improvement and assessing their economic and social impact.
  - Ability to understand the importance of negotiation, effective work habits, leadership and communication skills in all software development environments.
  - Ability to develop the technical specifications of a computer facility that meets the current standards and regulations.
  - Knowledge management and maintenance of computer systems, services and applications.
  - Knowledge and application of basic algorithmic procedures of information technologies to design solutions to problems by analysing the appropriateness and complexity of the proposed algorithms.
  - Knowledge, design and efficient use the data types and structures best suited to solving a problem.
  - Ability to analyse, design, build and maintain applications in a robust, secure and efficient way, by choosing the most appropriate paradigm and programming languages.
  - Ability to learn, understand and evaluate the computer's structure and architecture and the basic components that comprise them.
  - Knowledge of operating systems characteristics, functionality and structure and design and implement applications based on their services.
  - Knowledge and application of distributed systems, computer networks and Internet characteristics, functionality and structure, and design and implement applications based on them.

  - Knowledge and application of databases characteristics, functionality and structure that allow their proper use, and design and analyze and implement applications based on them.
  - Knowledge and application of tools for storage, processing and access to information systems, including those web-based.
  - Knowledge and application of fundamental principles and basic techniques of parallel, concurrent, distributed and real-time programming.
  - Knowledge and application of fundamental principles and basic techniques of intelligent systems and their practical application.
  - Knowledge and application of principles, methodologies and life cycles of software engineering.
  - Ability to design and evaluate human computer interfaces to ensure accessibility and usability of computer systems, services and applications.
  - Knowledge of the computer domain rules and regulations at national, European and international levels.
- Specific technology module: software engineering, 48 ECTS
  - Ability to develop, maintain and evaluate software systems and services that satisfy all user requirements and behave reliably and efficiently, be affordable to develop and maintain and meet quality standards, applying the theories, principles, methods and practices of Software Engineering.
  - Ability to assess client needs and specify the software requirements to meet these needs, reconciling conflicting objectives by finding acceptable compromises within limitations imposed by cost, time, and the existence of systems already developed and own organizations.
  - Ability to solve integration problems in terms of strategies, standards and available technologies.
  - Ability to identify and analyse problems and design, develop, implement, verify and document software solutions based on adequate knowledge of the current theories, models and techniques.
  - Ability to identify, evaluate and manage potential associated risks that might arise.
  - Ability to design appropriate solutions in one or more application domains using methods of software engineering that integrate ethical, social, legal and economic aspects.
- Specific technology module: computer engineering, 48 ECTS
  - Ability to design and build digital systems, including computers, microprocessor based systems and communications systems.
  - Ability to develop specific processors and embedded systems as well as to develop and optimize software for such systems.
  - Ability to analyse and evaluate computer architectures, including parallel and distributed platforms as well as to develop and optimize software for them.
  - Ability to design and implement system and communications software.
  - Ability to analyse, evaluate and select hardware and software platforms better suited to support embedded and real-time applications.

  - Ability to understand, implement and manage the systems security and safety.
  - Ability to analyse, evaluate, select and configure hardware platforms for the development and implementation of computer applications and services.
  - Ability to design, deploy, and manage computer networks.
- Specific technology module: computing, 48 ECTS
  - Ability to have a thorough understanding of the fundamental principles and models of computing and to know how to apply these to interpret, select, assess, model, and create new concepts, theories, practices and technological developments related to computer science.
  - Ability to know the theoretical foundations of programming languages and the lexical, syntactic and semantic processing techniques associated to them, and to know how to apply them for the language creation, design and processing.
  - Ability to evaluate the computational complexity of a problem, to know algorithmic strategies that can lead to its resolution and recommend, develop and implement one that ensures the best performance in accordance with the requirements.
  - Ability to learn the fundamentals, paradigms and techniques of intelligent systems and to analyse, design and build systems, services and applications that use these techniques in any application scope.
  - Ability to acquire, obtain, formalize and represent human knowledge in a computable way for problem solving using a computer system in any application scope, particularly those related to computing, perception and action aspects in smart environments.
  - Ability to develop and evaluate interactive systems and of presentation of complex information and its application to solving design problems of human-computer interaction.
  - Ability to recognize and develop computational learning techniques and design and implement computer systems and applications that use them, including those dedicated to extracting information and knowledge from large volumes of data.
- Specific technology module: information systems, 48 ECTS
  - Ability to integrate solutions of Information and Communications Technology and business processes to meet the information needs of organizations, enabling them to achieve their goals effectively and efficiently, giving them competitive advantages.
  - Ability to determine requirements for information and communication systems of an organization in response to safety issues and compliance with regulations and legislation.
  - Ability to actively participate in the specification, design, implementation and maintenance of information and communication systems.

  - Ability to understand and apply the principles and practices of organizations so that they can act as liaison between technical and management communities within an organization and actively participate in the training of users.
  - Ability to understand and apply the principles of risk assessment and correctly apply them in the development and implementation of action plans.
  - Ability to understand and apply the principles and techniques of quality and technological innovation management in organizations.
- Specific technology module: information systems, 48 ECTS
  - Ability to understand the organization environment and its needs in the field of information and communications technology.
  - Ability to select, design, deploy, integrate, evaluate, build, manage, operate and maintain hardware, software and networks technology, within appropriate parameters of cost and quality.
  - Ability to use user-centred methodologies and organization development, evaluation and management of applications and systems based on information technologies to ensure accessibility, ergonomics and usability of systems.
  - Ability to select, design, deploy, integrate and manage network and communications infrastructure in an organization.
  - Ability to select, deploy, integrate and manage information systems that meet the needs of the organization, with identified cost and quality criteria.
  - Ability to design systems, applications and services based on network technologies, including Internet, web, e-commerce, multimedia, interactive services and mobile computing.
  - Ability to understand, implement and manage the security and safety of computer systems.
- End of *Grado* project: 12 ECTS
  - Original exercise to be performed individually and presented and defended to a university jury, consisting of a professional nature project in the area of the specific technologies in Informatics Engineering in which the skills acquired in the teachings are synthesized and integrated.

The rest, up to 240 ECTS, is at disposal of the University for strengthening the above mentioned topics, for stages in companies and for collateral activities (participation in representative activities, sports, volunteer work for the community, etc.).

With the existing degree of freedom for defining the title of the *Grado*, two alternatives were presented to the Universities:

- To propose a unique name with several technological specialties (or professions with professional capabilities, if any)
- To propose one or several names related to the technological specialties (or professions with professional capabilities) they want to develop.

In favour of independent titles for each specific technology of professions with recognized capabilities they are to be aligned with these professions and use socially recognized names. Instead, for a unique title with technological specialities there are the limitations imposed by the local governments concerning the number of careers they were ready to fund, to be aligned with the White Book recommendations and to find social recognition.

In order to avoid misunderstandings by future students the criteria followed by the verification commission was to accept a specific name if it was adapted to a specific technology or profession with capabilities (e.g. Software Engineer) and a generic name (e.g. Informatics Engineer) in the case of including several specific technologies.

7.2 Follow Up

Two years after the implementation of the new studies, the ANECA has started a follow up procedure in order to have a look at how the Universities have implemented these new studies. The goal of this exercise is to assess the Universities about the points missing or not correctly implemented, especially with respect to the quality assurance procedures newly implemented by the university and to the new teaching orientation proposed by the Bologna agreement (instead of teaching the students, to learn them how to learn).

8 Conclusion

How will the future be? I think it will be necessary to wait ten years until we will be able to have an opinion about the success of this reform, mainly to evaluate how professors and students have changed their teaching and learning habits. This is, may be, the strongest challenge of this reform.

References

[1] ACM Curriculum Committee on Computer Science, Curriculum ‘68— Recommendations for Academic Programs in Computer Science. Comm. ACM 11(3), 151–197 (March1968)

[2] Campos, J., Casanovas, J., Colom, J.M., Martín, G., Martínez, J., Pont, A., Puigjaner, R., Robles, A., Sancho, M.R.: Informe sobre la adaptación de los estudios de TIC a la declaración de Bolonia. Unpublished document (For getting it, contact the author of this article) (2002)

[3] Casanovas, J., et al.: Libro Blanco sobre las titulaciones universitarias de informática en el nuevo espacio europeo de educación superior. ANECA (2004)

[4] Conferencia de Decanos y Directores de Centros Universitarios de Informática (CODDI): Nuevas directrices generales propias de los títulos de Ingeniero en Informática, de Ingeniero Técnico en Informática de Sistemas y de Ingeniero Técnico en Informática de Gestión. Unpublished document (for getting it, contact the author of this article) (1999)

[5] Ministerio de Educación y Ciencia (BOE de 14/04/1969 - Sección I): Decreto 554/1969, de 29 de marzo (Educación y Ciencia), por el que se crea un Instituto de Informática, dependiente del Ministerio de Educación y Ciencia, con sede en Madrid, y se regulan las enseñanzas del mismo
[6] Ministerio de Educación y Ciencia (BOE de 26/03/1976 - Sección I): Decreto 593/1976, de 4 de marzo, por el que se crean Facultades de Informática en Barcelona, Madrid y San Sebastián
[7] Ministerio de Educación y Ciencia (BOE de 27/07/1977 - Sección I): Orden de 7 de junio de 1977 por la que se aprueba el plan de estudios de la Facultad de Informática de Barcelona
[8] Ministerio de Educación y Ciencia (BOE de 20/11/1990 - Sección I): Real decreto 1459/1990, de 26 de octubre, por el que se establece el título universitario oficial de Ingeniero en Informática y las directrices generales propias de los planes de estudios conducentes a la obtención de aquél
[9] Ministerio de Educación y Ciencia (BOE de 20/11/1990 - Sección I): Real decreto 1459/1990, de 26 de octubre, por el que se establece el título universitario oficial de Ingeniero Técnico en Informática de Sistemas y las directrices generales propias de los planes de estudios conducentes a la obtención de aquél
[10] Ministerio de Educación y Ciencia (BOE de 20/11/1990 - Sección I): Real decreto 1459/1990, de 26 de octubre, por el que se establece el título universitario oficial de Ingeniero Técnico en Informática de Gestión y las directrices generales propias de los planes de estudios conducentes a la obtención de aquél
[11] Ministerio de Educación y Ciencia (BOE de 30/10/2007 - Sección I): Real decreto 1393/2007, de 29 de octubre, por el que se establece la ordenación de las enseñanzas universitarias oficiales
[12] Ministerio de Educación y Ciencia (BOE de 4/08/2009 - Sección III): Resolución de 8 de junio de 2009, de la Secretaría General de Universidades, por la que se da publicidad al Acuerdo del Consejo de Universidades, por el que se establecen recomendaciones para la propuesta por las universidades de memorias de solicitud de títulos oficiales en los ámbitos de la Ingeniería Informática, Ingeniería Técnica Informática e Ingeniería Química
[13] Puigjaner, R., Vergés, M.: Bases para un plan de estudios. Novatica (5), 5–12 (September-October 1975) ISSN: 0211-2124
[14] Puigjaner, R.: Computer Education in Spain: from Early to Recent Times. In: IFIP 20th World Computer Congress, Proceedings of the Third IFIP Conference on the History of Computing and Education WG 9.7/TC9, History of Computing, Milano, Italy, pp. 79–98 (September 2008) ISBN: 978-0-387-09656-8

The History of Computer Language Selection

Kevin R. Parker[1] and Bill Davey[2]

[1] College of Business, Idaho State University, Pocatello, Idaho USA
parkerkr@isu.edu
[2] School of Business Information Technology, RMIT University, Melbourne, Australia
billd@rmit.edu.au

Abstract. This examines the history of computer language choice for both industry use and university programming courses. The study considers events in two developed countries and reveals themes that may be common in the language selection history of other developed nations. History shows a set of recurring problems for those involved in choosing languages. This study shows that those involved in the selection process can be informed by history when making those decisions.

Keywords: selection of programming languages, pragmatic approach to selection, pedagogical approach to selection.

1 Introduction

The history of computing is often expressed in terms of significant hardware developments. Both the United States and Australia made early contributions in computing. Many trace the dawn of the history of programmable computers to Eckert and Mauchly's departure from the ENIAC project to start the Eckert-Mauchly Computer Corporation. In Australia, the history of programmable computers starts with CSIRAC, the fourth programmable computer in the world that ran its first test program in 1949. This computer, manufactured by the government science organization (CSIRO), was used into the 1960s as a working machine at the University of Melbourne and still exists as a complete unit at the Museum of Victoria in Melbourne. Australia's early entry into computing makes a comparison with the United States interesting.

These early computers needed programmers, that is, people with the expertise to convert a problem into a mathematical representation directly executable by the computer. The earliest history of programming languages was not of selection but of invention. Groups would construct a computer with a means of programming in mind. To simplify the programming process, various methods of helping humans cope with the demands of digital devices created the need for shortcuts and these became the first programming languages. The first programmers were mostly mathematicians or engineers who programmed in machine code of some form. Many of them used hardwiring to achieve their ends. These early programmers had no formal education in machine language programming.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 166–179, 2012.

The Computer History Museum (http://www.computerhistory.org/) provides a timeline for the creation of early languages, which is shown in Table 1.

Table 1. Timeline for Creation of Early Languages (http://www.computerhistory.org/)

Year	Development
1945	Kruse works on Plankalkul
1948	Claude Shannon identifies the bit as the fundamental unit of data and shows how to code data for transmission
1952	Grace Hopper complete the A-0 compiler
1953	John Backus creates 'speedcoding' for the IBM 701
1956	Bob Patrick of GM and Owen Mock of North American Aviation create the batch processing system GM-NAA for the IBM 704
1957	FORTRAN runs for the first time (Backus at IBM)
1960	COBOL created by a team representing manufacturers under Howard Bromberg
1960	LISP created by John McCarthy
1962	Kenneth Iverson develops APL
1963	ASCII determined
1964	BASIC created at Dartmouth by Thomas Kurtz and John Kemeny
1965	Simula written by Kristen Nygaard and Ole-John Dahl
1969	UNIX developed at AT&T by Kenneth Thompson and Dennis Ritchie

Eventually computer languages became codified and distributed. This led to the need to provide a trained workforce, and formal institutions such as universities became providers of this training. For example, the University of Sydney introduced a course called *'The Theory of Computation, Computing Practices and Theory of Programming'* in 1947 (Tatnall and Davey, 2004).

The speed of the introduction of specialized degrees paralleled the introduction of hardware and software in industry. At that time the computing industry and academia were intertwined. Industry progressed due to innovations made by university academics, and many industry leaders moved to teaching and research positions. In Australia the 1960s saw Gerry Maynard move from the Post Office to set up a course at the Caulfield Technical College, Donald Overheu move from the Weapons Research Establishment to the University of Queensland, and Westy Williams leave the public service to start a program at Bendigo Technical College (Tatnall and Davey, 2004). Computing founders in the USA were also intimately connected with Universities. Grace Hopper, originally a teacher of mathematics at Vassar, became a research fellow at Harvard while she worked on the Mark I and Mark II for the navy (Sammet, 1981). Alternatively, Backus produced FORTRAN as an IBM employee and the language became rooted in industry before being introduced in academia (Perlis, 1981). The later development of ALGOL was the result of a conglomeration of actors from industry and academia representing many stakeholders in play today, including D. Arden (MIT), J. Backus (IBM), P. Desilets (Remington-Rand Univac), D. Evans (Bendix), R. Goodman (Westinghouse), S. Gorn (University of Pennsylvania), H. Huskey (University of California), C. Katz (Remington-Rand Univac), J. McCarthy

(MIT), A. Orden (University of Chicago), A. Perlis (Carnegie Tech), R. Rich (Johns Hopkins), S. Rosen (Philco), W. Turanski (Remington-Rand Univac), and J. Wegstein (National Bureau of Standards) (Perlis, 1981).

These examples indicate that the history of computer language selection should be viewed in light of both the nature of languages and the stakeholders that determine the lifetime of each language. In fact, it can be argued that a language becomes mature when it is recognised by a University for teaching.

2 History of Language Development

The plethora of new languages in the period from 1960 to 1971 makes the task of identifying trends difficult. As early as 1960 there were 73 languages in existence (Sammet, 1972). By 1967 there were 117, and by 1971 there were 164 (Sammet, 1972). One response to the difficulty of determining language significance was taken by the ACM Special Interest Group on Programming Languages (SIGPLAN). The program committee for the History of Programming Languages (HOPL) conference in Los Angeles assessed language importance via the following criteria: (1) the language has been in use for at least 10 years, (2) the language has significant influence, and (3) the language is still in use (Bergin and Gibson, 1996).

The development of FORTRAN began in 1954 and culminated in the first release in 1957. Smillie (2004) recalls FORTRAN's amazing appearance in light of how it changed programming from almost electronics into a human activity:

> *I remember a lecture given by a colleague, Peter Sefton, in the late 1950s on a new language called FORTRAN, which he said he thought might relieve some of the tedium of programming in machine language.*

ALGOL, released in 1958 and updated in 1960, introduced recursion, indirect addressing, and character manipulation, among other features. Many universities adopted ALGOL as the language for use in their computer programming courses because it was a precise and useful way for capturing algorithms (Keet, 2004). COBOL was developed in 1959 and was widely used for a number of decades in business applications. By 1972, most universities in Australia and the USA had established computer science or information systems (the latter often called 'data processing') degree programs. Almost all computer science degree programs offered ALGOL, FORTRAN, or LISP, while most data processing programs offered COBOL. In Britain, BASIC was also important. During the late 60s, departments experimented with various languages like PL/I.

The mid-1970s brought about another important change – the introduction of the microcomputer. These machines came with BASIC and revolutionised the teaching of computer courses in high schools. Most secondary schools immediately started using BASIC, but this trend did not impact university programs.

With the introduction of Pascal in the 1970s, most universities adopted Pascal for their introductory programming course. Some authors attribute this to two pragmatic factors: the invention of the personal computer, and the availability of Pascal compilers (Levy, 1995). Pascal compilers were always far slower than the languages used in industry, but the speed was well within the limits needed in a teaching environment. At this time academics used arguments to justify the divergence from using industrially common languages. For example, Merritt (1980) wrote:

> *Since Pascal is a widely available and well-designed language, it was suggested that Pascal provided a unique language environment in which these features that support high quality program construction can be learned. However, it is reasonable to expect that reliable software will be a priority, that the connections between good programs and language features will continue to be made, and that language features will develop along the lines presented here. Information Systems graduates will be in systems development and management roles.*

The use of Pascal in academia was eventually superseded by languages used in industry, beginning with C and C++, and eventually shifting to Java and C#. As recently as 1996 a survey of CSAB accredited programs showed the most popular first language was still Pascal at 36% of the responding institutions, followed by C++ at 32% and C at 17% (McCauley and Manaris, 1998).

3 Trends in Language Selection

The debate over programming language selection has been ongoing since the introduction of programming classes in university curricula. A sampling of papers published over time provides some insights into the trends observed during given time periods.

3.1 The 1970s

Dijkstra (1972, p. 864) stated that:

> *...the tools we are trying to use and the language or notation we are using to express or record our thoughts are the major factors determining what we can think or express at all! The analysis of the influence that programming languages have on the thinking habits of their users ... give[s] us a new collection of yardsticks for comparing the relative merits of various programming languages.*

Sime (1973) noted a need for an empirical approach to evaluate programming languages for unskilled users rather than experienced users, a trend that he observed in language evaluation papers prior to his work. Yohe (1974) pointed out that the development of problem-oriented languages began in the late 1950s, and they now offered an alternative to assembly language, although that was still the most basic tool available to most programmers. The availability of so many languages, however, presented a new problem in the selection of a language best suited for a particular task. Friedman and Koffman (1976) stressed the need for structured programming as a replacement to the older versions of FORTRAN, noting that *"teaching disciplined programming at an elementary level is a nearly impossible task in the absence of a suitable* *implementation language"* (p. 1). Smith and Rickman (1976) were also seeking a replacement for FORTRAN, developing a well-designed set of criteria, including pedagogical factors, resource constraints, and political issues through which they 'graded' ALGOL W, APL, Assembler, Basic, COBOL, EULER, Structured FORTRAN, LISP, Pascal, PL/I, and SNOBOL. Furugori and Jalics (1977) reported that the results of their survey indicated that over half of the respondents still used FORTRAN in their introductory courses, while PL/I was used in a quarter of the schools. Finally, in 1978, Schneider indicated a trend toward the use of Pascal in classes. He pointed out that Pascal was the language that best met two critical and apparently opposing criteria – richness and simplicity. Pascal was rich in those constructs needed for introducing fundamental concepts in computer programming, but simple enough to be presented and grasped in a one-semester course.

3.2 The 1980s

The 1980s were marked by an increase in the number of available languages, which led to increased uncertainty about which to choose for the introductory programming course. Various paradigms were also introduced during this period. Boom and Jong (1980) performed a critical comparison of multiple programming language implementations available on the CDC Cyber 73, including ALGOL 60, FORTRAN, Pascal, and ALGOL 68. Tharp (1982) also pointed out the variety of languages available, including FORTRAN, COBOL, Jovial, Ada, ALGOL, Pascal, Pl/I, and Spitbol. He discussed several recent comparisons of programming languages on the basis of their support of good software engineering practices, availability of control structures, the programmer time required for developing a representative non-numeric algorithm, and the machine resources expended in compiling and executing it. Soloway, Bonar, and Erlich (1983) discussed recent research into finding a better match between a language and an individual's natural skills and abilities. Their study explored the relationship between the preferred cognitive strategies of individuals and programming language constructs. Luker (1989) discussed the alternatives to Pascal, noting that many instructors at that time were choosing between Ada and MODULA-2. He then examined the paradigms available, including functional programming, procedural programming, object-oriented programming, and concurrent programming.

3.3 The 1990s

King (1992) looked at the evolution of the programming course from the Computing Curricula 1978 to the Computing Curricula 1991 recommendations. He noted that the 1980s saw the creation of several important languages while at the same time several languages of the 1970s became popular. He also discussed the increasing popularity of various programming paradigms during the 1980s, including the imperative or procedural paradigm, the concurrent or distributed paradigm, the database paradigm, the functional or applicative paradigm, the logic-programming paradigm, and the object-oriented paradigm. He continued by proposing a set of criteria for the selection of programming languages. Howatt (1995) also proposed an evaluation method for programming languages. His criteria included the broad categories of language design and implementation, human factors, software engineering, and application domain. He went on to provide an evaluation approach. Howland (1997) also presented an extensive list of criteria that the author felt were important in choosing a language for introductory computer science instruction, but concluded that the selection of a programming language should be made primarily on the basis of how well key programming concepts may be expressed in the language.

3.4 The 2000s

By the turn of the century, the object-oriented paradigm was becoming more prominent, as was the importance of security. The Ad Hoc AP CS Committee (2000) noted that in their study of language selection for CSl and CS2 classes three main principles emerged: emphasis on object-orientation, need for safety in the language and environment, and a desire for simplicity. Wile (2002) stated that programming language choice is subject to many pressures, both technical and social. He organized the pressures into three competing needs: (1) those of the problem domain for which languages are used for problem solving; (2) the conceptual and computing models that underlie the designs of the languages themselves, independent of their particular problem domains; and (3) the social and physical context of use of the languages. He also observed a trend away from writing an entire application 'from scratch' in a single language to build a stand-alone system toward using general-purpose languages as the integrating medium for extensive functionality offered by database packages, web-based services, GUIs, and myriad other COTS and customized products that interface via an application program interface (API). At the same time, 'contextual concerns' for security, privacy, robustness, safety, etc., universally dominate applications across the board (p. 1027). Roberts (2004a) observed another trend, that the growth in the popularity of the object-oriented paradigm and the decision by the College Board to move the Advanced Placement Computer Science program to Java led an increasing number of universities to adopt Java as the programming language for their introductory course. He further pointed out (2004b) that there were two additional challenges in which dramatic increases had a negative impact on pedagogy: (1) the number of programming details that students must master has grown, and (2) the languages, libraries, and tools on which introductory courses depend are changing more rapidly than they have in the past. Finally, Gee, Wills, and Cooke (2005) pointed out another trend that is becoming increasingly evident (and controversial), that is, the use of scripting languages to teach programming concepts because they provide *"not only a proper programming environment but also an instant link into the formation of active web pages"*. Parker et al. (2006a, 2006b) examined a multitude of studies, including many of those mentioned above, and presented a set of criteria for use when selecting a computer programming language for an introductory programming course, and developed an instrument that allows weighting of each of those selection criteria to specify their relative importance in the selection process.

4 Language Selection Studies

The problems that must be faced in designing an introductory course are many and varied. These range from those of interdepartmental politics in the case of service courses to logistical challenges if substantial numbers of students must be accommodated (Solntseff, 1978). A cursory glance through back issues of computer-related journals such as the *ACM Special Interest Group on Computer Science Education (SIGCSE) Bulletin* makes it apparent that discussions about the introductory programming language course and the language appropriate for that course have been numerous and on-going (Smolarski, 2003). The selection of a programming language for instructional purposes is often a tedious chore because there is no well-established approach for performing the evaluation. The informal process may involve faculty discussion, with champions touting the advantages of their preferred language, and an eventual consensus, or at least surrender. As the number of faculty, students, and language options grows, this process becomes increasingly unwieldy. As it stands, the process currently lacks structure and replicability (Parker et al., 2006a).

A list of the factors that affected the choice of a programming language for an introductory course at one US university is ably discussed in Smith and Rickman (1976). According to Solntseff (1978), there *"appears to be no other discussion in the literature of comparable thoroughness"*. A current study carefully examines a first programming language for IT students (Gee et al., 2005). A more recent study examines over 60 papers relevant to language selection in academia (Parker et al., 2006a). The selection of programming languages in university curricula in the US and Australia is almost identical, with some interesting differences. The current distribution in Australia is shown in Table 2.

Table 2. Languages taught (de Raadt et al. 2003b)

Language	Number of courses	Weighted by students
Java	23	43.9%
VB	14	18.96%
C++	8	*15.2%*
Haskell	3	8.8%
C	4	5.5%
Eiffel	2	3.3%
Delphi	1	2.0%
Ada	1	1.7%
jBase	1	0.8%

This is a close approximation to the statistics in US universities. One historical difference between the countries involved Ada. When the US Department of Defense mandated Ada for their applications the language experienced a surge in US colleges, but its use declined after 1997 when the mandate was removed.

5 Selection Approaches

Over the years languages have been invented to solve problems. Other languages have been invented to make teaching algorithms easier. This has led to two sometimes conflicting lines of arguments by academics about which languages they should use in university courses: choose a language that is commonly used or is expected to be commonly used in industry, or choose a language that best supports concept development in students. Thus, there have been two distinct arguments for language selection that have been extant throughout the history of languages: pragmatic versus pedagogical.

5.1 Pragmatic Selection

The pragmatic approach recommends choosing a language that will help students get a job after graduating. The pragmatic approach is impacted by a language's industry acceptance as well as the marketability of individuals proficient in its use.

5.1.1 Industry Acceptance

Industry acceptance refers to the market penetration (Riehle, 2003) of a particular language in industry, i.e., the use of a language in business and industry. Often referred to as industrial relevance, this can be assessed based on current and projected usage, as well as the number of current and projected positions. Stephenson (2000) claims that this factor has the greatest influence in language selection, as indicated by 23.5% of schools that participated in his study. Lee and Stroud (1996) point out that real-world acceptability is a factor that once had little weight, as indicated by the earlier use of ALGOL and Pascal, but that attitude does seem to be changing. They note that for their students being able to have an industrially accepted language on their résumé is a significant consideration for them. A 2001 census of all Australian universities revealed that perceived industry demand was the major factor in the choice of an introductory language (de Raadt et al., 2003a). King (1992) agrees that many language decisions are made on the basis of current popularity or the likelihood of future popularity; he notes that choosing popular languages has a number of practical benefits, including increased student motivation to study a language that they have heard of and know is in demand, as well as a good selection of books and language implementations that will be available for a popular language.

5.1.2 Marketability

Marketability refers to the employability of graduates. This may include regional or national/international marketability, based on the placement of a program's graduates. Language selection is often driven by demand in the workplace, i.e., what employers want. Not only are marketable skills important in future employability, but students are more enthusiastic when studying a language they feel will increase their employability (de Raadt et al., 2003a). Language marketability is stressed in several studies. The census of introductory programming courses conducted by de Raadt et al. (2003a) emphasizes the importance of employability. In fact, the most commonly listed factor in language selection (by 56% of the participants) was the desire to teach a language that provides graduates with marketable skills. Watt (2000) discusses the need for transferable skills that will be useful in whatever career the student chooses to pursue. Emigh (2001) agrees that the primary concern in language evaluation must be the demand in the workplace and argues that when deciding on a new language one must take into account employers' expectations of graduates. Further, graduates' marketability can be improved by exposing them to several languages (de Raadt et al., 2003a). They cite, for example, that a progression from C to C++ to Java will qualify a graduate for more advertised positions than exposure to any single language in isolation. Extrinsically motivated students aspiring to a lucrative career will demand to be taught those tools that are currently in vogue in the industry. Universities may have to accept that pedagogical issues in the choice of platform and language must be secondary to marketing concerns (Jenkins, 2001).

5.2 Pedagogical Selection

Smolarski (2003), Mclver and Conway (1996), and Howland (1997) question whether changes in the curriculum and programming courses should be as driven by industry as they often seem to be. They argue that decisions about the language used in an introductory course should be made based on how well it underscores fundamental skills that prepare the student for subsequent courses and helps to make any student-developed software well-written and error-free, rather than on what language would be most useful for a student in finding a job (Smolarski, 2003).

5.2.1 Avoiding the Complexities of Industrial Environments

These arguments also call attention to the possibility that the purposes of teaching problem solving and introducing a professional grade language into the first course conflict because students end up focusing on difficulties associated with that language and its environment (Johnson, 1995; Jenkins, 2002; Gee et al., 2005; Allison et al, 2002; Kelleher and Pausch, 2005). *"A language that requires significant notational overhead to solve even trivial problems forces the language rather than the techniques of problem-solving to become the object of study"* (Zelle, 1999).

5.2.2 Clear Problem-Solving Principles

A teaching language should have attributes that help teach fundamentals of all programming tasks. This is the argument used by Wirth (1993), Kölling et al. (1995), and all the other inventors of languages designed for classroom use, and is exemplified by proponents of the various 'pure' teaching languages. The argument quickly becomes one that urges use of a language not common in industry. Some urge development of a new teaching language to meet the needs for teaching, one that does not have to be a real world production language and thus can avoid the compromises in conceptual cleanness for efficiency that cause many of the problems with existing languages (Kölling et al., 1995).

6 Primary Selection Criterion

The relevant importance ascribed to both the pragmatic and practical approaches is illustrated by a recent survey of academics, shown in Table 3. The primary reason for language selection reported by the survey is marketability, cited by 56.1% of the respondents, followed by pedagogical benefits, cited by 33.3% of the academics.

Table 3. Reasons for choosing language (de Raadt et al. 2003b)

Used in industry / Marketable	56.1%
Pedagogical benefits of language	33.3%
Structure of degree/dept politics	26.3%
OO language	26.3%
GUI interface	10.5%
Availability/Cost to students	8.8%
Easy to find appropriate texts	3.5%
OS/Machine limitations of dept	1.8%

7 Caveats

The task of anticipating industry needs is complex. Emigh (2001) points out that four to five years pass between when a student begins a program of study and when he or she attains a position requiring programming skills. Even if a curriculum teaches a newer programming language, there is no guarantee that employers will still be looking for that language when the student enters the work force. Further, some trends are difficult to understand. Currently in Australia there seems to be a demand for multi-skilled programmers (de Raadt et al. 2003a). The average advertisement required 1.84 languages. 48% of jobs required more than one language. C++ appeared as a requirement in around 30% of advertisements, as did Java. Visual Basic was next with 21%, followed by C with 17% (de Raadt et al. 2003b). The Gottleibsen reports (Gottliebsen 1999; Gottliebsen 2001) on job advertisements in Australia for a sample of years shows 128 languages advertised in 1999, 3822 positions for C++, 2555 for Visual Basic, 1052 for Java, and 4678 for COBOL. By 2001 there were 206 languages in demand by industry, with 4359 positions for C++, 2680 for Java, 3369 for Visual Basic, and 1087 for COBOL.

An interesting omission from most programming language selection approaches is the ability to produce output using the language. Experiments such as that conducted by Zeigler (1995) could be used to help decide the issue. The same 60 programmers developed code in both Ada and C, the same work environment was used, as were the same debugging tools, same editors, same testing tools, and the same design methodology. Most of these programmers had masters degrees in computer science, and the more experienced programmers tended to work more in C. When first hired, 75% of the programmers knew C, while only 25% knew Ada. Despite the bias in C's favor, the experiment showed that the cost of coding in Ada is about half the cost of coding in C, because code written in Ada contained 70% less bugs discovered before product delivery and 90% less bugs discovered after product delivery (Zeigler, 1995). Note that this approach is limited by the shear quantity of programming languages available, well into the thousands today. A one-to-one comparison of all possible candidates cannot possibly be preformed.

Student perceptions also play a part in this debate. There exist several languages designed for teaching (e.g. Pascal, LOGO), but any department using one of these today would be an object of ridicule (Jenkins, 2002). It is true that programming languages designed for teaching purposes are not used to any extent by industry. Therefore student perception is that these languages are of little practical worth and they further assume that, in general, they lack the advanced facilities of other languages (Gee et al., 2005). If that argument were to be carried to absurdity then the overwhelming choice would be COBOL, which now has an installed base of *"more than 200 billion lines of code, and 5 billion lines of COBOL are written every year"* (Langley, 2004).

As noted earlier, Parker et al. (2006a, 2006b) propose a set of criteria for the selection of a programming language in an academic setting. Their work is based on papers by researchers in both Australia and the United States. Each of the criteria has been used in one or more previous studies that evaluate programming languages. This extended set of selection criteria points to a more formal and mature approach to language selection. As our current period moves into history, we may be able to see the early years of the twenty-first century as a time of fundamental change in language choice.

8 Conclusion

While there have been various differences throughout the years between Australia and the United States in the teaching of programming languages, there is a pattern that seems culturally independent. Across the two countries there have been, and still exist, two primary approaches to language selection. The pragmatic approach recommends choosing a language that will enhance student employability. The pedagogical approach insists that the language used in introductory programming classes should be designed for teaching programming concepts and problem solving and should minimize complexities so that more time can be spent on developing design skills. There has been no consensus on which approach is optimal, but the ultimate lesson is that neither approach is sufficient by itself.

There are additional critical factors that must be considered when selecting a programming language. Recent studies have examined a variety of factors that must be taken into account, and while pragmatic and pedagogical concerns are still near the forefront, they must be tempered by an awareness that other factors impact the selection process. The bottom line is that academics must carefully assess the best interests of the students, weigh all variables in the language selection process such as those listed by Parker et al. (2006a, 2006b), and choose a language accordingly. As Johnson (1995) points out, *"the greatest danger to our university system is the lemming-like rush to do the same thing, to be one with the crowd, to be part of the current fashion industry of computing"*.

References

Ad Hoc AP CS Committee. Round 2: Potential Principles governing language selection for CS1-CS2 (2000), http://www.cs.grinnell.edu/~walker/sigcse-ap/99-00-principles.html

Allison, I., Ortin, P., Powell, H.: A virtual learning environment forintroductory programming. In: Proceedings of the 3rd Annual Conference of the Learningand Teaching Support Network Centre for Information and Computer Sciences, Loughborough, UK (2002)

Bergin, T.J., Gibson, R.G.: History of Programming Languages-II. ACM Press, USA (1986)

Boom, H.J., de Jong, E.: A critical comparison of several programming language implementations. Software: Practice and Experience 10, 435–473 (1980)

de Raadt, M., Watson, R., Toleman, M.: Introductory programminglanguages at Australian universities at the beginning of the twenty first century. Journal of Research and Practice in Information Technology 35(3), 163–167 (2003a)

de Raadt, M., Watson, R., Toleman, M.: Language tug-Of-war: Industry demand andacademic choice. In: Australasian Computing Education Conference (ACE 2003). Australian Computer Society, Inc., Adelaide (2003b)

Dijkstra, E.W.: The humble programmer. Communications of the ACM 15(10), 859–866 (1972)

Emigh, K.L.: The impact of new programming languages on university curriculum. In: Proceedings of ISECON 2001, Cincinnati, Ohio, vol. 18, pp. 1146–1151 (2001), http://isedj.org/isecon/2001/16c/ISECON.2001.Emigh.pdf (retrieved July 10, 2005)

Friedman, F.L., Koffman, E.B.: Some pedagogic considerations in teaching elementary programming using structured FORTRAN. In: Proceedings of the ACM SIGCSE-SIGCUE Technical Symposium on Computer Science and Education, pp. 1–10 (1976)

Furugori, T., Jalics, P.: First course in computer science, a small survey. ACM SIGCSE Bulletin 9(1), 119–122 (1977)

Gee, Q.H., Wills, G., Cooke, E.: A first programming language for ITstudents. In: Proceedings of the 6th Annual Conference of the Learning and Teaching Support Network Centre for Information and Computer Sciences, York, UK (2005)

Gottliebsen, C.: Computer market results 1999. C. Gottliebsen. Bayswater, GIMA (1999)

Gottliebsen, C.: Icon index trend report 2001. Icon index Trend Report. B.Youston. Bayswater (2001)

Howatt, J.W.: A project-based approach to programming language evaluation. ACM SIGPLAN Notices 30(7), 37–40 (1995), http://academic.luther.edu/~howaja01/v/lang.pdf

Howland, J.E.: It's all in the language: yet another look at the choice ofprogramming language for teaching computer science. Journal of Computing in Small Colleges 12(4), 58–74 (1997), http://www.cs.trinity.edu/~jhowland/ccsc97/ccsc97/

Jenkins, T.: The motivation of students of programming. In: ACM SIGCSEBulletin, Proceedings of the 6th Annual Conference on Irmovation and Technology In Computer Science Education, ITiCSE 2001, vol. 33(3) (2001)

Jenkins, T.: On the difficulty of learning to program. In: Proceedings of the 3rd Annual Conference of the Learning and Teaching Support Network Centre for Information and Computing Science, Loughborough, UK (2002)

Johnson, L.F.: C in the first course considered harmful. Communications of the ACM 38(5), 99–101 (1995)

Keet, E.E.: A personal recollection of software's early days (1960-1979): Part1. In: IEEE Annals of the History of Computing (October-December) (2004)

Kelleher, C., Pausch, R.: Lowering the barriers to programming: Ataxonomy of programming environments and languages for novice programmers. ACM Computing Surveys 37(2), 83–137 (2005)

King, K.N.: The evolution of the programming languages course. In: Proceedingsof the Twenty-Third SIGCSE Technical Symposium on Computer Science Education, Kansas City, Missouri, pp. 213–219 (1992)

Kölling, M., Koch, B.: Requirements for a first year object-orientedteaching language. In: ACM SIGCSE Bulletin, Proceedings of the Twenty-Sixth SIGCSE Technical Symposium on Computer Science Education, vol. 27(1). Language History - A Tale of Two Countries 151 (1995)

Langley, N.: COBOL integrates with Java and .NET. Computer Weekly (2004), http://www.computerweekly.com/articles/article.asp?liArticleID=133085

Lee, P.A., Stroud, R.J.: C++ as an introductory programming language. In: Woodman, M. (ed.) Programming Language Choice: Practice and Experience. International Thomson Computer Press, London (1996), http://www.cs.ncl.ac.uk/oId/publications/books/apprentice/lnstructorsManual/C-H-_Choice.html

Levy, S.P.: Computer Language Usage In CS 1: Survey Results. SIGCSE 27(3), 21–26 (1995)

Luker, P.: Academic staff development in universities with special reference to small group teaching. (Unpublished PhD Thesis), University of Nottingham (1989)

McCauley, R., Manaris, B.: Computer science programs: what do they looklike? In: Proceedings of the 29th SIGCSE Technical Symposium on Computer Science Education, pp. 15–19 (February 1998)

Mclver, L., Conway, D.M.: Seven deadly sins of introductory programminglanguage design. In: Proceedings of Software Engineering: Education and Practice (SE:E&P 1996), Dunedin, NZ, pp. 309–316 (1996)

Merritt, S.M.: On the importance of teaching Pascal in the IS curriculum. In: ACM SIGCSE Bulletin, Proceedings of the Eleventh IGCSE Technical Symposium on Computer Science Education SIGCSE 1980, vol. 12(1) (1980)

Perlis, A.J.: The American Side of the Development of Algol. In: Wexelblat, R.L. (ed.) History of Programming Languages I, pp. 25–74. ACM (1981)

Parker, K.R., Ottaway, T.A., Chao, J.T.: Criteria for the selection of a programminglanguage for introductory courses. International Journal of Knowledge and Learning 2(1/2), 119–139 (2006a)

Parker, K.R., Chao, J.T., Ottaway, T.A., Chang, J.: A formal language selection process for introductory programming courses. Journal of Information Technology Education 5, 133–151 (2006b)

Riehle, R.: SEPR and programming language selection. CrossTalk – TheJournal of Defense Software Engineering 16(2), 13–17 (2003), http://vkfww.stsc.hill.afmil/crosstalk/2003/02/Riehle.html

Roberts, E.: Resources to support the use of java in introductory computer science. In: Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education,Norfolk, Virginia, pp. 233–234 (2004)

Sammet, J.E.: Programming languages: History and future. Communicationsofthe ACM 15(7), 601 (1972)

Sammet, J.E.: The Early History of COBOL. In: Wexelblat, R.L. (ed.) History of Programming Languages I. ACM (1981)

Schneider, G.M.: The introductory programming course in computer science: Ten principles. ACM SIGCSE Bulletin 10(1), 107–114 (1978)

Sime, M.E., Green, T.R.G., Guest, D.J.: Psychological evaluation of two conditional constructions used in computer languages. International Journal of Man-Machine Studies 5(1), 105–113 (1973)

Smillie, K.: People, languages, and computers: A short memoir. IEEE Annals ofthe History of Computing (April-June), 60–73 (2004)

Smith, C., Rickman, J.: Selecting languages for pedagogical tools in the computer sciencecurriculum. In: Proceedings of the 6th SIGCSE Technical Symposium on Computer Science Education, pp. 39–47 (1976)

Smolarski, D.C.: A first course in computer science: Languages and goals. Teaching Mathematics and Computer Science 1(1), 137–152 (2003), http://math.scu.edu/~dsmolars/smolar-e.pdf (retrieved November 10, 2005)

Soloway, E., Bonar, J., Ehrlich, K.: Cognitive strategies and looping constructs: an empirical study. In: Soloway, E., Spohrer, J.C. (eds.) Studying the Novice Programmer, pp. 853–860. Lawrence Erlbaum Associates, Hillsdale (1989)

Solntseff, N.: Programming languages for introductory computing courses: aposition paper. ACM SIGCSE Bulletin 10(1), 119–124 (1978)

Stephenson, C.: A report on high school computer science education in five USstates (2000), http://www.holtsoft.com/chris/HSSurveyArt.pdf

Tatnall, A., Davey, B.: Stream in the history of computer education in Australia. In: History of Computing in Education. J. Impagliazzo and J. A. N. Lee, Kluwer Academic Publishers (2004)

Tharp, A.L.: Selecting the ‘right’ programming language. ACM SIGCSE Bulletin 14(1), 151–155 (1982)

Watt, D.A.: Programming languages-Trends in education. In: Proceedings of Simposio Brasileiro de Linguagens de Programacao, Recife, Brazil (2000), http://www.dcs.gla.ac.uk/~daw/publications/PLTE.ps

Wile, D.S.: Programming languages. In: Marciniak, J.J. (ed.) Encyclopedia of Software Engineering, 2nd edn., pp. 1010–1023. John Wiley and Sons, Hoboken (2002)

Wirth, N.: Recollections about the development of Pascal. In: ACM SIGPLAN Notices, The Second ACM SIGPLAN Conference on History of Programming Languages HOPL-II 28(3) (1993)

Yohe, J.M.: An overview of programming practices. ACM Computing Surveys 6(4), 221–245 (1974)

Zeigler, S.F.: Comparing development costs of C and Ada. Rational SoftwareCorporation, Santa Clara, Calif, March 30 (1995)

Zelle, J.M.: Python as a first language. In: Proceedings 13th Annual Midwest Computer Conference (MCC 1999), Lisle, IL, March 18-19 (1999)

History of Data Centre Development

Rihards Balodis and Inara Opmane

Institute of Mathematics and Computer Science,
University of Latvia (IMCS UL), Riga, Latvia
inara.opmane@lumii.lv

Abstract. Computers are used to solve different problems. For solving these problems computer software and hardware are used, but for operations of those computing facilities a Data Centre is necessary. Therefore, development of the data centre is subordinated to solvable tasks and computing resources. We are studying the history of data centres' development, taking into consideration an understanding of this. In the beginning of the computer era computers were installed in computing centres, because all computing centres have defined requirements according to whom their operation is intended for. Even though the concept of 'data centre' itself has been used since the 1990s, the characteristic features and requirement descriptions have been identified since the beginning of the very first computer operation. In this article the authors describe the historical development of data centres based on their personal experience obtained by working in the Institute of Mathematics and Computer Science, University of Latvia and comparing it with the theory of data centre development, e.g. standards, as well as other publicly available information about computer development on the internet.

Keywords: Computing facilities, Data Centre, historical development.

1 Basic Characteristics of Data Centre Facilities

1.1 Data Centre Definition

A data centre is a physical environment facility intended for housing computer systems and associated components. Data centres comprise the above-mentioned computer systems and staff that maintains them. The necessary physical environment facility encompasses power supplies with the possibility to ensure backup power, necessary communication equipment and redundant communication cabling systems, air conditioning, fire suppression and physical security devices for staff entrances.

Usually data centres are created for specific task classes; it is particularly referable to computing facilities. Also, the development of other components' can be influenced by the class of tasks solved in the data centre; e.g. if computation result necessity is not so critical in time, then in the centre's equipment there is no duplication of resources used for electricity supply, i.e. according to present-day standards data centre's level from Tier1 to Tier4 is chosen.

Similarly, in the concept of a data centre, often, not only the data centre's maintenance staff is included, but also the centre's personnel that ensure functional

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 180–203, 2012.

actions. Frequently, such data centres are as a component in agencies, companies, institutes or even laboratories.

Historically the development of data centres has passed in a fluent way, growing from IT (information technology) departments or a computer building research laboratories' base. It is quiet problematic to determine a precise date in history when data centre was created: it could be taken from the documents of some data centre's creation or when the first computer was disposed in it, or when it began its work.

In our historical timelines we indicate the principal time, not going deep in defining precise dates. Examples of up-to-date data centre services:

- hardware installation and maintenance,
- managed power distribution,
- backup power systems,
- data backup and archiving,
- managed load balancing,
- controlled Internet access,
- managed e-mail and messaging,
- server co-location and hosting,
- virtual servers, GRID and Cloud computing services,
- managed user authentication and authorization,
- firewalls, data security,
- etc.

1.2 Electricity Supply System

Two aspects of energy use are critical for data centres. Firstly, both IT equipment and all supporting equipment are very energy consuming. Some data centres' facilities have power densities that exceed more than 100 times than in typical office use. For higher power density facilities, electricity costs are a dominant operating expense and account for over 10% of the total cost of data centre's ownership.

Secondly, it is not less important to have guaranteed energy for IT equipment and also for other equipment like cooling, or access control systems used in the data centre. Backup power systems consist of one or more uninterruptible power supplies (UPS) and/or diesel generators. To prevent single points of failure, all elements of the electrical systems, including backup systems, are typically fully duplicated, and computing facilities are connected to both power feeds. This arrangement is often marked as N+1 redundancy. Static switches are sometimes used to ensure instantaneous switchover from one supply to the other in the event of a power failure. For comparing electro-effectiveness in different data centres and also to evaluate different solutions, power usage effectiveness (PUE) is introduced. The most popular measurement of that is the proportion between total facility's power and IT equipment's power.

In addition to IT equipment the other energy consuming equipment, mainly consists of cooling systems, power delivery and other facility's infrastructure, like lighting. Data centres support equipment recalled as overhead. Often IT equipment consumes only 33% of the power in the data centre.

The most popular power components are as follows:

- generators,
- UPS,
- grounding,
- power and environmental control.

1.3 Low-Voltage Cable Routing

Communications in data centres today are most often based on networks running Internet protocols and special protocols for computing equipment interconnection. Data centres contain a set of ports, routers and switches that transport traffic between data centres' computing equipment and the outside world. Redundancy of the Internet connection is often provided by using two or more upstream Internet service providers.

Network security elements are also usually deployed: firewalls, VPN gateways, intrusion detection systems, etc.

Data cabling is typically routed through overhead cable trays in modern data centres. But some are still recommending under raised floor cabling for security reasons and to consider the addition of cooling systems above the racks in case this enhancement is necessary. Structural cabling elements are:

- service provider cables
- backbone cables (to data centre and in-between floors)
- horizontal cables (within floors)
- zone distribution (in data centre)

1.4 Raised Floors

Data centres typically have raised flooring made up of 60 cm removable square tiles. The trend is towards 80–100 cm void to cater for better and uniform air distribution. These provide a plenum for air to circulate below the floor, as part of the air conditioning system, as well as providing space for power cabling.

Raised floors have better a appearance than overhead cabling and allow higher power densities, better control of cooling, and more flexibility in location of cooling equipment. Most stand-alone computer systems are designed for cabling from below.

Overhead cable trays are less expensive than raised floor systems, cable trays can be attached to the top of racks (if they are uniform in height). Cable trays suspended from the ceiling provide more flexibility for supporting racks of various heights and for adding and removing racks.

Cable trays can be installed with several layers, for example, a cable tray system for low-voltage signals, a middle layer for power and a top layer for fibre.

1.5 Environmental Control

The requisite physical environment for a data centre is rigorously controlled. Air conditioning is used to control the temperature and humidity in the data centre. The temperature in a data centre will naturally rise because electrical power used heats the air. Unless the heat is removed, the ambient temperature will rise, resulting in electronic equipment malfunction. By controlling the air temperature, the server components at the board level are kept within the manufacturer's specified temperature/humidity range. Air conditioning systems help control humidity by cooling the return space air below the dew point. In data centres there are several technologies being used to realize environmental control. The principles of data centre cooling-air delivery, movement, and heat rejection are not complex.

The under-floor area is often used to distribute cool air to the server racks. Eventually, the hot air produced by the servers recirculates back to the intakes of the CRAC units (the term 'Computer Room Air Conditioning' was introduced in 1960) that cool it and then exhaust the cool air into the raised floor plenum again.

This cold air escapes from the plenum through perforated tiles that are placed in front of server racks and then flows through the servers, which expel warm air in the back. Racks are arranged in long aisles that alternate between cold aisles and hot aisles to avoid mixing hot and cold air.

- **Computer Room Air Conditioners (CRAC).** Refrigerant-based (DX), installed within the data centre's floor and connected to outside condensing units. Moves air throughout the data centre via a fan system and delivers cool air to the servers, returns exhaust air from the room.
- **Computer Room Air Handler (CRAH).** Chilled water based, installed on the data centre's floor and connected to outside cooling plant. Moves air throughout the data centre via a fan system: delivers cool air to the servers, returns exhaust air from the room.
- **Humidifier.** Usually installed within CRAC / CRAH and replaces water loss before the air exits the A/C units. Also available in standalone units.
- **Chiller.** The data centre's chiller produces chilled water via refrigeration process. Delivers chilled water via pumps to CRAH.
- **In-Rack Cooling.** In-rack cooling products are a variant on the idea of filling the entire room with cool air and can also increase power density and cooling efficiency beyond the conventional raised-floor limit. Typically, an in-rack cooler adds an air-to-water heat exchanger at the back of a rack so that the hot air exiting the servers immediately flows over coils cooled by water, essentially short-circuiting the path between server exhaust and CRAC input. In some solutions, this additional cooling removes just part of the heat, thus lowering the load on the room's CRACs (i.e. lowering power density as seen by the CRACs), and in other solutions it completely removes all heat, effectively replacing the CRACs. The main downside of these approaches is that they all require chilled water to be brought to each rack, greatly increasing the cost of plumbing and the concerns over having water on the data centre's floor with couplings that might leak.
- **Different trends in Environmental control.** Energy-saving technology – 'green' data centres and rational cooling technologies.

Modern data centres try to use economizer cooling, where they use outside air to keep the data centre cool. Many data centres now cool all of the servers using outside air. They do not use chillers/air conditioners, which create potential energy savings.

A water side economizer uses the outside air in conjunction with a chiller system. Instead of compressors, the outside air cools the water, which is then pumped to data centre CRAHs. Water side economizers are marketed as either evaporative coolers or dry coolers.

The industry is exploring progressive cooling solutions because the current generation, discussed earlier, has proven insufficient and inflexible with increased computing requirements. (Chillers, for instance, are estimated to consume 33% of a facility's total power in current layouts).

To optimize the cooling in your data centre a good first step is an in-depth analysis of your current environment to gain a holistic understanding of your data centre's environment, increased awareness of your critical risk factors, benchmark of performance metrics, and generate a punch list of opportunities for cooling improvement.

1.6 Fire Protection

Data centres feature fire protection systems, including passive and active design elements, as well as implementation of fire prevention programs in operations. Smoke detectors are usually installed to provide early warning of a developing fire by detecting particles generated by smouldering components prior to the development of flame. This allows investigation, interruption of power, and manual fire suppression using hand held fire extinguishers before the fire grows to a large size. A fire sprinkler system is often provided to control a full scale fire if it develops. Fire sprinklers require 46 cm of clearance (free of cable trays, etc.) below the sprinklers. Clean agent fire suppression gaseous systems are sometimes installed to suppress a fire earlier than the fire sprinkler system. Passive fire protection elements include the installation of fire walls around the data centre so a fire can be restricted to a portion of the facility for a limited time in the event of the failure of the active fire protection systems, such as making sure the door is not left open or if they are not installed. For critical facilities these firewalls are often insufficient to protect heat-sensitive electronic equipment, however, because conventional firewall construction is only rated for flame penetration time, not heat penetration. There are also deficiencies in the protection of vulnerable entry points into the server room, such as cable penetrations, coolant line penetrations and air ducts.

1.7 Security

Physical security also plays a large role in data centres. Physical access to the site is usually restricted to selected personnel, with controls including bollards and mantraps. Video camera surveillance and permanent security guards are almost always present if the data centre is large or contains sensitive information on any of the systems within. The use of finger print recognition mantraps is starting to be commonplace.

1.8 Data Centre Spaces

A data centre lies in a number of rooms: a computer room with computing facilities, a computer room for telecommunication equipment, an operation centre, entrance facilities, rooms outside computer rooms for mechanical and electrical spaces - for power generators, UPS, cooling refrigerants, structured cabling and communication entry cabling rooms etc. In data centres there are also stationed support staff offices and client service front office. The most important spaces in data centres are: computer room, computer room TR, operations centre, entrance facility.

Some of the spaces are illustrated by photos from IMCS UL.

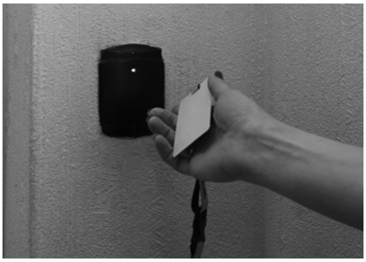
Computer room

Fig. 1. Back side of racks

Fig. 2. Front side of racks

Support Equipment (Backup Power)

Entrance Facility



2 Framework of Data Centres Development

Computing facilities is a kernel of data centre and other conditions were adjusted so that it secured effective operation of computing. In the table below, we will summarize a short overview about the computing development history, mainly pointing out the information available on the Internet for further independent reader investigation. Often, to display historical development timelines are used, and they have different forms:

- timelines as lists with parameters (time and description),
- timelines as tables,
- timelines with photos and videos,
- interactive timelines with specially made software,
- posters,
- wiki descriptions,
- museum descriptions.

Table 1. Historical development timelines

Timeline content	Link	Type of timeline
Computer science timeline	http://www.atariarchives.org/deli/Time_Line.php	Timeline and photo
	http://www.rci.rutgers.edu/~cfs/472_html/Intro/timeline.pdf To commemorate the 50th year of modern computing and the Computer Society	Timeline and photo
	http://www.ieeeghn.org/wiki/index.php/Category:Computers_and_information_processing Category: Computers and information processing IEEE Global History Network	Description and Photo, Interactive
	http://cms.uhd.edu/Faculty/BecerraL/Mycourses/History_timelines.htm Some History Timelines In Computer Science, Mathematics And Statistics, List of links	Link

Table 1. (*continued*)

Industrial Technology timelines	http://www.thocp.net/timeline/timeline.htm Chronology of the History of Computing	Timeline and photo
	http://www.timetoast.com/timelines/computer-science-through-the-decades	Interactive
	http://en.wikipedia.org/wiki/History_of_technology History of technology	WIKI
	http://www.feb-patrimoine.com/histoire/english/information_technology/information_technology_3.htm Information Technology Industry TimeLine	Table
	http://www.saburchill.com/HOS/technology/008.html History of science and technology industrial revolution timeline	Table
	http://cs-exhibitions.uni-klu.ac.at/index.php?id=187 The history of (computer) storage	Timeline
	http://www.slideshare.net/wizbee/timeline-of-computer-history-8498853 Slideshare	Timeline and photo
Timeline of virtualization development	http://en.wikipedia.org/wiki/Timeline_of_virtualization_development Timeline of virtualization development	WIKI
	http://www.hds.com/go/hds-virtualization-timeline/ Storage Virtualization - Hitachi Data	Interactive
History of computing	http://en.wikipedia.org/wiki/Timeline_of_computing Timeline of computing	WIKI
	http://www.history-timelines.org.uk/events-timelines/07-computer-history-timeline.htm Computer History Timeline	Timeline and photo
	http://www.computerhistory.org/timeline/?category=cmpny Timeline of Computer History	Museum
	http://www.warbaby.com/FG_test/Timeline.html Full Timeline	Table
	http://www.tnmoc.org/timeline.aspx The National Museum of Computing	Museum
	http://www.askthecomputertech.com/computer-history-timeline.html Computer History Timeline	Timeline
	http://archives.icom.museum/vlmp/computing.html The Virtual Museum of Computing	Museum
	http://en.wikipedia.org/wiki/History_of_computer_hardware_in_Soviet_Bloc_countries History of computer hardware in Soviet Bloc countries	WIKI
The History of the Mainframe	http://web-friend.com/help/general/pc_history.html The History of Computers: From PC to mainframe	Timeline

Table 1. (*continued*)

History of software	http://en.wikipedia.org/wiki/History_of_software_engineering History of software engineering	WIKI
	http://en.wikipedia.org/wiki/Category:History_of_software Category: History of software	WIKI
	http://www.inf.ethz.ch/personal/wirth/Articles/Miscellaneous/IEEE-Annals.pdf A Brief History of Software Engineering (Niklaus Wirth)	Description
	http://www.softwarehistory.org/ The Software Industry Special Interest Group at the Computer History Museum	Description
Operating systems timeline	http://en.wikipedia.org/wiki/Timeline_of_operating_systems Timeline of operating systems	WIKI
	http://www.networkdictionary.com/Software/Computer-Operating-Systems-Development-Timeline.php Computer Operating Systems Development Timeline	Timeline
	http://www.templejc.edu/dept/cis/ccollins/ITSC1305/os-timeline.html Operating System Timeline	Timeline and photo
	http://www.scribd.com/doc/36626225/Timeline-of-Operating-Systems Timeline of operating systems	Timeline
	http://wiki.jumba.com.au/wiki/List_of_Operating_Systems List of Operating Systems	WIKI
	http://trillian.randomstuff.org.uk/~stephen/history/timeline-OS.html Brief History of Computing - Operating Systems	Timeline
Programming language timeline	http://en.wikipedia.org/wiki/Timeline_of_programming_languages http://en.wikipedia.org/wiki/History_of_programming_languages Timeline of programming languages	WIKI
	http://oreilly.com/news/graphics/prog_lang_poster.pdf History of Programming Languages	Poster
	http://encyclopedia.thefreedictionary.com/Programming%20language%20timeline Timeline of programming languages	WIKI
	http://oreilly.com/news/languageposter_0504.html The History of Programming Languages	Poster
A Timeline of Database History	http://quickbase.intuit.com/articles/timeline-of-database-history A Timeline of Database History	Timeline
	http://en.wikipedia.org/wiki/User:Intgr/RDBMS_timeline RDBMS timeline	WIKI, poster

Table 1. (*continued*)

Timeline of Systematic Data and the Development of Computable Knowledge	http://www.wolframalpha.com/docs/timeline/	Table and photo
	http://store.wolfram.com/view/misc/popup/timeline.html Timeline of Computable Knowledge Poster	Poster
	http://timerime.com/en/timeline/71986/History+of+Data+Storage/ History of Data Storage	Timeline and photo
History of the Internet	http://www.webopedia.com/quick_ref/timeline.asp Brief Timeline of the Internet	Interactive
	http://www.zakon.org/robert/internet/timeline/ Hobbes' Internet Timeline 10.2	Timeline
	http://www.history-timelines.org.uk/events-timelines/11-internet-history-timeline.htm Internet History Timeline	Timeline
History of multimedia	http://www.timetoast.com/timelines/the-history-and-development-of-multimedia-a-story-of-invention-ingenuity-and-vision	Interactive
	http://www.cs.cf.ac.uk/Dave/Multimedia/node8.html History of Multimedia Systems	Timeline
History of Telecommunications	http://www.ieeeghn.org/wiki/index.php/Category:Communications IEEE Global History Network	Interactive
	http://en.wikipedia.org/wiki/History_of_telecommunication History of telecommunication	WIKI
	http://www.telcomhistory.org/timeline.shtml The Telecommunications History Group	Timeline and photo

3 Standardization for Data Centres

For ensuring computer operation we are using data centres – appropriately equipped environment. Hardware that is deposited in data centres needs wires of power supply and communications, appropriate HVAC (heating, ventilation, and air conditioning) climate conditions as well as fire safety, physical and logical security must be ensured. For all this, technological solutions were developed. Currently there are standards made up for data centres' equipping.

To ensure housing of computer systems in the data centre we are using a number of: continual research and development, their industrial production and standards among telecommunication, power usage, air conditioning, fire protection, prevention and suppression, physical premises' security systems etc. Necessary industrial solutions for data centres are used in many other economic areas. For example, ASHRAE was formed as the American Society of Heating, Refrigerating and Air-Conditioning Engineers by the merger in 1959 of The American Society of Heating and Air-Conditioning Engineers (ASHAE) founded in 1894 and The American Society of Refrigerating Engineers (ASRE) founded in 1904. The recommendations and standards worked out by those associations are taken into consideration when designing data centres, focusing on building systems, energy efficiency, indoor air quality and sustainability within the industry.

The development of data centres' provision technologies is facilitated by management of associations, research institutions and specific companies, for example:

- AFCOM (Association for Computer Operations Management), was originally established in 1980;
- The Open Data Center Alliance was formed in 2010;
- European Data Centre Association, (EUDCA), 2011;
- Data Center University (DCU) offers industry-leading education;
- etc.

The most important in data centres' standardization is the Tier level classification system. The Data Centre Tier Performance Standards are a user set of requirements used to clearly define expectations for the design and management of the data centre to meet a prescribed level of availability. The Tier Level Classification system is the foundation used by many data centres' users, consultants and design professionals in establishing a 'design-versus performance' ranking approach to today's data centre projects.

Established in 1993, the Uptime Institute is an unbiased, third-party data centre research, education, and consulting organization focused on improving data centre performance and efficiency through collaboration and innovation. The Uptime Institute serves all shareholders of the data centre industry, including enterprise and third party operation, manufacturers, providers, and engineers. This collaborative approach, completed with the Uptime Institute's capability to recognize trends on a global basis and to interface directly with owners, results in solutions and invocations freed from regional constraints for the benefit of the worldwide data centre industry.

Founded in 1993, the Institute pioneered the creation and facilitation of end-user knowledge communities to improve reliability and uninterruptible availability—uptime—in data centre facilities and Information Technology organizations.

Uptime Institute's activity began with UUUG. The establishment in 1989 of the Uninterruptible Uptime Users Group (UUUG) still left a need in the data centre community for the user's ability to disseminate information freely due to the open forum of the group meetings. Through a member-driven, collaborative learning experience, members have steadily achieved higher levels of site uptime.

Significant aspects for evaluation related to data centre performance are site selection and performance evaluation.

Table 2. Site selection and performance evaluation

Site selection	Factors	Performance evaluation	Factors
Location	Earthquake Zone Flood Plains Hurricanes or Tornadoes Proximity to Major Highways Proximity to Railway Lines Proximity to Hazardous Areas Proximity to Airports or Flight Corridors	Electrical	Utility Service Lightning Protection Power Backbone UPS Systems UPS Batteries Engine Generator Load Bank Critical Power Distribution Grounding

Table 2. (*continued*)

Infrastructure	Availability of Electrical Capacity Availability of Diverse Power Feeders Utilities Expansion/Upgrades History of Outages	Mechanical	Raised Floor Cooling UPS Cooling Mechanical Plant
Water	Diverse Source Supplies Water Storage	Support Systems	Contamination Fire Detection and Protection Physical Security Alarms and Monitoring
Communications	Availability of Diverse Carriers Availability of Diverse Services Physical Security Alarms and Monitoring		
Economics	Land Construction Utilities Labour Communications		
Staffing	Accessibility Public Transportation Recreational Facilities Housing Amenities		
Security			

The Tier I data centre has non-redundant capacity components and single non-redundant distribution paths (for power and cooling distribution, without redundant components) serving the site's computer equipment. The data centre has computer room cooling and power distribution but it may or may not have a UPS or and engine generator. The data centre must be shut down for annual predictive maintenance and repair work. Corrective maintenance may require additional shutdowns. Operation errors or spontaneous failures of infrastructure components will cause a data centre disruption. As an example, a Tier I data centre may be suitable for small businesses where IT is intended for internal business processes.

The Tier II data centre has redundant capacity components, but only single non-redundant distribution paths (for power and cooling distribution, with redundant components) serving the site's computer equipment. They have UPS and engine generators but their capacity design is Need plus One (N+1), with a single power path. Maintenance of the critical power path and other parts of the site infrastructure will require a shutdown of computer processes. The benefit of this level is that any redundant capacity component can be removed from service on a planned basis without causing the data processing to be shut down. As an example, a Tier II data centre may be appropriate for internet-based companies without serious financial penalties for quality of service commitments.

The Tier III data centre is concurrently maintainable and has redundant capacity components (dual-powered equipment) and multiple independent distribution paths serving the site's computer equipment composed of multiple active power and cooling distribution paths, but only one path active, has redundant components, and is concurrently maintainable. Generally, only one distribution path serves the computer equipment at any time. This topology allows for any planned site infrastructure activity without disruption of the computer systems operation in any way. An example of a Tier III application would include companies that span multiple time zones or whose information technology resources support automated business process.

The Tier IV data centre is fault tolerant and has redundant capacity systems and multiple distribution paths simultaneously serving the site's computer equipment, including uplinks, storage, chillers, HVAC systems, servers etc. Everything is dual-powered. All IT equipment is dual powered and installed properly to be compatible with the topology of the site's architecture. Fault-tolerant functionality also provides the ability of the site infrastructure to sustain at least one worst-case unplanned failure or event with impact to the critical load. Examples of a Tier IV requirement include companies who have extremely high-availability requirements for on-going business such as E-commerce, market transactions, or financial settlement processes.

As a rule, the overall Tier Level is based on the lowest Tier ranking or weakest component. For example, a data centre may be rated Tier 3 for electrical, but Tier 2 for mechanical performance evaluation and the data centre's overall Tier rating is then 2. In practice, a data centre may have different tier ratings for different portions of the infrastructure.

Table 3. Tier Classifications according to the Uptime Institute

	Tier 1	**Tier II**	**Tier III**	**Tier IV**
Site availability	99.67%	99.75%	99.98%	99.99%
Downtime (hours/yr)	28.8	22.0	1.6	0.4
Operations centre	Not required	Not required	Required	Required
Redundancy for power, cooling	N	N+1	N+1	2(N+1)
Gaseous fire suppression system	Not required	Not required	FM200 or Inergen	FM200 or Inergen
Redundant backbone pathways	Not required	Not required	Required	Required
UPS power outage			72 hours	96 hours

Uptime Institute has awarded 129 certificates in 25 countries around the world.

Another example of the Tier level system is presented by Syska Hennessy Group. The higher the Tier level is, higher the total costs of data centre are. A data centre's Tier level is defined by the business needs of a concrete data centre. Below, in the table, there are examples of businesses and Tier levels applied to them.

Table 4. Businesses and Tier Levels

Applied Tier level	Business
Tier 1	Professional services, Construction & engineering, Branch office (financial)
Tier 1 or Tier 2	Point of sale, Customer Resource Management (CRM), 7x24 support centres, University data centre
Tier 1 or Tier 2 or Tier 3	Enterprise Resource Planning (ERP), Online hospitality & travel reservations
Tier2	Local real time media
Tier 2 or Tier 3	Online data vaulting and recovery, Insurance, Work-in-progress tracking (manufacturing), Global real time media, Voice over IP (VoIP), Online banking, Hospital data centre, Medical records, Global supply chain
Tier2 or Tier3 or Tier4	E-commerce
Tier3	Emergency call centre
Tier 3 or Tier4	Energy utilities, Electronic funds transfer, Global package tracking
Tier 4	Securities trading and settlement

Based on the four-Tier system of The Uptime Institute in April 2005, ANSI/TIA approved ANSI/TIA-942 Telecommunication Infrastructure Standard for Data Centers (see the list). Important purposes of TIA-942 are as follows:

- specifications for data centre pathways and spaces,
- planning of data centres, computer rooms, server rooms, and other spaces,
- defining a standard of telecommunications infrastructure for data centres,
- structured cabling system for data centres,
- recommendations on media and distance for applications over structured cabling,
- establish a standard for data centres' Tiers,
- requires infrastructure administration.

Informative annex with TIA-606-A standards compliant labelling scheme for all components. All cabinets, racks, patch panels, cables, and patch cords should be labelled. Labelling scheme extended for use in data centres. Cabinets and racks labelled by location using tile grid or row/position identifiers.

Data centre infrastructure management is the integration of information technology and facility management disciplines to centralize monitoring, management and intelligent capacity planning of a data centre's critical systems. Achieved through the implementation of specialized software, hardware and sensors, a data centre's infrastructure management enables a common, real-time monitoring and management platform for all interdependent systems across the data centre facilitys infrastructures. ISO 20000 is the world's first standard for IT service management. The standard specifies a set of inter-related management processes, and is based heavily upon the ITIL (IT Infrastructure Library) framework.

However, the standards do not give answers to all questions, e.g. the Tier level of dual site data centres and redundancy in this situation.

Table 5. Overview of Telecommunications Industry Association (TIA) standards related to data centre housing (In Europe there are others, with prefix DIN)

No.	Content
TIA 568-C.0	Generic cabling
TIA 568-C.1	Commercial Building cabling
TIA 568-C.2	Balanced Twisted Pair cabling and components
TIA 568-C.3	Optical fibre cabling components
TIA 569-B	Pathways and spaces
TIA 570-B	Residential cabling
TIA 606-A	Cabling administration
TIA 607-A	Grounding (Earthing) and bonding requirements Generic Telecommunications Bonding and Grounding (Earthing) for Customer Premises, REVISION B
TIA 758-A	Customer owned outside plant cabling (approved for publishing)
TIA 862	Building Automation Systems cabling (approved for publishing)
TIA 942	Data centres infrastructure
TIA 1005	Industrial cabling infrastructure
TIA 1152	Requirements for field test instruments
TIA 1179	Development of a new Healthcare standard (TR 42.1)
ISO/IEC NP 24764	Information technology --Generic cabling for data-centres
IEEE1100-2005 -	IEEE Recommended Practice for Powering and Grounding Electronic Equipment

The history of data centres standards development is summarized in the image below:

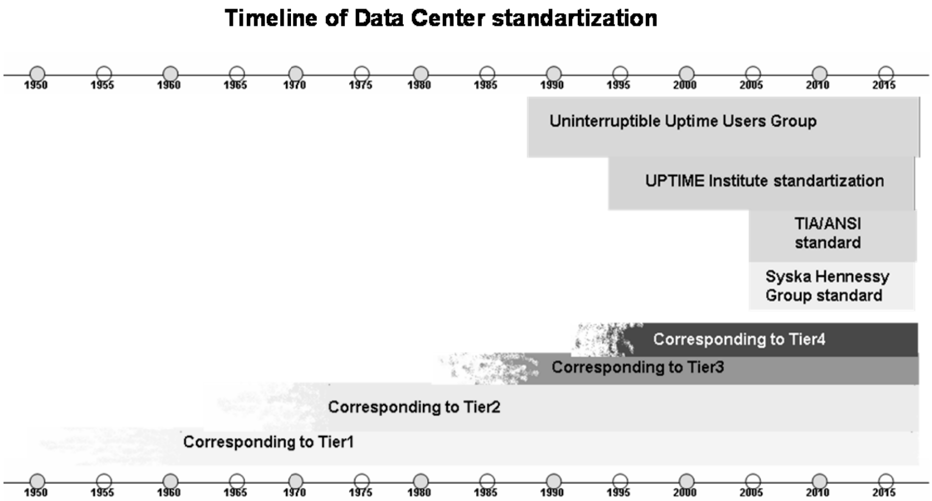


Fig. 3. Timeline of Data Centre Standardization

4 Data Centres Timelines and Dots on It

4.1 The Beginning of Data Centres

The concept of a data centre is related to industrial IT utilization. The beginning of data centre development is firmly involved with mainframe computer maintenance. The first unique mainframe computers were produced in research laboratories and also disposed there. The data centres' initial period we associate with such computer disposing and maintenance that were manufactured outside of research laboratories at least in a couple of instances. The first industrially manufactured computers and the first data centres had accumulated previous experience in mechanical data processing with different tabulators, adding machine and other devices.

Early computer systems were room-sized machines and required a lot of space. The complexity of operating and maintaining these machines also led to the practice of secluding them in dedicated rooms. Historically, assessment of data centre physical infrastructure's business value was based on two core criteria: availability and upfront cost. A data centre is a building where the primary function is to house the computer room and its support area. Four functionalities must be ensured:

- hardware disposal,
- power to maintain equipment,
- HVAC temperature controlled environment within the parameters needed,
- structured cabling in and out.

We recount examples of the first data centres formed in the following fields:

- banking sector - digital banking;

During the 1950s, researchers at the Stanford Research Institute invented the Electronic Recording Method of Accounting computer processing system (ERMA). ERMA was first demonstrated to the public in 1955, and first tested on real banking accounts in 1956. Production models (ERMA Mark II) were delivered to the Bank of America in 1959 for full-time use as the bank's accounting computer and check handling system.

In 1961 Barclays opened Britain's first computer centre for banking. The company, now called Automatic Data Processing, Inc. went public and leased its first computer in 1961, an IBM 1401.

- **Statistics.** UNIVAC I, was signed over to the United States Census Bureau on 1951.
- **Climate data processing and weather forecasting.** The Weather Bureau (USA) start in 1954- 1955 with usage of an IBM 701.
- **Medicine.** Usage of computerized citoanalyzer was started in 1954. In 1960 an IBM 650 was used to scan medical records for subtle abnormalities.
- **Civil aviation.** The first passenger reservations system offered by Sabre, installed in 1960.
- **Military field, including space.** The building of the USSR Ministry of Defence Computer Centre No1, created by A.I.Kitov in 1954. That was the first Soviet computer centre.

- Universities and research laboratories.
  - CERN's first computer was installed in 1958.
  - M.I.T. Computation Center establishing from 1956 until 1966.
  - Russian Academia of Science Computing Centre established in 1955.
  - Research Computing Center of Moscow State University was founded in 1955.
  - St.Petersburg University computer STRELA was installed in 1957.
  - In 1959 was established the Computing Center – now Institute of Mathematics and Computer Science University of Latvia (IMCS UL).

By data centre we mean the physical environment – premises and all the necessary equipment for effective operation. Data centre staff are qualified employees that ensure computer maintenance. Data centres historically have developed according to computer operation technical requirements that depend on computing facilities. As mentioned before, the data centres' initial period is connected with 1st, 2nd and 3rd generation mainframe computers. During the computers' development they became smaller and smaller, but could ensure the same computing capacities – minicomputers derived. Wherewith, by emergence of personal computers data centres' impressiveness and uniqueness entirely shrank. For personal computer usage data centres were not necessary – they could be used right there in the office. However, this IT development did not eradicate data centres entirely – other technological needs and data centres' applications derived. With the boom of personal computers started the growth of the internet and sharply in more extensive sense developed data transmission technologies, telecommunications transferring to digital technologies and a sharp merging of information and telecommunication technologies happened. Disposal of telecommunications hardware in data centres started, so they became internet nodes. Since the early 1960s, the design of data centre infrastructure has advanced through at least four clear stages:

- Tier I appeared in the early 1960s,
- Tier II in the 1970s,
- Tier III in the late 1980s and early 1990s,
- Tier IV in the mid-1990s.

These stages provide the foundation for the Four Tier Classification system defined by owners and users in association with the Uptime Institute.

4.2 Supercomputing

With great computing capacities (supercomputers) it is rationale to dispose these in data centres. Supercomputing necessity was sustained by a growing need of digital data storage amounts and computing capacities. Technological solutions and standardization developed and allowed to dispose computers and telecommunication hardware in data centres in a unified way, rack standards for hardware disposal were developed.

Data centres were always important for disposal of supercomputers and high performance computing (HPC). Supercomputers and HPC are used for highly calculation-intensive tasks such as problems including quantum physics, weather forecasting, climate research, oil and gas exploration, molecular modelling and physical simulations. Below we show growth of computing capacities in correspondence with the maximal capacity assured.

1949: 1,000 (1 KiloFlop)
1961: 100,000
1964: 1,000,000 (1 MegaFlop)
1987: 1,000,000,000 (1 GigaFlop)
1997: 1,000,000,000,000 (1 TeraFlop)

4.3 Virtualization Timeline

Communications channels historically were not powerful enough and there was no vast penetration of them, therefore personal computers-servers had to dispose in data centres as internet nodes. In that way the new service packages - colocation and hosting services were offered by data centres. In the further development of data centres services, virtualization technologies development has an important role, which currently concludes in cloud computing functionality. A short timeline of virtualization is as follows:

- 1960s: IBM introduces virtualization as a way for mainframes to share expensive memory and split mainframes into multiple virtual machines, work out timesharing and hypervisor ideas;
- 1971: IBM begins commercial production of system S/370 and adds virtual memory as a standard feature;
- 1994: Colocation services started;
- 1999: Rackspace Hosting opens their first data centre to businesses;
- 1999: VMware launches its Virtual Platform product;
- 2009: start a transition to cloud computing structures.

4.4 Racks and/or Blades

There are several advantages for each solution, but every time we design anew, there are investigations we should think about and weight ...

- **RACKS:** *A 19-inch rack is a standardized frame or enclosure for mounting multiple equipment modules. Equipment* module has a front panel that is 19 inches (482.6 mm) wide, including edges or ears that protrude on each side which allow the module to be fastened to the rack frame with screws. Developers first placed complete microcomputers on cards and packaged them in standard 19-inch racks in the 1970s soon after the introduction of 8-bit microprocessors. Equipment designed to be placed in a rack is typically described as rack-mount, rack-mount instrument, a rack mounted system, a rack mount chassis, subrack, rack mountable, rack unit or U (less commonly RU). The industry standard rack cabinet is 42U tall.

- **BLADES:** A blade server is a stripped-down server computer with a modular design. A blade enclosure, which can hold multiple blade servers, provides services such as power, cooling, networking, various interconnects and management. Cooling during operation, electrical and mechanical components produce heat, which a system must dissipate to ensure the proper functioning of its components. Most blade enclosures remove heat by using fans. Networking blade servers generally include integrated or optional network interface controllers. A blade enclosure can provide individual external ports to which each network interface on a blade will connect. Alternatively, a blade enclosure can aggregate network interfaces into interconnect devices (such as switches) built into the blade enclosure or in networking blades.

4.5 A Portable Modular Data Centre

Container-based data centres go one step beyond in-rack cooling by placing the server racks into a standard shipping container and integrating heat exchange and power distribution into the container as well. Similar to full in-rack cooling, the container needs a supply of chilled water and uses coils to remove all heat from the air that flows over it. Air handling is similar to in-rack cooling and typically allows higher power densities than regular raised-floor data centres. Thus, container-based data centres provide all the functions of a typical data centre room (racks, CRACs, PDU, cabling, lighting) in a small package. Like a regular data centre room, they must be complemented by outside infrastructure such as chillers, generators, and UPS units to be fully functional. Several large hardware vendors have developed mobile solutions that can be installed and made operational in very short time.

Computing developments which have an influence for Data Centres

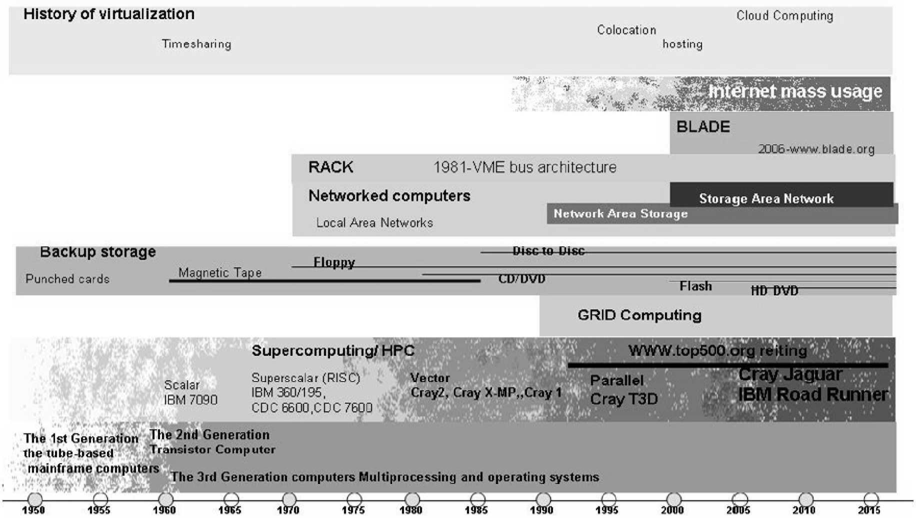


Fig. 4. Development of Data Centres in relation to IT technology transformations

5 Data Centre of the Institute of Mathematics and Computer Science, University of Latvia

Latvia is one of the Baltic region's countries in the Northern Europe. During World War I the territory of Latvia as we know today, similarly to other areas of the Russian Empire's west, was destroyed. However, after various historical events since that time, Latvia regained its independence in 1991 for good.

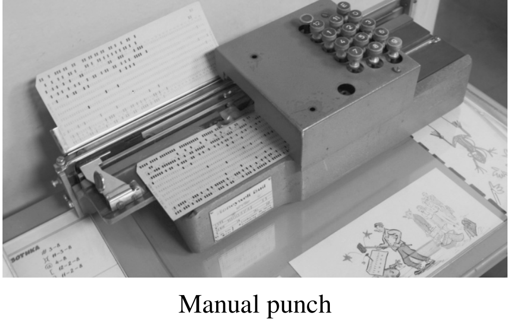
The Institute of Mathematics and Computer Science, University of Latvia (IMCS UL) was established in 1959 as a computing research centre. It was the fourth computing research centre in the Soviet Union established with a goal to develop Latvian industry and public computing services for the country. From very beginning the best computing machines available in the USSR were installed and used there. Over the years the use of computing technology, relevant science field and technology itself has changed significantly. The number of people employed at the institute has varied through different times, ranging from 120-450; at present 213 employees work in IMCS UL.

Currently IMCS is the largest and the most relevant research institution in Latvia in the field of information technology, mathematics, computer science and computational linguistics. The main research fields in IMCS UL are:

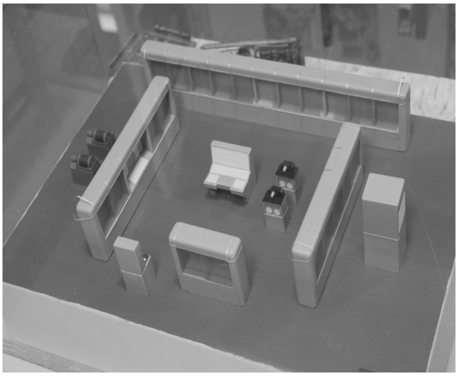
- **Computer Science**
  - Mathematical foundations of computer science
  - Complex systems modelling languages and development tools
  - Graph theory and visual information processing
  - Semantic web technologies
  - Real time systems, embedded systems
  - Computational linguistics
  - Bioinformatics
- **Mathematics**
  - Mathematical modelling for technologies and natural sciences
  - Theoretical problems of mathematical methods.

In 1984 IMCS UL established a Computing Museum. Documents and equipment in the museum reflect on computing machines of a passing age, their description and tasks which were solved with them (Balodis, Borzovs, Opmane, Skuja, Ziemele 2010)(Балодис, Опмане 2011). The Computing Museum holds 13,116 exhibits, including 504 equipment units, 287 mainframe and workstation parts, 98 computers, 44 printers, various documents and photos; and all of it is shown to approximately 500 visitors per annum.

Below we publish some photographs showing the exhibits from the museum.

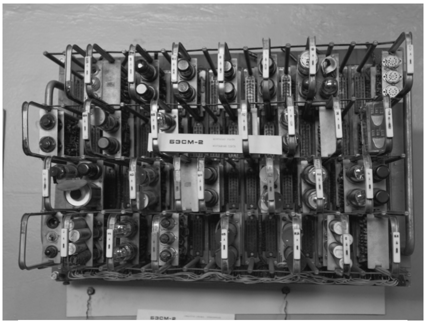


Manual punch



BESM 2 Model

Tape device



1/8 of ‘mother board’

Magnetic DRUM – ‘hard disk’

IMCS UL data centre’s main part – computing, has experienced technological transformations:

- from 1959 till 1970 – initial development of IMCS UL, operation on the basis of «BESM» and «MINSK» computers;
- from 1970 till 1990 – transition from «BESM» and «MINSK» computers to production of «ES EVM» and adaptation (cloning) of IBM’s mainframes;

from 1990 till nowadays –application of personal computers and internet, development of internet connectivity international node, development of GRID computing, establishing of unified computing facility as regional partner facility in European Union.

The first industrial soviet computers had many technological deficiencies:

- for effective work innovative laboratory of engineers and electricians had to be established;
- installation of first computers substantially contributed to the research growth, particularly research in the development of methods of mathematical modelling of various physical processes, research in the development of software and research in theoretical computer science;

- along with the research practical information systems for Latvian economics were developed;
- as from the seventies there was a trend for cloning, however, IMCS UL retained its initiative to carry out research for original software development.

BESM 2 Console (~1/4 part)

BK-0010 for schools – components: home colour TV, pseudo-membrane keyboard combined with processor and memory

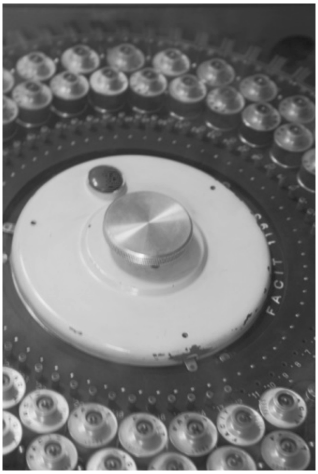


Table 6. Computers located in IMCS UL's data centre from 1959 to 1992

Exploitation time	Computers
11.04.1961–21.08.1970	BESM-2 (first generation, vacuum tubes, 5000 op/s)
29.06.1964–3.04.1972	BESM-2M
11.04.1967–06.04.1978	BESM-4 (second generation, transistors, 20 000 op/s) Computer modernization with FACIT ECM 64
03.09.1969.- 03.04.1983	GE-415 (40 000 - 90 000 op/s)
16.05.1974–31.12.1978	ES EVM -1020 (11 800 op/s)
1976–31.12.1987	ES EVM -1022 (80 000 op/s)
02.1980–31.12.1987	ES EVM -1022-02 (80 000 op/s)
12.1982–1990	ES EVM -1060-02 (100 000 op/s)
02.03.1983–06.1989	ES EVM -1055M (450 000 op/s)
25.10.1989–10.1992	ES EVM -1037 (4 000 000 op/s)

Table 6. (*continued*)

Minicomputers	
23.03.1981–06.1989	CM-4 (180 000 op/s)
24.03.1985–05.1989	IZOT-1016S
30.04.1987–1990	IZOT-1055S
07.06.1989–1990	IZOT-1080 (4 500 000 op/s)
Personal computers	
1985–1993	Acorn (BBC education class, UK)
1986–1993	ISKRA-226 , DVK, KUVT-86, BK-0010, Robotron – 1715 (Eastern Germany), Yamaha (Japan)
1989–1993	ISKRA-1030, IBM XT, AT, Mazovia, PS2

Soviet economic planners decided to use the ES EVM design (IBM clones). Prominent Soviet computer scientists had criticized these ideas and suggested instead choosing one of the Soviet indigenous designs, such as BESM or Minsk. The first works on the cloning began in 1968; production started in 1972. After Latvia regained independence, operation of ES EVM was suspended in 1992 due to costly service – electricity and a great maintenance staff.

All together transition from BESM to ES EM and mini computers did not affect the institute's research, nor the data centre's development, because ES EVM were not smaller in their volume; they were even more exacting as for data centre facility's provision.

In the existence of the data centre, there was decadence for 5 years in the end of 1980s, because mainframe computers were disassembled and other computing resources of such scale were not set up.

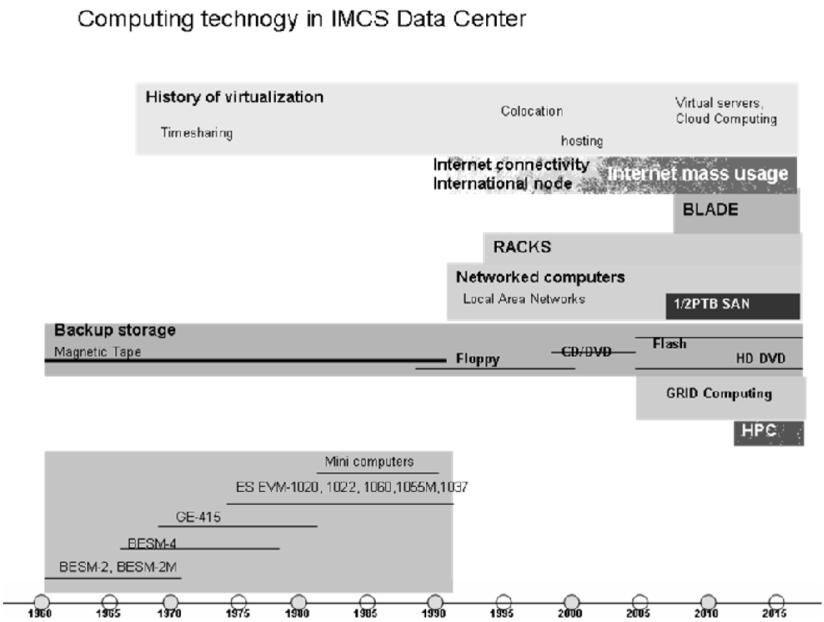


Fig. 5. Computing Technology in IMCS Data Centre

Technological reform of the data centre began when Latvia regained its independence and introduction of the internet in 1992. In the data centre there was disposed and further developed an international internet node with appropriate server connection; but in the mid-1990s collocation services were offered, but in 2000 – hosting services. This allowed to rationally use the early data centre's premises and practical knowledge of scientists. If we are talking about the data centre provisions functionality (power, HVAC, fire, security), then the best knowledge was gained with requirement ensuring of GE-415. By the beginning of the 1990s, completely new equipment and modern technologies were used, but the functional equipment of the data centre was maintained. The latest in IMCS UL was introduced the structured cabling system (only from 2006). Mentioned statements are confirmed in the picture above.

6 Conclusions

There are standards and theory about data centres in accordance with which data centres are classified. The latest generation of computing technique's maintenance requirements for environment are increasing, and as a result, every data centre in its development is tending to become the highest category data centre. To ensure such environment it is also necessary to modernize the components that form it, therefore investments in environment development are increasing. Scientific institutions do not have such resources to create the highest category data centres, although the development of data centres in them is necessary; as well as it is necessary to ensure sustainable exploitation of computing resources, it is not critical to turn-off data centre, e.g. during the time environment maintenance equipment is being changed.

Literature

1. Balodis, R., Borzovs, J., Opmane, I., Skuja, A., Ziemele, E.: Research Directions Profile in the Computing Museum of the Institute of Mathematics and Computer Science, University of Latvia (IMCS). In: Impagliazzo, J., Lundin, P., Wangler, B. (eds.) HiNC3. IFIP AICT, vol. 350, pp. 453–461. Springer, Heidelberg (2011)
2. Р. Балодис, И. Опмане. Институт и три социально- технологические волны ИТ.стр. 36-40,in International Federation for Information Processing Yaroslav-the-Wise Novgorod State University A.P. Ershov Institute of Informatics Systems, Siberian Branch of the Russian Academy of Sciences Computer History Museum, CA, USA Scientific Council of the Russian Virtual Computer Museum Polytechnic Museum, Moscow SORUCOM 2011 PROCEEDINGS Second International (2011)

Early Italian Computing Machines and Their Inventors

Silvio Hénin

silvio.henin@fastwebnet.it

Abstract. Nineteen centuries of Italian inventors and inventions in the field of aids to arithmetic and algebraic computing, before the electronic computer era, are reviewed; most of them forgotten or still unknown. Not meant to be a complete or ultimate treatise on the topic, this paper hopefully wants to be a starting point for more multidisciplinary research of Italian history of technology.

Keywords: mechanical calculating machines, history, Italy, 1st – 20th century.

1 Introduction

Taking into account all kinds of arithmetical and mathematical instruments ever invented, since Sumerian arithmetic tablets (c. 2,500 BCE) and Chinese and Greek abacus (5th – 4th Century BCE), tools to simplify the annoying calculation labor appeared rather soon in the history of civilization (Campbell-Kelly 2003 pp. 19-43; Needham 1959 pp. 29-63; Schärlig 2001, pp. 61-104). To be sure, multiplication tables and abaci were adequate for the limited calculation need of merchants and court bookkeepers until seventeenth century, when the Scientific Revolution and growing government bureaucracies asked for much more intricate and precise computing to be carried out, inducing the mathematician-philosopher Gottfried Leibniz to cry out "[...] *it is beneath the dignity of excellent men to waste their time in calculation when any peasant could do the work just as accurately with the aid of a machine"* and to invent one of the first mechanical calculators in 1672, the third one after Schickard's 'calculating clock' and Pascal's 'pascaline' .

In the following two centuries a number of scientists and artisans devoted themselves to design and build computing machinery, slowly transforming those contrivances from '*useless toys … doomed to failure*' (Williams 1990, p. 50) — though often aesthetically beautiful and of great ingenuity — to useful and reliable instruments, mass produced and sold in quantity. It was the coming of the industrial revolution and industrial capitalism in the United States, last decade of nineteenth century, which produced the 'crisis of control' (Beniger 1986), in turn creating a market for reliable, fast and error-free way to process large quantities of numerical and statistical data. In fact, despite the foregoing example in France by Thomas de Colmar, it was in the US that the data-processing industry began to bloom, in a few decades becoming a successful and highly profitable business (Williams 1997, pp. 145-151).

In the two centuries (c. 1650 – c. 1850) of 'incubation' mentioned above, more than one hundred prototypes of mechanical calculating machines have been proposed by an almost equal number of inventors, in all European countries. Particularly prolific in

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 204–230, 2012.

this field were France and the German-speaking countries[1], followed at a distance by the United Kingdom (Campbell-Kelly 1989, pp. 6-7). In this panorama, only a score of Italian inventors and inventions are recorded and even fewer have been mentioned in international scholarly literature; thus it seems fair to revive their remembrance. Though modern Italy, as unified state/nation, is only one century and a half old (1861), what follows is an account of inventors and inventions belonging to the Italian cultural environment, regardless of coeval political situation.

2 The Roman Pocket Abacus

The abacus is the oldest and the longest lasting calculating device, possibly anticipating consistent number naming and writing (Menninger 1992, pp. 295-388); the independent invention of variously shaped abaci in different ancient cultures is evidence of both ingenuity and need. In the Etruscan and Roman culture 'table abaci' (akin to Greek Salami's table of 300 BCE) were largely diffused — due to the cumbersome, almost impossible, procedure to reckon with Roman numerals — and several Latin writers (e.g. Cicero, Martial, Juvenal and Pliny) mentioned the use of loose *calculi* (pebbles, beads), laid on wooden tablets to help computing (Schärlig 2006, pp. 70-71) . Pictures of table abaci are recorded on an Etruscan cameo, on a Roman pier found in Trier (Germany) and on a bas-relief of the Capitoline Museums in Rome (Schärlig 2006, pp. 72-74). Unfortunately, no relics of Roman wooden or marble counting tables have yet been found.

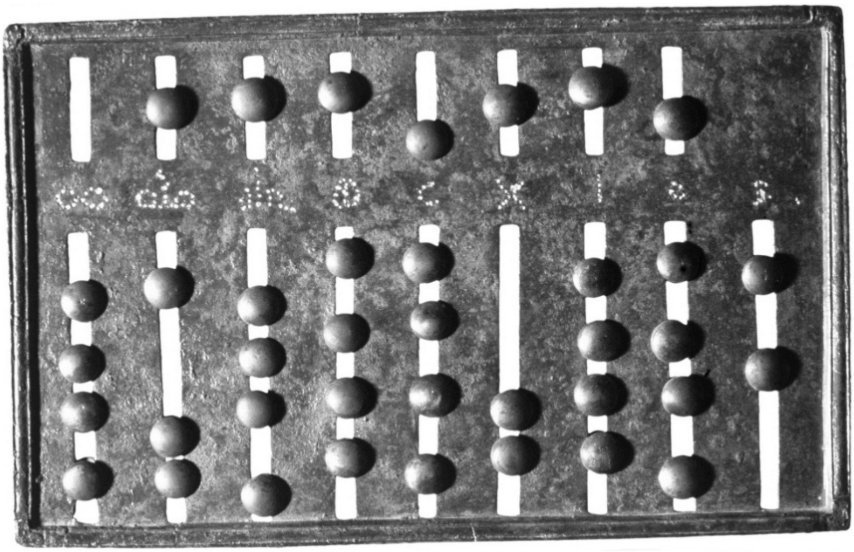


Fig. 1. Roman pocket abacus (1st century CE) (Museo Nazionale Romano, Ministero per i Beni Culturali, Soprintendenza Speciale per i Beni Archeologici di Roma)

[1] Although a loose German League of 39 sovereign states was founded in 1814 (Deutscher Bund), Germany became a unified nation in 1871.

Specimens of a more interesting, perhaps unique, sort of Roman abacus, the 'portable' or 'pocket' abacus, came to light in 16th-17th century. Two of them have been thoroughly described (Welser 1594, pp. 241-243, Pignoria 1613, pp. 170-173) and a third, belonging to Ursinus[2], just briefly mentioned (Smith 1925, p. 165). Nowadays three specimens only are known to still exist: one at the *Museo Nazionale Romano* in Rome, the second at the *Bibliothéque National, Cabinet des medailles,* in Paris, and the third at the *Museo Archeologico Regionale* in Aosta (Northern Italy) recently found in a burial ground (Martin 1989, Fellman 1983). Comparing Welser's and Pignoria's descriptions with the three extant relics, Renaissance's specimens are hardly the ones collected in the European museums nowadays[3].

Roman pocket abaci have been dated around 1st – 2nd century CE, possibly used for calculation with Roman coin '*as'* and its fractions (1/12, *uncia*; 1/24, *semuncia*; 1/48, *sicilico*; 1/72, *duella*); their owners were probably local administrators or imperial travelling inspectors. Pocket abaci are little bronze tablet (10 x 7 cm) with pairs of slots holding sliding beads — this arrangement prevents the counters to be lost, a solution to safely carry them around in pockets or bags. Each slot pair represents an order of magnitude (1 – 10 -100 -…) each with a lower 'units' slot (with four counters) and an upper 'fives' slot (with one counter); thus to represent the number 8 one has to move one 'fives' counter and three 'units' counters up. On the right side other three slots were used for *as'* fractions, with one or two counters each. The strong similarity, in terms of architecture and mode of use, with old and modern Japanese *soroban* and Chinese *suan-pan* is impressive, but evidence of reciprocal influence can hardly be proved; though irregular and indirect contacts between China and the Roman Empire have been recorded (e.g. for silk trade) no evidence of direct cultural cross-influence has yet been found.

3 The Renaissance: Analog Devices

Since 13th-14th century, the study of mathematics, particularly algebra, began to flourish in Italy thanks to the discovery and the translation into Latin of ancient Greek and Hellenistic treatises and of Middle Age Islamic scholars. Mathematicians like Leonardo Pisano (aka Fibonacci), Luca Pacioli, Niccolò Tartaglia, Gerolamo Cardano, Raffaele Bombelli kept Italian mathematics well in advance of other European countries for the next two centuries. Also in applied mathematics Italian schools excelled; as George Ifrah recounts : *"A wealthy German merchant seeking to provide his son with a good business education, consulted a learned man as to which European institution offered the best training. 'If you only want him to be able to cope with additions and subtractions,* the expert replied*, 'then any French or German University will do. But if you are intent on your son going on to multiplication or division, then you will have to send him to Italy*" (Ifrah 2000, p. 577). Ancient technology was also

---

[2] Possibly Fulvius Ursinus (1529-1600), antiquarian and philologist, librarian of Cardinal Farnese in Rome.

[3] Another alleged Roman abacus (2nd – 5th century CE), in ivory, is reported in the European IBM Collection in Paris (http://www-sop.inria.fr/amisa/piece2Eng.html, Accessed January 12, 2012), but all my attempts to gather more information from IBM remained unanswered.

re-discovered (e.g. Hero of Alexandria's), design and construction of mathematical and astronomical instruments flourished and able instrument makers were prized at sovereigns' courts.

A number of practical instruments were invented to help sailors, gunners, architects and painters (Williams 1997, pp. 66-83; Favaro 1907). One instrument was particularly appreciated for ease of use and practical applications: the *proportion compass* or *reduction compass,* an analog device with two brass arms hinged together. Indexed scales were sometimes engraved, either on a separate rule or on the arms, to solve trigonometric and geometrical problems, thanks to the proportionality between homologous sides of triangles; in this case the instrument was better known as a '*sector*'. Sectors helped calculations for navigation, survey and gunnery, until the engineers' *slide rule* replaced it (Williams 1997, pp. 75-83). The origin of the proportion compass can be traced back to **Leonardo da Vinci** (1452-1519) (Camerota 2000, pp. 14-15), afterwards a number of diverse types for different applications were proposed by several inventors all over Europe. The mathematician **Niccolò Tartaglia** (1499-1557) invented two gunner's sectors in 1560 for gun aiming and evaluation of target distance and height. Another more advanced Italian sector-like instrument is ascribable to **Federico Commandino** (1509-1575), mathematician known for the translation of several Greek treatises by Archimedes, Aristarchus of Samos, Euclid, Ptolemy and Hero of Alexandria. Commandino designed a 'polymetric compass' (c. 1568) (Bianca 1982) and one of his friends, the anatomist Bartolomeo Eustachi asked him to build one for his anatomical studies on the 'cartography' of human body (Andretta 2009, pp. 93-124) . One of Commandino's pupils was Marquis **Guidobaldo Del Monte** (1545 –1607) who studied at the University of Padua, was appointed supervisor of Grand Duchy of Tuscany's munitions and published several books on mathematics, astronomy, perspective and mechanics; he was also Galileo's friend and helped him to obtain the chair of mathematics at the University of Pisa. Del Monte built a sector (Drake 1988, pp 76-77) and a mechanical calculator that, by a gear train, converted any fraction of a degree into minutes and seconds (Gamba 1988, pp. 85-87, Camerota 2000, pp. 71-72). A previous inventor was **Fabrizio Mordente** (1532-1608) who at the age of twenty left Italy and travelled for ten years in North Africa, Middle East, India and most of Europe, where he paid visit to several scholars and instrument makers. Eventually he worked in the service of Emperor Maximillian II in Vienna, of Rudolph II in Prague and of Alessandro Farnese, Duke of Parma and Piacenza. In 1567 Mordente began to design a simple reduction compass to divide lines and arcs, perhaps influencing del Monte and Commandino, and improved it in the following decades with a better instrument of 1572 (*compasso magistrale*), an even better one in 1585 and an ultimate type in 1591 known as 'eight-point compass' (Boffito 1931; Camerota 2000; Camerota 2003). The latter had four sliding and four fixed points and a detached rule with proportional scales. Italian philosopher Giordano Bruno prized Mordente's compass and published *De Mordentii circino* (On Mordente's sector, Paris, 1586) where he postulated Mordente's mathematical approach could help confute Aristotelian hypothesis of the incommensurability of infinitesimal quantities (Aquilecchia 1957, Camerota 2000, pp. 83-105].

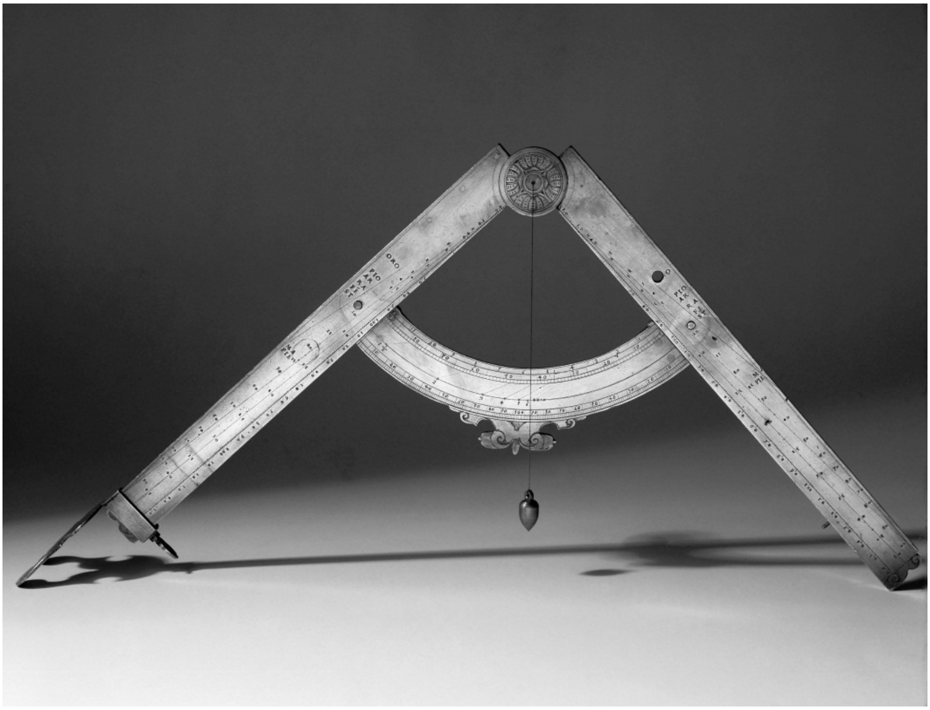


Fig. 2. Galileo Galilei's *Compasso Geometrico-Militare* (1597-1606) (Museo Galileo, Florence, - Photography Franca Principe)

Certainly the most known Italian sector is the one invented by **Galileo Galilei** (1564-1642) (Favaro 1907; Geymonat 1970, pp. 189-222; Vergara-Caffarelli 1992; Drake 1999, pp. 5-32) called '*compasso geometrico militare*' (geometric-military compass). Galileo designed it in 1597, while professor of mathematics at the University of Padua, and began to build some with the help of Marcantonio Mazzoleni, able instrument maker, to teach his student practical applications of geometry . In the following years Galileo's *compasso* was modified transforming it from a 'Swiss knife' for practical use — like most other coeval sectors — to a 'general purpose analog computer' for abstract quantities. More than 100 specimens have been built in the following ten years and 30 of them were presented to European sovereigns (Vergara-Caffarelli 2006), several others sold to Galileo's students. In 1606 Galileo published a 'user manual' for his *compasso* (Galilei 1606) in 60 copies — his first printed publication — to help using the contrivance. The success was impressive: soon the number of booklet and *compassi* available were insufficient to satisfy the request, though several instrument makers had started to produce copies, sometimes rough and useless (Vergara-Caffarelli 2006). Already in 1607 Baldassarre Capra (c. 1580-1626) tried to claim credit for the invention of the instrument in his '*Usus et fabrica circini cuiusdam proportionis',* but Galileo strenuously counteracted and Capra was censored by the Paduan University board. Galileo's invention assured him the return to the University of Pisa in 1610 and the protection of Grand Duke Cosimo II de' Medici.

Galileo's priority for the invention of the sector is still debated nowadays (Favaro 1883, p. 166; Drake 1999, p 7-12; Vergara-Caffarelli 1992). As we have seen, accounts of history of mathematics and technology mention several 16th-17th century inventors of 'sectors' in Italy[4] and in Europe[5]; multiple independent inventions are common in the history of technology (Merton 1973, pp. 343–70.). Indeed the idea was 'in the air' for a long time and many inventors followed it; moreover it is quite difficult to distinguish between marginal improvements and true innovations and to argue of possible plagiarism seems even harder, apart from Capra's gullible attempt. Thomas Hood's sector (1598) is possibly Galileo invention's nearest thing, but any reciprocal influence can be ruled out (Drake 1999, p. 7). Certainly Galileo's *compasso* was the first 'calculating sector' in use for a variety of problems, it shows several important and useful improvements over previous and contemporary models: a higher number of scales, a better precision and simplicity of use; nevertheless, as all its competitors, remained a low precision tool — due to the small size precision was two-figures at the best — more suited for education than for practical application, as Galileo himself stated in his writings (Vergara-Caffarelli 1992).

4 17th – 18th Century: Early Mechanical Digital Calculators

4.1 The Mysterious N. 3179

In the first half of 17th Century mechanical digital calculators to execute the four arithmetic operations appeared in Europe, achievement made possible thanks to the advancement of clock technology[6]. No history of computing fails to mention Blaise Pascal's and Gottfried Wilhelm Leibniz's inventions, the *pascaline* (1645, for addition and subtraction) and the *stepped reckoner* (1673 for multiplication and division) respectively. A previous contrivance by Wilhelm Schickard, the *rechenuhr* (calculating clock) of 1623 is sometimes forgotten, but it cannot have influenced Pascal and Leibniz, having been lost in the Thirty Years War and re-discovered in 1935 (Williams 1997, pp. 118-45). In the following two centuries more than one hundred different attempts were made by an almost equal number of astronomers, mathematicians, mechanics, self-taught scientists and noblemen; most of them were unreliable, fragile, expensive and could not compete with paper-and-pen computing.

---

[4] Besides the names mentioned above: Baldassarre Lanci (1557), Antonio Bianchini (1564), Carlo Teti (1575), Antonio Lupicini (1582), Latino Orsini (1583), Ostilio Ricci (1590), Ottavio Fabri (1598). (Camerota, 2000).

[5] Christopher Schissler (1566,1580), Jost Bürgi (1588), Thomas Hood (1598), Christopher Scheiner (1603), Michel Coignet (1610) (Drake 1999, p. 7, Camerota, 2000, pp. 124-125).

[6] Early mechanical clocks were known in Italy in 14th century, in Milan (1335) and Padua (1364).

In *Museo Galileo*'s collection in Florence[7] visitors can admire a beautiful and bizarre brass object in the shape of baroque frieze with small dials on its face. The tag says: 'N. 3179, **Tito Livio Burattini (attr.)** Calculating machine, First half of 17th cent.' The machine has nine main dials, the upper six numbered 0-9, the lower three 0-6, 0-19 and 0-11 respectively. Paired with each main dial is a smaller one, likewise numbered. The instrument is a 'money adder' of the kind Samuel Morland (1625-1695)[8] built in London in 1673 and described in a booklet (Morland 1673; Williams 1997, pp. 136-140). Unlike *pascaline*, both N. 3179 and Morland's machines have no mechanism to transmit 'carry' from one digit to the next, but only auxiliary small dials moving one position at every turn of the respective main wheel. The 10 positions dials are for *lire* (pounds) and the 20 and 12 positions dials are for *soldi* (shillings) and *denari* (pence); the uncharacteristic 7 position wheel was possibly used for the conversion of Florentine *lira* to *ducato*[9] (1 *ducato* = 7 *lire*) (Cipolla 1987, p. 131) or of German *schillings* to *gulden* (1 *gulden* = 7 *schillings*) (Schäring 2003, p. 94).

Fig. 3. Dial adder attributed to Tito Livio Burattini (Museo Galileo, Florence, - Photography Franca Principe)

[7] Now 'Museo Galileo', Florence, http://www.museogalileo.it/en/index.html, Accessed January 14, 2012.

[8] Morland gifted two mathematical instruments to Cosimo III De' Medici, now at the Museo Galileo in Florence.

[9] *Ducato* was the currency in Florence, *lira* was a fictitious unit for bookkeeping and money exchange.

Tito Livio Burattini (1617 – 1681) was an Italian polymath, egyptologist, inventor, instrument maker, architect and engineer who left his birthplace — Agordo, a small village in the North of Italy — to travel to Egypt and Germany and eventually settled in Poland at the court of King Wladyslaw IV where he was appointed chairman of the Polish mint (Tancon 2005). In 1675 Burattini suggested a universal length unit he dubbed *metro cattolico* (universal meter),[10] well in advance of the French metric revolution of 1793. He also built a flying machine, in every respect a glider, and made lenses for telescopes and microscopes, presenting them to Cardinal Leopoldo de' Medici. Burattini corresponded with many scholars and philosophers all over Europe and kept himself informed of scientific and technical achievements.

In 1647, while in Krakow, Burattini had the opportunity to scrutinize one *pascaline*, gifted by Pascal to King Wladyslaw's wife, Maria Luisa Gonzaga. He then decided to adventure upon building one himself; in 1658 he succeeded to finish a model with eight dials 'like Pascal's contrivance' (Targosz 1992, pp. 164-165) and sent it to Grand Duke Ferdinando II de' Medici. In fact, two letters from Alfonso Borelli[11] to Cardinal Leopoldo de Medici, Ferdinando's brother, dated November 15 and December 1st, 1658, mention an '*istrumento o cassettina numeraria'* [instrument or small box for numbers] sent by Burattini. Burattini's gift was wrote down on the inventory of the de' Medici's collection[12] in 1660 : 'N.585 *in data 1659 uno strumento di ottone per fare abaco*[13] *che ha otto ruote, lungo 3/4 largo 1/5 a S.A: serenissima donato da Tito Livio Burattini il 22 giugno*' [N. 585. In 1659 a brass instrument for calculation with 8 wheels, 3/4 long and 1/5 wide[14] presented to His Serene Highness [Ferdinando II de' Medici] by Tito Livio Burattini on June, 22]. The same description is repeated in later catalogues (1704 and 1738). In 2007 the historian Vanessa Ratcliff noticed (Ratcliff 2007) that the object recorded on the 1660 inventory do not match with N.3179: the number of wheels of the latter is nine (eighteen if the nine small dials are taken into account), instead of eight. Moreover the size of N.3179 (20 x 10 cm) do not fit with those recorded in 1660 (44 x 12 cm). It is also worth noting that the 1660s specimen has, more or less, the size of the eight-wheel *pascalines* (36 x12 cm) (Marguin 1994, p. 62) Burattini tried to reproduce. Interestingly enough in a catalogue of 1779 [*Real Gabinetto*, 1776-1779] the description is without a doubt different : *"Una macchinetta forse aritmetica di due lastre di ottone centinate che racchiudono 18 cerchi tra grandi e piccoli, numerati, imperniati, e da muoversi a mena dito. La macchinetta ha la faccia dorata, ed è lunga nel più pollici 7.3.."* [- a small machine, perhaps arithmetic, made of two ribbed brass plates that enclose 18 large and small circles, numbered, hinged and to be operated with fingers. The machine has golden face and is 7.3 inches long (ca. 21 cm)]. The new description fits well with the object N.3179. It is noteworthy that in 1737 the last Medici, Gian Gastone, died and

[10] Equivalent to the length of a free one second pendulum, as the English philosopher John Wilkins proposed in 1668.

[11] Giovanni Alfonso Borelli (1608-1679) was a mathematician and philosopher in Rome.

[12] The Medicean collection was started by Grand Duke Cosimo I in 16th century.

[13] In Renaissance Italian '*fare abaco*' meant 'to compute', thus synonym of *arithmetics*, even if abacus was not used.

[14] Taking the Florentine unit (braccio da panno = 58,3 cm), it is cm 43,8 x 11,7. Even with different units the aspect ratio is inconsistent with N. 1379.

the Grand Duchy of Tuscany was entrusted to the House of Habsburg-Lorraine; nine years later almost all the Medicean scientific collection was moved to the Imperial Museum of Physics in Vienna,[15] never to return (Bedini 1995). In the meantime the Florence collection was enriched with hundreds of pieces from the Lorrainese Chamber of Physics of Lunéville, under the supervision of Philip Vayringe[16]. Thus we can conclude that Burattini certainly made a pascaline-like calculator and sent it to Florence where it remained for more than a century, but sometime before 1779 it disappeared (sent to Vienna ?) and was replaced with the actual N.3179. As for the origin of the latter, Ratcliff suggests that it can be attributed to an unknown Italian maker or to Samuel Morland — due to the similarity with his money adders — but a third possibility is that N.3179 comes from Lunéville, perhaps due to Vayringe or another Lorrainese maker.

4.2 Giovanni Poleni's Calculator (1709)

In December 1709 a booklet was printed in Venice bearing the title *Miscellanea* (Poleni 1709); one of the three chapters is the description of a calculating machine invented by the author, Marquis **Giovanni Poleni** (1683-1761). Poleni was born in the Republic of Venice, son of a wealthy man titled marquis by the Emperor Leopold I of Augsburg. Not much interested in political career, Poleni preferred the study of physics, architecture and mathematics. In 1709, thanks to his book, a sort of doctoral thesis, Poleni was appointed professor of astronomy and meteorology at the University of Padua, where he worked until his death as professor of physics, mathematics, experimental physics and naval engineering. In Padua he founded one of the first European laboratories of physics, certainly the best equipped with more than 400 fine instruments. Marquis Poleni was renowned in European scientific *milieu* and was a fellow of the *Royal Society,* the French *Académie des Sciences* and the *Prussian Academy of Sciences* (Soppelsa 1963; Soppelsa 1983).

At the age of twenty-four Poleni heard news of Pascal's and Newton's calculators ' from scholars in person and from their writings' [17] and, though having no drawings nor technical description of the two previous inventions, decided to design and build an original one. After some attempt, the machine was finished in 1709. Poleni's calculator — in the shape of a big grandfather clock, in wood and iron — is a four-operation device, enabling the user to perform additions, subtractions (resorting to complement-to-nine additions), multiplications (by multiple additions), and divisions (by multiple subtractions) (Poleni 1709; Soresini 1971, pp. 95-105; Bonfanti 1988, Hénin 2009). To perform multiplication and division Poleni followed, independently, Leibniz's idea of a device where multiplicand and divisor could be stored and to be added or subtracted a fixed number of times. Differently from Leibniz' 'stepped drum', the marquis conceived the 'pinwheel', a wheel with an

[15] Francesco II Stefano Grand Duke of Tuscany was also Emperor of the Holy Roman Empire, and King of Austria.

[16] Philip Vayringe (1684-1746) was court mechanician and clockmaker of the Duke of Lorraine and professor of experimental physics in Lunéville. Vayringe followed the Duke to Florence where he took care of Medicean collections.

[17] Perhaps from John Wallis' '*Opera mathematica*' (1695) (Gennari, 1839).

adjustable number of teeth (0-9) that could be raised or lowered to set the figure to be operated upon. Poleni's machine had a three-sector wheel, thus enabling to set a three-digit figure, and a clockwork mechanism — with verge escapement — to drive the entire machine by the energy of a falling weight. The pinwheel, albeit with different design, was implemented in a number of calculators in the following three centuries, e.g. the popular Baldwin-Odhner architecture (Williams 1997, pp. 146-149, Hénin 2009). Poleni's Invention was acknowledged in Europe thanks to its description published in 1727 in the first comprehensive review on mathematical instruments by Jacob Leupold (Leupold 1727, pp. 27-35).

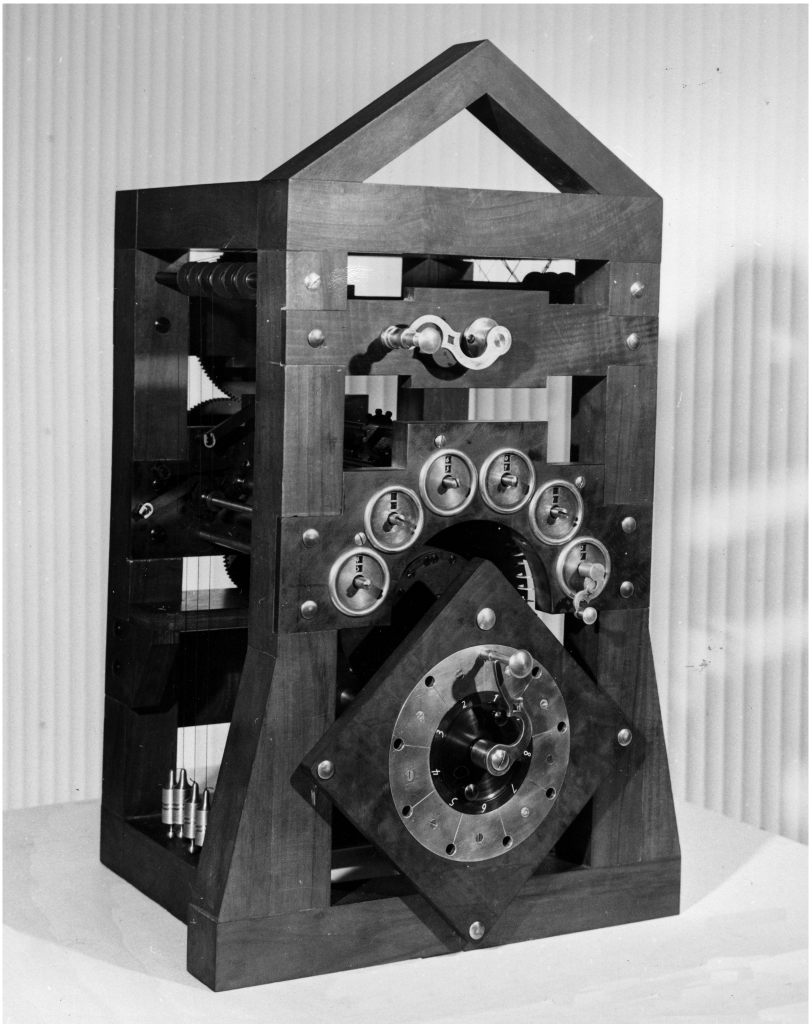


Fig. 4. Giovanni Poleni's calculator (1709). Replica of 1959 (Courtesy of Museo Nazionale della Scienza e della Tecnologia "Leonardo da Vinci", Milan)

As Poleni's biographers recount (e.g. De Fouchy 1763, pp. 152-153), after some years the Italian physicist was informed of a similar, but better machine built in 1727 by the Austrian Anton Braun, court mechanic of Emperor Karl VI in Vienna; having known this Poleni 'smashed his machine into pieces and never built it again'. Around 1720 Braun worked in the Grand Duky of Milan (North Italy) as a surveyor, under the supervision of the imperial engineer Johann Jacob Marinoni, and possibly got information of Poleni's machine (Marinoni was in correspondence with Poleni and perhaps visited him in Venice) (Habacher 1960). Nevertheless Braun's calculator, although resorting to pinwheels, was for many other features quite different from Poleni's: much smaller, all metal, hand driven; thus a possible plagiarism, but of the general idea, can be ruled out. Moreover Poleni's temper was neither cantankerous nor vengeful and his reaction could be ascribed to sincere appreciation of the technical superiority of Braun's contrivance. The lost Poleni's machine was reconstructed in 1959 by Franco Soresini thanks to the help of IBM Italy and *Museo Nazionale della Scienza e della Tecnologia* in Milan (Soresini 1991); after another 50 years the replica was restored and successfully put into action (Hénin 2009).

4.3 Bernardo Facini's Slide Rule (1714)

Bernardo Facini (1665-1731) was an astronomer, mathematician and skilled instrument designer who worked in Venice. To him several mathematic instruments for navigation, survey and time measurement have been attributed, some of them still in museums in Europe and United States (Anastasio 1994). In 1714 Facini designed a circular logarithmic slide-rule, now at the Adler Planetarium in Chicago. After the invention of logarithms by John Napier in 1614 and their perfection by Henry Briggs, the idea of a device exploiting their properties to mechanize computing — as alternative to logarithm tables — soon arose: the first by Edmund Gunter (1620), followed by Richard Delamain's (1630)[18] and William Oughtred's (1632), both of circular shape (Williams 1997, pp. 105-110). The first circular slide rule with a spiral scale, to increase effective length (and precision) without increasing overall size too much, is due to Milburne (1650) (Horsburgh 1914, p. 155). Facini's design is a very simple one: a brass disc of 22 cm diameter brings a number scale (1 to 10) and a logarithm scale (0 to 1), respectively on the two sides. Two movable pairs of dividers, one for each side, are hinged in the disc center to help setting numbers. Scale divisions are such that a three-four digit precision can be obtained, with the help of a transverse vernier engraved on number scale. (Righini 1980). Thanks to the spiral scale, Facini's device is equivalent to a 100 cm long linear rule.

5 19th Century: Advanced Mechanical Calculators

In Eighteenth Century Italian science and technology underwent a slowdown; some historians try to explain the phenomenon with Galileo's conviction for heresy, but this cannot be the only reason, nor the main: political division and a general state of

[18] The priority, and possible plagiarism, between Delamain's and Oughtred's inventions is still debated. Oughtred was helped by his pupil William Foster.

recession also contributed. The beginning of 19th century saw, luckily, a revival, particularly in technology and applied science, thanks to the founding of several new institutions for education, training and promotion of technical and industrial advancement. Almost all these bodies — akin to the French *Conservatoire des Arts et Métiers* – were established by Napoleonic governments but, happily enough, survived Napoleon's fall and 1814 European Restoration. Those institutes flourished in particular in Lombard-Venetian Kingdom and in Grand-Duchy of Tuscany.

5.1 Luigi Torchi's Keyboard and Direct-Multiplication Calculator (1834)

All we know about Luigi Torchi's life (1812-?) is that he was an uneducated mill carpenter working in the suburbs of Milan. In 1834 Torchi was awarded the Golden Medal by the Royal-Imperial Lombard Institute of Sciences, Letters and Arts for the invention of a calculating machine (Hénin 2010a). In the following years Torchi invented other two contrivances: a horse-less cart that, moving along the canal towing-path, dragged a barge, exploiting the energy of the canal's stream; the invention was acknowledged in Italy and France and was awarded a silver medal by the Lombard Institute in 1837. In 1858 Torchi also built an improved pendulum level. Two descriptions only of Torchi's calculating machine have been found: the jury's handwritten report[19] for the prize and an anonymous article on the magazine *La Fama* (La Fama 1840), where an artist impression of the machine was printed. From the two sources and the picture an attempt to understand the principles of action has been made (Hénin 2010a). First of all the calculator makes use of an extended keyboard — four rows of nine keys —to enter numbers up to 9999. The earliest keyboard calculating machines reported in literature are those by Jean-Baptiste Schwilgué (1844) (Roegel 2008), Du Bois D. Parmelee (1850), Victor Schilt (1851), Thomas Hill (1857) (Turck 1972), all were 'column adders' conceived to carry out additions of single-digit addenda without tens-carry mechanism, differently from Torchi's, which can sum up four-digit numbers. The second and more impressive feature is the ability to perform direct multiplications, without resorting to multiple additions. In fact, the jury's report state that "... *a device is conceived that gives the <u>product of that figure by each of the nine simple</u> numbers* ... [the multiplication] *is obtained by means of pinions* ... "[author's underscore], in accordance with the mode of use described in *La Fama* (La Fama 1840). Early direct multiplication machines were conceived forty years later by Edmund Barbour in 1872, followed by Ramon Verea (1878), Léon Bollée (1889) and Otto Steiger (1892) (Turck 1972, p. 181; Soresini 1971, pp. 125-132; Marguin 1994, pp.130-134).

Torchi's calculator gathered much local interest: a sum of 1000 lire was appropriated by the government for an improved model in metal — the prototype being in wood — and the Milanese Census Board asked for another to help land surveying, but the inventor was possibly not skilled enough to satisfy the requests. The machine was publicly exhibited in the next decades until found by astronomer Giovanni Schiaparelli in 1872, but "*lacking many pieces, almost all the front part containing the device*

[19] The whole dossier is collected at the Historical Archive of the Lumbard Institute, in Milan.

*for the preparati*on *and the reading of the results is missing"* (Henin 2010a). Built with perishable materials and already damaged in 1872, the calculator was probably shattered or disposed and no traces of it have yet been found.

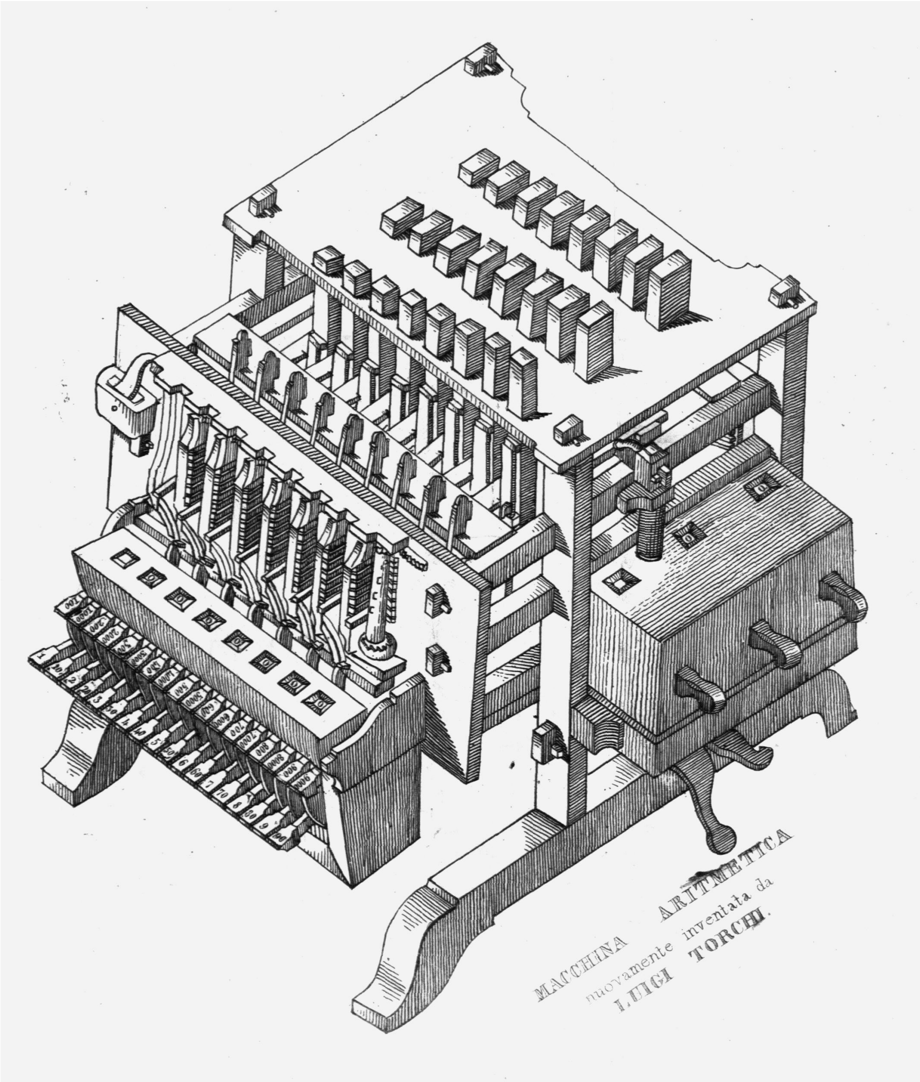


Fig. 5. Luigi Torchi's direct mutiplier (1834) (La Fama 1840)

5.2 Tito Gonnella's Analog and Digital Machines (1825-1859)

In another Italian State, the Grand Duky of Tuscany, technology was appreciated and technical development stimulated by an active network of institutions and schools; Florence hosted also several skilled instruments makers. One of those was Tito Gonnella (1794- ca. 1867), professor of mathematics and mechanics at *Accademia di Belle Arti* in Florence (Borchi 1997), who made also an improved Newtonian reflection telescope. In 1825 Gonnella published a paper (Gonnella 1825) describing an early planimeter or 'machine to square plain surfaces' to calculate the surface area of any closed figure. Planimeters were the most common instrument available in the 19th Century; they gave the area of irregular figures, useful for surveyors, and were soon employed as mechanical integrators for mathematical functions (Carse 1982; Bromley 1990).

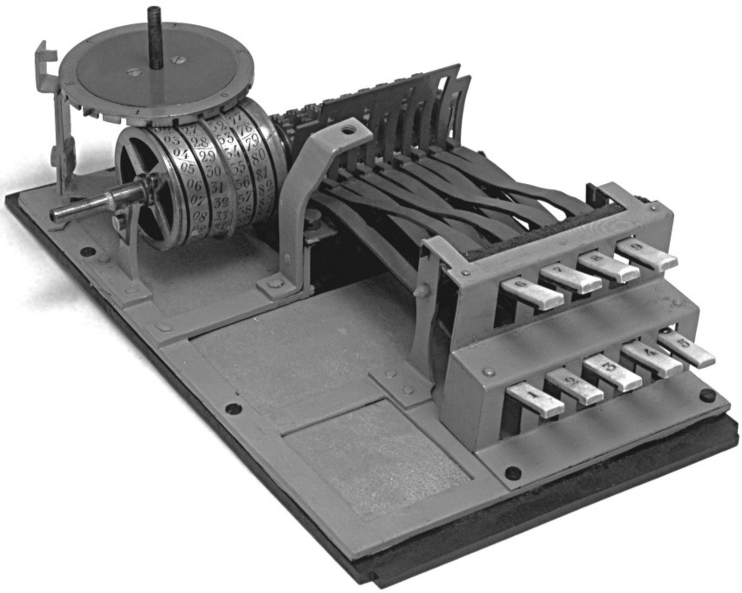


Fig. 6. Tito Gonnella's keyboard adder (1859) (Museo Galileo, Florence, - Photography Franca Principe)

An earlier model of planimeter was invented by Johann Martin Hermann in 1814 but remained almost unnoticed. Gonnella's design was based on the so called 'disk-and-wheel' mechanism, after having tried the 'disk-and-cone' and discarded it for the lack of precision. In the following decades, Gonnella exhibited his planimeter in various instances, e.g. the 1851 Great Exhibition in London, where his invention was awarded with the Council Medal, and Paris *Exposition Universelle* in 1855. Gonnella had only one planimeter built in Florence, then he realized that the construction of his instrument required the precision of Swiss manufacturers and sent his drawings to some of them, amongst whom Johannes Oppikofer, who designed a planimeter in 1827. Henrici

(Henrici 1894) suspects that Oppikofer's device was 'strongly influenced' by Gonnella's. That of planimeters and integrators is another case of multiple, more or less independent, inventions; in London Great Exhibition four planimeters were shown, by John Sang, Kaspar Wetli, Jean Antoine Laur and Ausfeld[20].

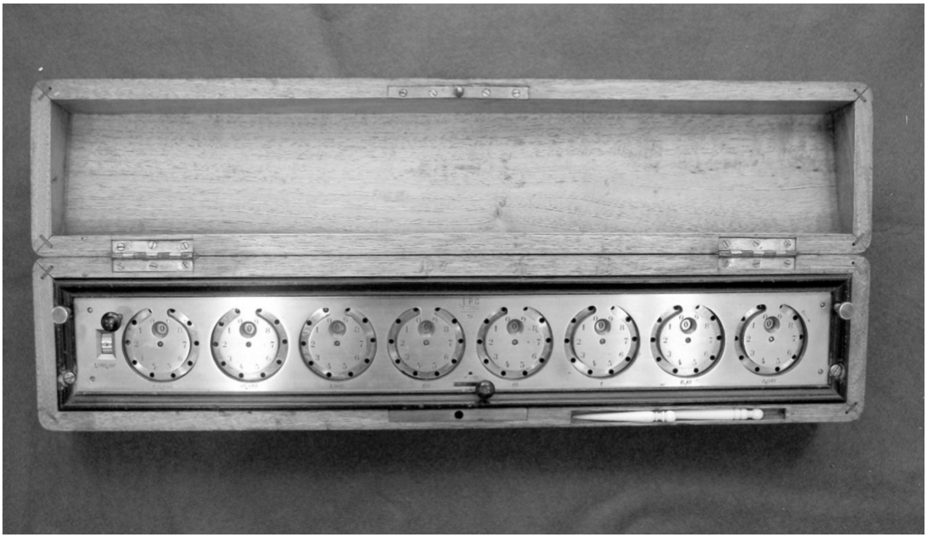


Fig. 7. Tito Gonnella's dial adder (1859) (courtesy of Arithmeum, Rheinische Friedrich-Wilhelms-Universität, Bonn)

Gonnella's other contributes to calculator technology were two small machines he described in a booklet (Gonnella 1859). The first is a dial adder, akin to Pascal's invention and to many others, three exemplars were finished in 1857 and shown in public. The carry transfer mechanism is unique and original, but still unreliable: in case of ripple carry (e.g. adding 99999+1) it tends to jam; the inventor acknowledged this flaw and tried to correct it with an anti-jamming device. Until recently the machine was thought lost, but in 2010 the author found one at the Arithmeum in Bonn (Germany), already described by Hans-Joachim Vollrath (Vollrath 1997), but not yet ascribed to Gonnella. The ascription is still not assured for some small particulars are different from Gonnella's original drawings (Hénin 2012), but, although not original, it was certainly based on Gonnella's design. The second calculator described in the booklet is a keyboard single-digit adder (column adder) (Hénin 2010a), akin to Parmelee's prototype (Turck 1972, pp. 16-29), but with a different mechanism: nine keys moved, via as many toothed sectors, a numbered helical drum to add digits up to 599. The small and simple machine is now in the collection of *Museo Galileo*[21] in Florence, still in good conditions. Both Gonnella's digital calculators were exhibited several times in the second half of 19th century.

[20] Exhibition of the Works of Industry of all Nations, 1851: reports by the juries on the subjects in the thirty classes into which the exhibition was divided. Clowes, London, 1852.

[21] Previously known as 'Museo di storia della Scienza'.

5.3 Niccola Guinigi and His One-of-a-Kind Calculator

Count Niccola Guinigi (1818-1900) was born in Lucca (Tuscany), offspring of a powerful and wealthy merchant family who ruled the city since 15th century[22]. Few scattered information can be gathered about Niccola's life and almost nothing can be found about his possible technical and scientific interests.[23] On January 16, 1659 Count Guinigi submitted a calculating machine of his invention to the *Accademia Toscana di Arti e Manifatture* in Florence, the calculator was scrutinized by two *Accademia*'s referees[24] who found it "*particularly commendable for its simplicity and robustness*", but suggested improvements to overcome some flaws (Corridi 1859). After that date no other record of Guinigi's contrivance can be found, but a letter by Tito Gonnella (Gonnella 1859) to the *Accademia*, claiming priority over Guinigi. In spring 2010 the author serendipitously came across the forgotten Guinigi's machine in an antique shop in Milan (Hénin 2012), with autograph notes by Guinigi, and was granted the opportunity to examine it. The calculator is a dial adder of unique architecture; in fact it has a single large dial to be turned by hand to enter each digit of the addenda and the decimal order of magnitude is selected by eight mutually exclusive levers. As in the case of Gonnella's dial adder, Guinigi's suffers from ripple-carry jamming and the inventor tried to solve the problem with two levers to prevent it. No other dial adder with such a feature has been reported in the literature. The making of Guinigi's adder is all in wood with small metal details (springs and screws) but, after 150 years, still in good and almost working conditions. Guinigi's machine has recently been acquired by Arithmeum in Bonn (Germany).

Fig. 8. Niccola Guinigi's dial adder (1859) (courtesy of Arithmeum, Rheinische Friedrich-Wilhelms-Universität, Bonn)

[22] Niccola's most famous ancestor was Paolo Guinigi (1372-1432).
[23] Neither printed works nor manuscripts have been published under his name.
[24] The mathematicians Giovanni Novi and Antonio Ferrucci.

5.4 Other 19th Century Italian Inventors

Several other inventions are briefly mentioned in contemporary literature: In 1824 **Benedetto Isidoro Brun** submitted to the *Istituto Lombardo di Scienze e Lettere,* the invention of a keyboard calculator**,** in 1832 abbot **G. P. Genevois** submitted a simpler one for the four operations and in 1833 **Carlo Mezzanotte** tried with his portable calculating machine. **Giuseppe Mozzoni** was another Lombard inventor who was awarded a medal in 1847 for a small mechanical adder. The Turinese **Luigi Palagi Palmarini** in 1829 was granted a ten years patent (*privilegio*) by the government of the Kingdom of Sardinia for a computing machine. A watch maker from Mondovì (near Turin), **Opprandino Musina**, exhibited a pocket dial adder at the *Exposition Universelle* of 1867 in Paris, a short description has been published by the American Committee officially appointed to report on the exhibition (Barnard 1869). Musina's was another prototype of small dial-adders of the pascaline kind. For all the a.m. names, unfortunately, coeval printed sources are scant, inaccurate and often unreliable; a long effort to search several scattered local archives for more information would be required.

A different kind of computing machinery was in the meantime appearing, for statistical data processing: the 'punched card' technology invented by Herman Hollerith (Austrian 1982). Hollerith's invention of 1889 was well known in Italy thanks to his friendship with the Italian statistician Luigi Bodio and an attempt to use punched card tabulators was proposed by Bodio for the Italian census of 1891 (Hénin 2010b). Unluckily, scarce resources were available and the census was postponed. Ten years before and independently, one of Bodio's collaborators, **Luigi Perozzo** (1856-1916), invented a mechanical tabulator to be used for 1881 census, but finished one year later. It was a desk-like device with 144 mechanical counters, each identified by a label for a particular item to be counted (e.g. age, nationality, job, marital status …) . The census clerk read the data collected and pushed the proper counter, increasing it by one unit. When a group of data were finished the clerk put a large paper form on the desk and pressed it against the counters, previously inked, and the results were printed on the form. Perozzo's tabulator never went into use, but his idea was taken by the Frenchman Lucien March few years later (1899) and was widely used both in France and Italy. In a letter of 1904 to *Ministère du Commerce de l'Industrie*, March acknowledged Perozzo's priority (Pietra 1934).

6 20th Century: The Rise of Industrial Inventions

Almost all the Italian inventors, but Torchi and (perhaps) Galileo, we have seen so far were scholars or noblemen who attempted the construction of calculating machinery more to satisfy their curiosity and to challenge their ingenuity than to obtain a reward or income, they were often not in the need of. Moreover in Italy a possible market for their instruments was certainly negligible. It was only after 1850 — even more after the political unification in 1861 — that Industrial Revolution reached Italy, particularly in the Northern regions, with the appearance of railway networks, electric power plants, big industries and technical high-schools. Thus, inventions no longer were the activity of secluded amateurs, but became a professional undertaking strictly connected to industrial production, marketing and entrepreneurship.

6.1 Fossa-Mancini's Adder and Taeggi-Piscicelli's Electromechanical Calculator

Count Carlo Fossa-Mancini (1854-1931) was an engineer, physicist and political philosopher. In 1893 he invented an innovative dial adder (Soresini 1971, p. 60; Celli 2006; Jacob 1911, pp. 17-19; Martin 1925, p. 129), where the dials were no longer on the same plane on different vertical axles, like the *pascaline*, but vertically arranged on one common horizontal shaft. Digits were set by turning the dial borders, thanks to pins on the dial edge. This architecture allowed quick setting of operands, prevented carry jamming, and permitted a more compact structure of the calculator. In 1899 Fossa-Mancini applied for a British patent, granted the same year [Pat. N. 189904489, July 1, 1899]. A small scale production (perhaps a few hundred) was made by the French company Japy Fréres & Cie — a producer of clocks, farm and house appliances — beginning of 20th century. The calculator do not seem to have obtained a sensible sale success and production soon came to an end, but decades later Fossa-Mancini's design was resumed by several producers in Italy (Brevetti Lanza's *Addipresto*), Germany (*Addi, Comet, Argenta*), USA (*American Adder, Todd-Star, Little Geant*), Japan (*Addimat*) and Hong Kong (*Swift Handy*), until the mid-1950s (Celli 2006). All those 'clones' were cheap, reliable and easy to use. Of the original, Japy-made calculators only a few still remain: one of them is exhibited in the museum of Castelplanio, Fossa-Mancini's home town; another one has been recently offered on an internet auction site (March 2012).

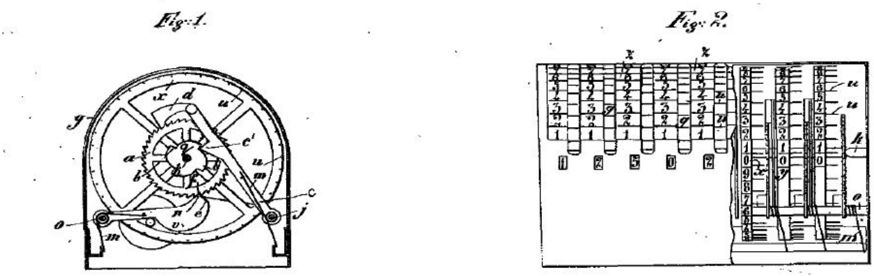


Fig. 9. Fossa-Mancini's calculator (1899) (UK Patent n. 4489, 1899)

Another forgotten inventor was the Neapolitan Count Roberto Taeggi-Piscicelli (Celli 2010; Martin 1925, p. 263), nobleman and engineer who was also political activist and economist. In 1903 Taeggi-Piscicelli invented a cash register patented in USA (US. Pat. N. 872845, granted 1907) the rights were later sold to the American National Cash Register Co. In 1911 the inventor submitted another patent in Germany, Austria and United States for an electrical motor-driven calculator (US. Pat. N. 1416974) one of the earliest electromechanical machine, perhaps the second after the *Autarith* created in 1902 by the Czechoslovakian Alexander Rechnitzer. Taeggi's machine, with the brand name *Sanders*, was manufactured by the French Société Industrielle des Téléphones.

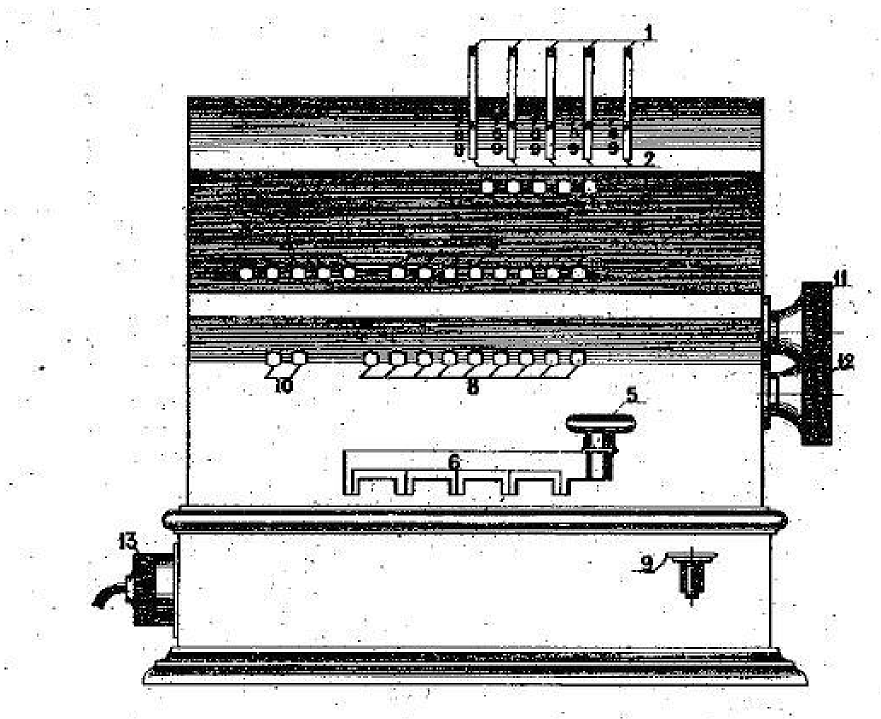


Fig. 10. Taeggi-Piscicelli's electrical calculator (1912) (British Patent n. 10148, 1912)

6.2 The Rise of the Italian Calculator Industry

In spite of Italian inventiveness and of growing national market, no calculator industry was created before 1930. It was only with the autarky enforced by the fascist regime — protectionism due to the economic crisis of 1929 and to the embargo imposed by the League of Nations as a consequence of the Italian-Abyssinian war of 1935 — that an effort to create national industries to minimize import from abroad was carried on. As a matter of facts, both causes were not so severe — the embargo lasted only a few months and was strictly observed by Great Britain only — but fascist propaganda exploited autarky as a stimulus for national pride and popular consensus. Thus, after 1930 a bunch of Italian manufacturer[25] came up (Torchio 1999), producing several models of full-keyboard (Comptometer's style) and reduced-keyboard adders (printing and non-printing) and also four-operations machines (Brunsviga's style). Their market was almost restricted to Italy, with limited export, and some of the models were manufactured on foreign patent licenses. The outbreak of WWII possibly caused some of those companies to switch the production to light weapons, ammunitions and other warfare technology. After the end of WWII and the

[25] Addicalco, Invicta, Italcalcolo, Serio, Inzadi, Ducati, F.A.C.S.A., M.U.S.A., Pozzi, PSIC, SIMAS-Stiatti (Celli, 2006).

following reconstruction, giving rise to the 1950s – 1960s economic boom, other producers entered the market with more advanced technology and electromechanical models[26]. Cash registers were also produced in Italy since 1925 by SIR (Società Italiana Registratori) and since 1936 by RIV (Società Anonima Officine di Villar Perosa), designed and patented by Biagio Beria (Celli 2010).

Only one Italian company was able to reach a world-wide market: **Olivetti.** Founded by Camillo Olivetti in 1908 for the production of typewriters, with the headquarters in Ivrea, near Turin, Olivetti was one of the most successful Italian manufacturer, renown not only for its products but also, and mainly, for its forefront management style and top level industrial design — several Olivetti's products are shown in international art museums and exhibitions — particularly under the guide of Camillo's son Adriano in the post-WWII era (Ochetto 2009; Semplici 2001). In 1935 Adriano Olivetti realized that, although successful in the typewriter market, his company was missing the opportunity of other office appliances, like desk top calculators. The first mechanical calculator was produced in 1940 (the MC 4 *Summa*) designed by Riccardo Levi with the help of a young uneducated, but ingenious worker: **Natale Capellaro**. After WWII Olivetti brought out a series of innovative electromechanical models (e.g. Divisumma, Logos, Elettrosumma,) and some complex bookkeeping machines (Tetractys), almost one new model every two years, until 1965, thanks to Capellaro's great ingenuity. Indeed Capellaro's contribution was highly prized in Olivetti ensuring him an unbelievable career for a blue collar with just an elementary education. Capellaro joined Olivetti in 1916 as apprentice worker, in 1943 was appointed Head of Project Office, then Technical Director in 1960, and was awarded the degree *honoris causa* in engineering at the University of Bari in 1962 (Silmo 2008; Salvetti 2002). Capellaro is author of more than thirty Olivetti's patents; thanks to him, Olivetti's machines had two main advantages over competitors: they were much more reliable and fast, although still electro-mechanical, and their manufacture was cheaper, permitting both a lower selling price and a higher profit (production cost of Divisumma 24 was one tenth of selling price). Due to these advantages Olivetti in 1956 exported more than 30% of its production and near 6 million dollars worth of calculating machines were sold in US the same year (Cortada 1993, pp. 251-255).

7 Other Inventions of the 20th Century

7.1 Annibale Pastore's Logical Machine (1906)

After the publication of George Boole's '*Mathematical Analysis of Logic*' in 1847, the discipline no longer pertained to philosophy but became the job of mathematicians; with the new symbolic logic notation, syllogisms could be solved by mathematical procedures, and algorithms can be, in principle, carried out automatically by

[26] Aldo Bona, ELMIS, ELMETECNIL, SAMAS, Steiner, Serio S.p.A., Luini, FACSA, I.M.C.A., Industria Calcolatrici Scriventi, Peghetti-Corsini (Celli, 2006).

mechanical devices. Already in the second half of 19th Century the British economist Stanley Jevons built a mechanical 'logical piano' to solve simple problems of class logic, followed by the American Allan Marquand who, in 1885, designed another logical machine, this time electromechanical (relay based), anticipating Claude E. Shannon's studies of fifty years (Gardner 1958; Buck 1999, Maas 2005, pp. 96-150). The Italian philosopher Annibale Pastore (1868 – 1956), later professor of theoretical philosophy at the University of Turin, tried another original approach to the mechanization of logic and published his design in a booklet of 1906 (Pastore 1906). Pastore's machine was a contrivance built with pulleys, differential gears and pendulums, arranged in three groups (for subjects, predicates and middle terms of a syllogism) connected by belts. If the syllogism was valid (true) the wheels rotate smoothly, if invalid (false) the wheels locked (Gardner 1958, pp. 114-116). Of course, all those machines were not intended for practical use or experimental demonstration, they were too clumsy, slow and could manage just a limited amount of propositions, but their educational value is still interesting.

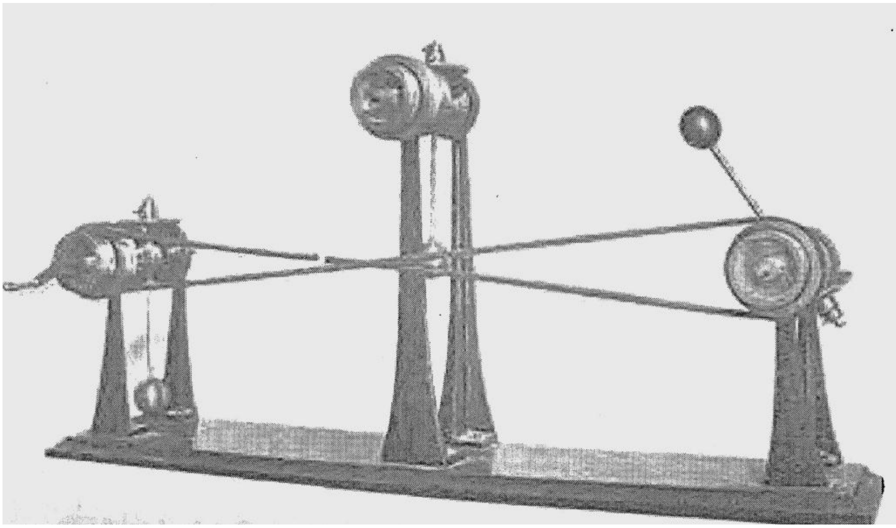


Fig. 11. Pastore's syllogistic calculator (Pastore 1906)

7.2 Mechanical Analog Calculators

In the first half of 20th century the need of scientific and technical calculation abruptly increased, due to the development of nation-wide electric power networks, aviation, electronics, chemistry and nuclear physics. As no electronic digital computer was available yet, mathematical problems could be solved only with desktop four-operation calculators or with analog computers like slide rules, planimeters and integrators. The mathematician **Ernesto Pascal** (1865-1940), professor of calculus at the University of Pavia until 1907, then at the University of Naples, devoted himself to the design of analog devices called ‘intergraphs’ . Differently from planimeters, intergraphs — invented by the Polish Bruno Abakanowicz (1880) — do not calculate the definite integral of a curve, but draw the integral function of a given curve and were used for the graphical-analytical solution of particular types of differential equations (e.g. Riccati’s equation). Pascal’s improved models, built by Roberto Marcolongo, were highly prized and reproduced for the use in mathematics departments of several universities. Other two Italian mathematicians**, Lorenzo Poggi** (1905 - 1978) from the University of Pisa, and **Lamberto Cesari** (1910 - 1990), from *Istituto Nazionale per le Applicazioni del Calcolo* of Rome (INAC – CNR), tried to build mechanical calculators for the solution of systems of linear equations. Poggi’s, built with spinning-wheel ribs,[27] was just a crude prototype, although described in print (Poggi 1930). Cesari’s design was patented. In 1939 **Francesco Vercelli** (1883 - 1952) published the description of a mechanical **‘**periodic analyzer’ he invented (Vercelli 1939) to calculate periodicity of meteorological phenomena. Vercelli’s design was built by the optic manufacture Salmoiraghi S.A. in Milan. It is now in the Museum of Physics of Turin University.

7.3 The FERMIAC (1946)

Between winter 1946 and summer 1947 ENIAC was dismantled and moved from Philadelphia’s Moore School, PA, to Aberdeen Proving Ground, MD. (Burks 1981) Thus the first and sole high-speed electronic calculator was unavailable just when nuclear physicists had greatest need of it to keep the advantage they had in atom bomb design. The design required the modeling of neutrons collisions and scatterings, task that was carried out with the statistical ‘Monte Carlo method’ invented by John von Neumann and Stanislaw Ulam. This method, in turn, required repeated random sampling and subsequent calculation of particles trajectory, a rather tiresome and annoying operation, better performed by computers. In the absence of ENIAC, the Italian born physicist **Enrico Fermi** (1901-1954), leading scientist of the Manhattan Project, came out with a simple gadget to help (Metropolis 1987). With his colleague Percy King, Fermi designed and built this simple mechanism to draw random neutron paths in a lattice of fissile atoms, which was dubbed FERMIAC in assonance with first computers acronyms, like ENIAC, EDVAC, EDSAC. Although not properly a computing device, FERMIAC can be seen as the last mechanical contrivance in the just born electronic era.

8 Conclusions

After the flourishing of Italian mathematics and instrument making in the Renaissance, due to the early rediscovery and translation of ancient Greek, Hellenistic and Roman documents, and to the patronage granted to artisans and scholars by enlightened rulers, Italian science and technology underwent a long period of decline, the

---

[27] Actually, ribs of the tool used for holding skeins of yarn under tension to unravel them and to wrap yarn into balls.

causes being too many and too intricate to be discussed here. Inventors and inventions were no longer prized by political and economic power until the late coming of industrial revolution (late 19th century). As a matter of facts, the static and stratified society prevented the social promotion of gifted individuals, more easily obtainable in Great Britain, France and German countries. In fact, most of the mentioned inventors, as we have seen, were noblemen or held an academic position, and did not depend on their mechanical ingenuity for their bread and butter (perhaps with few exceptions, like Galileo and Torchi), thus in Italy inventor-entrepreneurs (Thomas Edison's style) were always poorly represented (with the exception of Guglielmo Marconi and few others). After 1900 Italian industry grew rapidly, but the time of independent inventors was over, replaced by companies' R&D departments.

To conclude, certainly not all Italian inventors of computing instruments have been mentioned above and I apologize for all the names and inventions I have unintentionally forgotten; possibly twice as many are still to be discovered browsing through dusty documents, patents and reports scattered in the host of archives of universities, scientific societies and industrial companies. This paper is thus not meant to be a comprehensive and ultimate treatise on the topic, but rather a starting point to stimulate scholars to undertake a thoroughly study of the Italian history of technology, still a quite neglected subject in our country, putting together diverse competences of historians, scientists, engineers, sociologists and economists with a multidisciplinary approach.

References

Anastasio (1994) Anastasio, P.: Facini, Santo Bernardo. Dizionario Biografico Degli Italiani 44, Treccani (1994)

Andretta (2009) Andretta, E.: Bartolomeo Eustachi, il compasso e la cartografia del corpo umano. Quaderni Storici 44(1), 130 (2009)

Aquilecchia (1957) Aquilecchia, G. (ed.): G. BRUNO, Due dialoghi sconosciuti e due dialoghi noti. Edizioni di Storia e Letteratura, Rome (1957)

Austrian (1982) Austrian, G.D.: Herman Hollerith. Forgotten Giant of Information Processing. Columbia University Press, New York (1982)

Barnard (1869) Barnard, F.A.P.: Report on Machinery and Processes of the Industrial Arts and Apparatus of the Exact Sciences, pp. 637–638. D. van Nostrand, New York (1869)

Bedini (1995) Bedini, S.A.: The fate of the Medici-Lorraine Scientific Instruments. Journal of the History of Collections 7(2), 159–170 (1995)

Beniger (1986) Beniger, J.R.: The Control Revolution. Harvard University Press, Cambridge (1986)

Berzolari (1939) Berzolari, L.: Necrologio di Ernesto Pascal. Rend. Ist. Lombardo 3(12), 162–170 (1939-1940)

Bianca (1982) Bianca, C.: Federico Commandino. In: Dizionario Biografico Degli Italiani, vol. 27, Ist. Dell'Enciclopedia Italiana, Roma (1982)

Boffito (1931) Boffito, G.: Il primo compasso proporzionale costruito da Fabrizio Mordente e la operatio cilindri di Paolo dell'Abbaco. Il Facsimile (6) (1931)

Bonfanti (1988) Bonfanti, C.: Sulla macchina aritmetica di Poleni. In: Soppelsa, M.L. (ed.) Giovanni Poleni Idraulico, Matematico, Architetto, Filologo (1683-1761), Padua, pp. 163–174 (1988)

Borchi (1997) Borchi, E., Macii, R.: Tito Gonnella, inventore nella Firenze dell'Ottocento. In: Borchi, E., Macii, R., Vetrano, F. (eds.) Strumenti di Fisica e Cultura Scientifica Nell'Ottocento in Italia, Florence, pp. 103–110 (1997)

Bromley (1990) Bromley, A.G.: Analog Computing Devices. In: Aspray, W. (ed.) Computing before Computers, pp. 159–199. Iowa State University Press (1990)

Buck (1999) Buck, G.H., Hunka, S.M.: W. Stanley Jevons, Allan Marquand and the Origins of Digital Computing. IEEE Annals of the History of Computing 21(4), 21–27 (1999)

Burks (1981) Burks, A.W., Burks, A.R.: The ENIAC: The First General-Purpose Electronic Computer. Annals of the History of Computing 3(4), 310–389 (1981)

Camerota (2000) Camerota, F.: Il compasso di Fabrizio Mordente: per la storia del compasso di proporzione, L. S. Olschki, Florence (2000)

Camerota (2003) Camerota, F.: Two new attributions: a refractive dial of Guidobaldo del Monte and the Roverino Compass of Fabrizio Mordente. Nuncius XVIII, 1 (2003)

Campbell-Kelly (1989) Campbell-Kelly, M.: ICL. A Business and Technical History. Clarendon Press (1989)

Campbell-Kelly (2003) Campbell-Kelly, M., Croarken, M., Flood, R., Robson, E.: The History of Mathematical Tables. From Sumer to Spreadsheets. Oxford University Press, Oxford (2003)

Care Care, C.: Illustrating the History of the Planimeter, http://empublic.dcs.warwick.ac.uk/projects/planimeterCare2004/Docs/report.pdf (accessed March 14, 2012)

Carse (1982) Carse, G.A., Urquhart, J.: Planimeters. In: Horsburg, E.M. (ed.) Handbook of the Napier Tercentenary Celebrations, or Modern Instruments and Methods of Calculation, Bell, London (1914); Charles Babbage Institute Reprint Series for the History of Computing. Tomash Publishers, Los Angeles, pp. 190–206 (reprinted 1982)

Celli (2006) Celli, A.: Convegno sulla sommatrice automatica, Castelplanio (August 22, 2006), http://www.comune.castelplanio.an.it/Engine/RAServePG.aspx/P/2775160100/M/2500160101 (accessed February 6, 2012)

Celli (2010) Celli, A.: Appunti per una storia dei Registratori di Cassa italiani. L'ufficio d'Epoca (48) (2010)

Cipolla (1987) Cipolla, M.: La moneta a Firenze nel cinquecento. Il Mulino, Bologna (1987)

Corridi (1859) Corridi, F.: Adunanza ordinaria del dì 16 gennaio 1859. Atti verbali della I. e R. Accademia Toscana d'Arti e Manifatture vii(7), 170–178 (1859)

Cortada (1993) Cortada, J.: Before the Computer. Princeton University Press (1993)

De Fouchy (1763) De Fouchy, J.P.G.: Eloge de Jean POLENI, Marquis du St. Empire, (né 1683 mort 1761). Histoire et Mémoires de l'Académie Royale des Sciences [part. Histoire] 1763, p. 151 (1763)

Drake (1988) Drake, S.: Galileo. Una biografia scientifica. Il Mulino, Bologna (1988)

Drake (1999) Drake, S.: Essays on Galileo and the History and Philosphy of Science, vol. 3. University of Toronto Press, Toronto (1999)

Favaro (1883) Favaro, A.: Galileo Galilei e lo Studio di Padova. Le Monnier, Florence, vol. I (1883)

Favaro (1907) Favaro, A.: Per la storia del compassi di proporzione. Atti del reale Istituto Vento di Scienze e Lettere 67, 723–739 (1907-1908)

Fellman (1983) Fellmann, R.: Römische Rechentafeln aus Bronze. Antike Welt 14, 36–40 (1983)

Galilei (1606) Galilei, G.: Le operazioni del compasso geometrico et militare. Padova (1606), http://fermi.imss.fi.it/rd/bdv?/bdviewer/bid=300783# (accessed January 14, 2012)

Gamba (1988) Gamba, E.: Matematici urbinati del Cinque-Seicento. In: Gamba, E., Montebelli, V. (eds.) Le Scienze in Urbino nel tardo Rinascimento, Quattroventi, Urbino (1988)

Gardner (1958) Gardner, M.: Logic Machines and Diagrams. University of Chicago Press (1958)

Gennari (1839) Gennari, G.: Elogio del March. Gio. Poleni, Padua (1839)

Geymonat (1970) Geymonat, L.: Galileo Galilei. In: Geymonat, L. (ed.) Storia del Pensiero Filosofico e Scientifico, Garzanti, vol. II (1970)

Gonnella (1825) Gonnella, T.: Teoria e descrizione d'una macchina colla quale si quadrano le superfici piane. Antologia, Florence, V, XVII(52) (1825)

Gonnella (1859) Gonnella, T.: Descrizione di due macchine aritmetiche per l'addizione immaginate e costrutte da Tito Gonnella, Florence, Tip. Calasanziana (1859)

Hénin (2009) Hénin, S.: Poleni. Museo della Scienza e della Tecnologia, Milan, http://www.museoscienza.org/approfondimenti/documenti/macchina_poleni/default.asp (accessed February 4, 2012)

Hénin (2010a) Hénin, S.: Two Early Italian Key-driven Calculators. IEEE Annals of the History of Computing (1), 34–43 (2010)

Hénin (2010b) Hénin, S.: Buon compleanno Mr. Hollerith. Mondo Digitale (34), 47–57 (2010)

Hénin (2012) Hénin, S., Temporelli, M.: An Original Italian Dial Adder Rediscovered. IEEE Annals of the History of Computing 34(2) (2012)

Henrici (1894) Henrici, O.: Report on planimeters. Report of the Sixty-forth Meeting of the British Association for the Advancement of Science, pp. 496–523 (1894)

Horsburg (1914) Horsburg, E.M. (ed.): Napier Tercentenary Celebration. Handbook of the Exhibition of Napier relics and of Books, Instruments, and Devices for Facilitating Calculation. Royal Society of Edinburgh (1914)

Ifrah (2000) Ifrah, G.: The Universal History of Numbers. John Wiley & Sons (2000)

Jacob (1911) Jacob, L.: Le Calcul Mécanique. O. Doin et Fils, Paris (1911)

La Fama (1840) Anon: Macchina aritmetica inventata ed eseguita da Luigi Torchi milanese. In: Fama, L.: Rassegna di Scienze, Lettere, Arti, Industria e Teatri, vol. 10 Gabinetto di scienze, arti ed industria, Milan (1836)

Leupold (1727) Leupold, J.: Theatrum Arithmetico-Geometricum. Das ist Schauplatz der Rechen und Mess-Kunst, Leipzig (1727)

Maas (2005) Maas, H.: William Stanley Jevons and the Making of Modern Economics. Cambridge University Press (2005)

Marguin (1994) Marguin, J.: Histoire des intruments et machines à calculer. Hermann, Paris (1994)

Martin (1989) Martin, C.: Notes sur l'abacus d'Aoste. Memorie dell'Accademia Italiana di Studi Filatelici e Numismatici IV(fasc.1), 59–62 (1989)

Martin (1925) Martin, E.: Die Rechenmaschinen und ihre Entwicklungsgeschichte. J. Pappenheim, Berlin (1925)

Menninger (1992) Menninger, K.: Number Words and Number Symbols. A Cultural History of Numbers. Dover Pubbl., New York (1992)

Merton (1973) Merton, R.K.: The Sociology of Science: Theoretical and Empirical Investigations. University of Chicago Press, Chicago (1973)

Metropolis (1987) Metropolis, N.: The Beginning of the Monte Carlo Method. Los Alamos Science, Special Issue, 125–130 (1987)

Miniati (2009) Miniati, M. (ed.): Firenze Scienza. Le collezioni, i luoghi, i personaggi dell'Ottocento, Firenze (2009)

Morland (1673) Morland, S.: The Description and Use of Two Arithmetick Instruments, Together with a Short Treatise, Explaining and Demonstrating the Ordinary Operations of Arithmetick, London (1673)

Nagler (1960) Nagler, J.: Beschreibung der Rechenmaschine des Antonius Braun. In: Blätter für Technikgeschichte, Vienna, pp. 81–87 (1960)

Needham (1959) Needham, J., Wang, L.: Science and Civilization in China: Mathematics and the Sciences of the Heavens and the Earth. Cambridge University Press, Cambridge (1959)

Ochetto (2009) Ochetto, V.: Adriano Olivetti, Marsilio (2009)

Pascal (1914) Pascal, E.: I miei intergrafi per equazioni differenziali. Libreria Scientifica e Industriale, Napoli (1914)

Pastore (1906) Pastore, A.: Logica formale dedotta dalla considerazione di modelli meccanici: con 17 figure ed 8 tavole fuori testo, Turin, Fratelli Bocca (1906)

Pietra (1934) Pietra, G.: La prima classificatrice è stata ideata da un italiano. Barometro Economico Italiano, 463–467 (August 10, 1934)

Pignoria (1613) Pignoria, L.: De Servis, Amsterdam (1613)

Poleni (1709) Poleni, G.: Miscellanea. Hoc est 1. Dissertatio de barometris, & thermometris, 2. Machinae aritmeticae, ejusque usus descriptio, 3. De sectionibus conicis parallelorum in horologiis solaribus tractatus. Venice, Aloysium Pavinum (1709)

Pullan (1968) Pullan, J.M.: The History of the Abacus. Hutchinson (1968)

Ratcliff (2007) Ratcliff, J.R.: Samuel Morland and his calculating machines c.1666: the early career of a courtier–inventor in Restoration London. Brit. J. Hist. Science 40(2), 159–179 (2007)

Righini (1980) Righini, G., Righini, M.L.: Nota su un calcolatore logaritmico di Bernardo Facini. Annali dell'Istituto e Museo di Storia Della Scienza di Firenze 5(1), 61–74 (1980)

Roegel (2008) Roegel, D.: An Early Key-Driven Adding Machine. IEEE Annals of the History of Computing 30(1), 59–65 (2008)

Salvetti (2002) Salvetti, A., Pacchioli, E. (eds.): Le macchine sapienti di Natale Capellaro. Associazione Archivo Storico Olivetti, Ivrea (2002)

Schärlig (2001) Schärlig, A.: Compter avec des cailloux. Presses Polytechniques et Universitaires Romandes, Lausanne (2001)

Schärlig (2003) Schärlig, A.: Compter avec des jetons. Presses Polytechniques Universitaires Romandes (2003)

Schärlig (2006) Schärlig, A.: Compter avec des doigts. Presses Polytechniques et Universitaires Romandes, Lausanne (2006)

Semplici (2001) Semplici, S.: Un'azienda e un'utopia. Adriano Olivetti 1945-1960, Il Mulino (2001)

Silmo (2008) Silmo, G.: M.D.C., Macchine da calcolo meccaniche, Olivetti e non solo; Natale Capellaro - Il genio della meccanica. Tecnologicamente Storie, Ivrea (2008)

Smith (1925) Smith, D.E.: History of Mathematics. Special Topics on Elementary Mathematics, vol. II. Ginn and Company (1925)

Soppelsa (1963) Soppelsa, M.L. (ed.): Giovanni Poleni (1683-1761), nel bicentenario della morte. Atti e memorie dell'Accademia patavina di scienze lettere ed arti, vol. 74(suppl.) (1963)

Soppelsa (1983) Soppelsa, M.L.: Giovanni Poleni idraulico, matematico, architetto, filologo (1683-1761): atti della Giornata di studi, Padua, Erredici (1988)

Soresini (1971) Soresini, F.: La storia del calcolo automatico, Roma, Confindustria, vol. I (1971)

Soresini (1991) Soresini, F.: La macchina aritmetica di Giovanni Poleni e la sua ricostruzione. In: Atti Precongressuali del Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e dell'Informatica, Siena, Milan, AICA (1991)

Tancon (2005) Tancon, J.: Lo scienziato Tito Livio Burattini al servizio del Re di Polonia. Università di Trento (2005)

Targosz (1982) Targosz, K.: La cour çavant de Marie Louise de Gonzague, Krakow (1982)

Torchio (1999) Torchio, C.: L'ufficio d'Epoca, 14, 17, 20 (1999)

Turck (1972) Turck, J.A.V.: Origin of Modern Calculating Machines, New York (1972)

Vergara-Caffarelli (1992) Vergara Caffarelli, R.: Il Compasso Geometrico e Militare di Galilei, Testi, annotazioni e disputa negli scritti di. In: Galilei, G., Bernegger, M., Capra, B. (eds): Documenti e contributi di storia della scienza, ETS, Pisa (1992)

Vergara-Caffarelli (2006) Vergara-Caffarelli, R.: Il compasso geometrico militare. 2 – Studio del compasso galileiano di Pisa (2006), `http://www.illaboratoriodigalileogalilei.it/galileo/2006%20-%20Studio%20del%20compasso%20di%20Pisa.PDF` (accessed January 14, 2012)

Vollrath (1997) Vollrath, H.-J.: Eine alte italienische Addiermaschine im Holzkasten. Historische Bürowelt 49, 25–27 (1997)

Welser (1594) Welser, M.: Antiqua quae Augustae Vindelicarum extant Monumenta, Venice (1594)

Vercelli (1939) Vercelli, F.: Analizzatore meccanico delle curve oscillanti. Pontificia Academia Scientiarum, Commentationes 3(19), 659–692 (1939)

Williams (1990) Williams, M.R.: Early Calculation. In: Aspray, W., et al. (eds.) Computing Before Computers, Iowa State University Press (1990)

Williams (1997) Williams, M.R.: History of Computing Technology. IEEE Computer Society, Los Alamitos (1997)

Computing for the Masses? Constructing a British Culture of Computing in the Home

Tilly Blyth

Keeper of Technology and Engineering, Science Museum, London
Tilly.Blyth@ScienceMuseum.org.uk

Abstract. The creation of the personal computer during the late 1970s and early 1980s is heralded as a time that people were liberated by computers as tools for everyone. The proliferation of affordable and relatively powerful computers changed the landscape of computing across the globe. This chapter looks at the introduction of one machine, the BBC Microcomputer, and its influence on the culture of computing in Britain. It has been celebrated as a computer that transformed the educational landscape and brought the power of these tools to a new generation of users. The chapter shows how the machine was constructed within a broader ambition for computer literacy within Britain, and discusses the role of the BBC team in creating the meaning and values of the machine in the home. It illustrates the interplay between a broadcaster, government desire for a high-tech industry and perceived consumer needs. Drawing on the social construction of technology by a variety of actors (Woolgar, 1991) the chapter suggests that enthusiasm for the BBC Microcomputer came not only from the creation of a concept of utility for home machines, but in its role as a technology that embodied the future and symbolised the social capital of the home.

Recent nostalgia for the BBC Microcomputer and Computer Literacy Project has celebrated this moment in the 1980s as a time when the government seeded a new passion for computing. The chapter suggests that similar projects today, which aim to create an interest in programming, should facilitate a social need for empowerment and interaction in the home, rather than focus purely on the technical capabilities of the machine or push a concept of the perceived utility of computers in education.

Keywords: Computer Literacy Project, BBC television, BBC computer, Acorn.

1 We Are Afraid of the Future

"Perhaps the survival of a nation depends upon its people finding meaningful lives. The questions shout. What is shocking is that the government has been totally unaware of the effects that this technology is going to create. The silence is terrifying. It's time we talked about the future." (Goldwyn, 1978)

When the BBC's *Horizon* programme: *Now the Chips are Down*, was broadcast in 1978, these final words shocked the nation. Television screens across Britain turned black as the programme's director, Edward Goldwyn, stressed the impact the microprocessor and automation would have on employment.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 231–242, 2012.

The programme had taken over two and a half years to research and produce. It showed how microchips were made and involved some difficult filming, especially in the clean area of a chip production plant in Silicon Valley. Initially the BBC crew were denied access to this state-of-the-art facility, but the decision was overruled from the top when Robert Noyce, one of the original inventors of the integrated circuit, gave the BBC crew unique access to the clean room at his microchip company Intel.

Projecting dystopian views of the impact of technology on society, the television programme showed robots working without any human intervention, suggested there would be far fewer jobs in manufacturing as we moved to automation, and ended with a completely blank screen. Goldwyn had agonised for days about how the programme should end, concerned that nobody within UK's Callaghan government seemed to be taking the issue seriously and that the weight of the issue was all on his shoulders[1]. In the end he decided that a blank screen would bring home the gravity of the issues involved and a decision was made within the BBC that the transmission of the programme should be followed by a debate about what could be done to prevent a terrible future of mass unemployment.

The impact of the programme was enormous: that night the office of the Prime Minister, James Callahan, asked the BBC for a copy; questions about the elimination of jobs by microprocessors were raised in parliament[2]; a trade union made 78 copies of it and showed them to its members[3]. The programme's influence even began to spread overseas as the Organisation for Economic Cooperation and Development (1981) raised it in Paris during international discussions on microelectronics and productivity.

But *Now the Chips are Down* also created a massive impact as it tapped into a bigger culture of concern about the nation's response to its changing economic position and developing technology. There was considerable anxiety that Britain was not doing enough, in terms of industrial support and skills development, to embrace new technology. Britain had been through a period of rising unemployment, high inflation and economic crisis exemplified by the need for an International Monetary Fund loan. It was about to experience the 'winter of discontent' as widespread strikes spread across Britain when the unions reacted strongly to the Labour government's attempt to limit pay rises. The future did not look rosy, and people were looking for a serious response from the government.

Some organisations believed the government had done little to take account of technological change and the skills shortages that produced. The National Economic Development Office (1978) noted that there were specific skill shortages in the microelectronics industry and a need for a coordinated programme to develop

[1] Interview with Edward Goldwyn, Science Museum, 13th March 2006.

[2] Mr. Sherby asked the Secretary of State for Employment what consideration he had given to the effect on employment prospects during the next decade of the development of microprocessors and their use by industry and commerce in ways which will eliminate many jobs. Mr. Golding replied: "*The Employment effects of technological developments which are the subject of present studies are naturally a matter of great concern to me. The Government are fully determined to ensure that the additional wealth which they create will be used to expand employment in other sectors*". Parliamentary Debates (Hansard), 25th May 1978, Vol. 950, c. 680.

[3] Interview with Edward Goldwyn, Science Museum, 13th March 2006.

awareness of the potential uses of microelectronics in the rest of the industry. The Council for Educational Technology (1978) stressed the importance of learning microcomputing in schools. Government departments, such as the Department of Education and Science (DES) and the Department of Trade and Industry (DoI) began to stress the desirability of computer related learning and promote the use of new technology in schools as a panacea for economic recession (Linn, 1991). There was a general conviction that the country must jump on the microcomputer bandwagon, or it would get seriously left behind. With support from government, the BBC, began to think about how it could step into the frame.

2 Constructing Computer Literacy

'The aim was to democratise computing.
We didn't want people to be controlled by it, but to control it.'
David Allen, Project Editor, BBC Computer Literacy Project, 2012[4]

As a result of government concern for the future and the high-profile of the microelectronics revolution, the idea for a Computer Literacy Project began to be developed by the BBC. In the late 1970s the BBC's Continuing Education Department was very strong within the organisation and the Head of Continuing Education Television, Sheila Innes, believed fervently in adult self-improvement and the BBC's role in 'bringing education to the people'. For Innes, the power of television was its ability to educate the largest number of people in the broadest sense.

In early 1980 Innes sent two people from her team, David Allen and Robert Albury, to '*go and see if there is anything in this microelectronics business*'[5]. Their work was backed by the British government's Department of Industry (DoI), the Department of Education and Science (DES) and the Manpower Services Commission (MSC), a quango responsible for establishing the national training needs of the country. The DoI were to help with the consultancy, and the MSC would finance a global fact finding trip through a grant of £10,000 that included America, France, Germany, Holland, Norway, Sweden and Japan. The result of David Allen and Robert Albury's extensive research was a series of three television documentaries, *The Silicon Factor*[6], which looked at the social impact of the micro chip, and a report, *Microelectronics*, distributed in August 1980. The document was widely circulated to policymakers in Britain, and copied to all MPs.

As well as highlighting the need for changes to Britain's work structure and ethic, the *Microelectronics* report suggested there was a strong argument for educating children in computing, vocational training in Further Education, and Adult Education focused on the professions, so that at all levels there were advocates for computer

[4] David Allen at the Beeb 30th Anniversary panel debate held by the Centre for Computing History, at ARM, Cambridge, 25th March 2012.
[5] David Allen, BCS@50 conference, British Computer Society, 14 July 2007.
[6] Complete list of associated programmes in Appendix B.

literacy across society[7]. As a result of the report the BBC's Continuing Education Department decided they should start a computer literacy project.

The Continuing Education Department had recently completed a successful adult literacy scheme, with associated television series, called *On the Move*, featuring the then relatively unknown actor Bob Hoskins. This included training and activities to support the television programmes. The Continuing Education Department decided to base their computer literacy project on a similar model, offering viewers not just television programmes, but courses, supporting books and software, a strong liaison network for teachers and learners.

But during the development phase of the project the same question kept arising: how would viewers actually get their hands dirty programming if they didn't have a machine, with a standard language, to program on? How could the BBC hope to educate people in programming if there were so many different flavours? Eventually, this led to the decision for the BBC to license production of a BBC branded microcomputer.

3 Constructing the Machine

Within the Continuing Education Department at the BBC the team had evidence that television programmes work brilliantly at sparking enthusiasm and teaching them a little bit, but that the most important thing they do is to motivate people to go out build on their interest. Educational television can inspire people to buy books, or take a course in something, but the best way of learning what a computer could actually do was to run a program, adapt and change it and learn to control a computer by writing your own program.

The BBC computer Literacy Project team, led by Executive Producer John Radcliffe, looked at the range of different and incompatible programming languages in the market, and realised there was a need for one consistent language that could support the television programmes and offer a complete and integrated learning system across computing and television. Rather than create their own machine, the Department of Industry (DoI) and the BBC Education team originally tried to persuade the British computing industry to all implement a common language – ABC (Adopted Basic for Computers). All the manufacturers were brought into a room in Cavendish Square, London and were told that if they did this their machines could be used as part of the project. The manufacturers refused to do this unless there was some financial support for the development of the chips. The government was not forthcoming, so the BBC began to look at alternative arrangements.

Originally with the DoI they approached Newbury Laboratories, but when discussions at the end of 1980 had still not resulted in any concrete prototypes, John Coll (an educational adviser who worked at Arundel School and represented Micro Users in Secondary Education) and David Allen drew up a functional description of the machine[8]. This ambitious specification required the machine to be transparent, upwards expandable (so that the machine could work with future hardware) and

[7] D. Allen and R. Albury, 1980, *Microelectronics*, BBC Education, pp.1-64.
[8] John Coll, 'Outline Specification for the BBC Microcomputer.' No date.

downwards compatible (so that it worked with as much existing software as possible). It also had to give users the opportunity to programme at different levels of complexity, so that entry level programmers had as much fun as those with more developed skills. The specification had 'everything but the kitchen sink in it'[9] but just in case all that wasn't enough, it was agreed that the machine needed to be low enough in price for it to be accessible to a large audience.

The BBC team approached seven companies (Acorn, Tangerine, Newbury, Research Machines, Sinclair, Transam and Nascom) to submit bids to build the BBC machine. These British companies were specially selected because of their reputation in the existing industry and fears that an open tender would result in an overwhelming number of bids.

After a series of meetings the BBC gave the leading contenders less than a week to produce a prototype that went some of the way to meeting their specification. The Acorn team, made up in part by Hermann Hauser, Chris Curry, Steve Furber, Sophie Wilson and Chris Turner, worked three days and two nights almost continuously to get a working prototype together for the BBC visit on Friday. They called in Ram Banerjee, 'the fastest gun in the West'[10], to wire wrap (a simple way to prototype) the machine by Wednesday. The team started to debug it, but by Thursday night it was still not working.

As Steve Furber recollected, '*We were all getting very tired, but Hermann was very good at team motivation. It was always his job to go out and buy the kebabs and he would make the tea. He would do all these things just to keep people going. We were all staring at this thing that was still refusing to work and Hermann suggested something like, 'cut the umbilical cord from the prototype to the development system and let it run on its own', which seemed completely daft but we were all out of ideas. So we tried it and the whole thing sprung into life. It was major irritation that Hermann made the final suggestion that caused it to work!*'[11]

By 7am Friday morning the prototype was running and the BBC were coming at 10am. By the time they arrived BASIC was running, with real programs, and by the afternoon some graphics were working. As well as being impressed with the demo, the BBC also liked the 'we can' attitude at Acorn[12] and got on well with the team. The two teams went for a drink in Cambridge after the demonstration. A few days later there was an internal meeting of BBC staff and consultants, chaired by John Radcliffe, where the BBC made a careful comparison of the evidence from the preliminary meetings[13].

The eventual tender was won by Acorn Computers of Cambridge as they had a proven track record building rack-based machines for research laboratories and a reputation for producing reliable single board machines. They had experience of

---

[9] David Allen at the Beeb 30th Anniversary panel debate held by the Centre for Computing History, at ARM, Cambridge, 25th March 2012.
[10] Interview with Sophie Wilson, 16th Oct 2007.
[11] Interview with Steve Furber, 28th January 2008.
[12] David Allen at the Beeb 30th Anniversary panel debate held by the Centre for Computing History, at ARM, Cambridge, 25th March 2012.
[13] Email from John Radcliffe Wednesday 24th September 2008.

manufacturing the Atom in large numbers (around 10,000 were sold) and they were well known to be working on a successor machine, known as the Proton. This infuriated competitors, such as Sinclair who had already produced one of the leading and most successful consumer machines, the ZX80, and had developed, but had not yet launched, the new ZX81 machine. Later this computer would go on to sell a massive 1.5 million units.

Although at the time the decision was controversial, the BBC felt the Sinclair machine was not complex enough, offering less memory and less room for expansion. There were also concerns that Sinclair would not design a new BBC machine but that they would end up with a 'Sinclair machine badged in BBC colours'. Despite being unsuccessful at this stage, Sinclair machines continued to sit in constant competition to the BBC Microcomputer, making the most of the national drive for home computing and selling at far higher volumes than the BBC machine.

4 Constructing Need

Launched on the 11th January 1982, the BBC Computer Literacy Project was foremost a television series imaginatively called 'The Computer Programme'. It was originally planned that the BBC Microcomputer would support these programmes, but production delays by Acorn meant that very few viewers could use their machines alongside the television series.

After the first television series there were further series such as 'Making the Most of the Micro' and 'Micro Live'[14]. The BBC television series played a major role in promoting computing to the masses – showing consumers what it was for, what they could do with a computer and how to program. As Haddon (1988) has illustrated, many of the early 1980s activities of consumers around new home computers consisted of constructing a use for the machine that did very little. Often the act of programming became an end in itself.

Although the BBC television programmes focused around the BBC Microcomputer, their desire to construct the utility of the machines sparked a wider enthusiasm for other home computers and the concept of the computing 'revolution'. In this way the BBC team began to act as 'heterogeneous engineers' (Law, 1987), not only creating the new technology, but defining the social roles, meanings and values of the artefact in society.

The BBC's computing programmes reached audiences between 500,000 and 1.2 million late night on BBC One and, despite the computer's reputation as a schools machine, the television series reached 16% of the adult population through one programme or another[15]. The television programmes brought the Computer Literacy Project directly into the living room, and constructed a vision of a new engaged electronic consumer. Using multiple channels, such as radio, television, books and

---

[14] Speech summarising the key elements of the BBC Computer Literacy Project, R103/101/1 Acorn Computers BBC Master Series Micro Exhibition and Launch.

[15] John Radcliffe, "BBC Micro ignites memories of revolution", BBC website, 21st March 2008, `http://news.bbc.co.uk/1/hi/technology/7307636.stm`

training, the project aimed to 'raise the level of public awareness of what computers are and of how they can contribute to life at home.'[16] So forms of use and social meanings of the machines were not just defined through the television and hardware, but through additional software, courses and beyond.

BBC Education worked with BBC Publications to co-ordinate the production of books and software to accompany the microcomputer. This started with the team commissioning 'Welcome software' on a cassette which came with the BBC Microcomputer, demonstrating colour, sound, graphics, double height as well as single height teletext lettering, screen pages (not scrolling green text on a black background) and the ability of the computer to start and stop the cassette recorder.[17] They then looked at developing further software, as they thought that, just as in the early use of radio, people had become interested in the hardware and then the programmes, so with the microcomputer people began their interest in the hardware but would soon become shift their focus to the content and use of the machine.[18]

Another very significant part of the Computer Literacy Project as a whole was the formal agreement between the BBC and a referral service called Broadcasting Support Service (BSS). Financially supported by the National Extension College, Acorn Computers and the Department of Industry, it acted as a central information point for the Computer Literacy Project: on software; on the computer; on the television programmes and to put viewers in touch with local classes and clubs. The service didn't just deliver to enquirers, but crucially identified other people and organisations that could become part of a bigger network of local sources of help and advice. 800 agencies, including adult centres, universities and computer clubs registered within a month, with the final total being over 1,000. Soon the BSS were answering over 2,000 letters a week (Radcliffe, 1983, 30 - 31).

With computing on mainstream television, and courses such as '30 Hour BASIC' and support networks pushing the idea of computing as the future, the home computing revolution quickly became a phenomenon well beyond the BBC. National newspapers carried stories about the technology companies and the tabloids began to run 'get rich quick' stories about a new generation of computing entrepreneurs. At the same time newsagents were full of a plethora of new computing magazines, such as *Sinclair User*, *Micro User*, *Acorn World*, *Commodore Horizons* and *Amiga World*, which offered programs, often games, which could be copied out and run on the home devices.

Government educational initiatives had also got underway. By 1981 the DoI had set up the Microelectronic Programme (MEP). Headed by Richard Fothergill, it allocated £1.2m in its first year and set a new strategy (Linn, 1991). At the same time as this strategy appeared, the DoI announced a new Micros in Schools programme for secondary schools. This part funded a computer for each school, with the aim to invigorate British industry as well as education. A school taking part in the scheme could

---

[16] 'Acorn Computers BBC Master Series Micro Exhibition and Launch' in File R103/101/1.
[17] Email from David Allen, 3rd May 2012.
[18] Paper tabled by Bob Salkeld at EO's Meeting April 20-22 1982, R99/172/1 Microcomputer Software Policy.

purchase a Research Machine 380Z (which had become a Local Education Authority (LEA) *de facto* standard) or a BBC Microcomputer Model A or B. By October 1982 the scheme was extended to primary schools, with the opportunity to have the cost of the Research Machine 480Z, BBC Micro or a Sinclair Spectrum covered. In return for receiving the subsidy, schools were expected to provide teachers to attend a LEA in-service training programme[19].

The *Micros in Schools* scheme is how the BBC Microcomputer got its reputation as a schools machine, but as we have seen, the broader Computer Literacy Project played a significant role influencing perceptions of computing at home and specifically for enthusing a generation of adults and parents in the future opportunities presented by computing. It is a general misconception that the BBC Microcomputer was aimed only to be an educational machine. The Computer Literacy Project was targeted at a broader audience, particularly adults who did not have any previous experience of computers, and as such formed a vital role in defining the culture of home computing in Britain. BBC microcomputers were aimed at the home market, as an accompaniment to the television programmes and giving people a 'standard' model on which to learn. While BBC Micros were encouraged in schools, it was not the main aim of the BBC to create a schools machine. They wanted to work with a range of other organisations, both in and out of the MEP, who were interested in promoting computers in schools.[20]

5 Meaning in the Home

The Computer Literacy Project had the grand ambition to change the culture of computing in Britain's homes. But what was the impact of the project and did it promote a cultural shift in attitudes to computing? We have recently collected evidence of the Computer Literacy Project's lasting influence and legacy through an online survey. 372 respondents answered a questionnaire on the influence of the Computer Literacy Project and the BBC Microcomputer on their early and subsequent computing careers. Rather than identifying the importance of school and classroom learning, many identified the influence of their parents in developing their enthusiasm for computing[21]:

> *My dad was a science teacher at a local school, so he'd always bring home computers. I grew up surrounded by them and would spend some of my spare time learning to program in BASIC, copying code out of magazines so I could make games.*

Stef, Respondent 1755262610

---

[19] Letter from Bob Salkeld, to Don Steel, EO CE North, 1st July 1981, R99/123/1 BBC Microcomputer Acorn Computers Ltd.

[20] Interview with Richard Millwood, formerly of the Computers in the Curriculum project at Chelsea College, University of London, which was sponsored by the MEP, 27/03/2012.

[21] This was part of an online questionnaire in 2012 targeted at individuals who were interested in personal computing in the 1980s. The questionnaire was emailed to 150 individuals and sent out on Twitter; 372 people started the survey and 292 (78.5 per cent of the total) completed it. The larger findings can be found at: http://www.nesta.org.uk/bbcmicro.

> *My father. He was a university graphics lecture and graduated to multimedia design in the 80s. We would bring computers home over the summer holidays and eventually bought one for me when I was in my mid-teens.*

Tom, Respondent 1753570087

> *My Dad brought home a Sharp MZ-700 computer from work when I was seven. It was literally the most incredible thing in the world. I became absolutely fascinated by it, and would sit with him as a kid while he entered some of the sample computer programs from the (extensive) manual. With time, and by studying the sample programs, I started to write my own.*

Greg, Respondent 1752697753

> *Parents did punch cards and paper tape... Pestered them for a Spectrum 48K which we used to game and program, and in summertime, my Mum (now a teacher) would bring home a BBC from school, ostensibly for her own familiarization, but it also got used for 'Yellow River Kingdom' and 'Elite'...*

Lindsay, Respondent 1756199469

For others the computer was inextricably linked to the sense of novelty and the future:

> *I was fascinated by robots and computers from a very early age. Probably inspired by Star Trek and other SciFi, but I was interested in real computers not science fiction.*

Martin, 1749329528

> *I was amazed about computers since I was 8 years old. They were kind of mystical objects to me. Perhaps I felt that they represented the future, and also my future. I wasn't able to have one though until I was 16 years old.*

Hector, Respondent 1746442794

> *My Dad bought me a BBC Micro computer when I was about 10 - that would have been around 1984. Me and my younger sister used to love to copy lists of code instructions from manuals for hours, until we got the end result of our names flashing in cyan and moving across the screen. The code we'd copy got more complicated, until we were moving objects across the screen! I think it gave me a love of finding out how things work, and how you could change code and control the output. I remember my Dad saying to us that we really should learn, as 'computers were the future'.*

Katherine, Respondent 1744306281

> *The BBCs computer literacy programmes brought attention to computers and IT as a possible career to lots of bright contemporaries searching for a possible future career.*

Tim, Respondent 1750893805

I visited a friend's house whilst at school where he showed me a mouse and that was it. I could see the future play out before me.
Matt, Respondent 1762010972

As Selwyn (2002) has shown, the notion of the computer as an inherently 'educational' machine has been a powerful influence on both the conceptualisation and consumption of information technology, but this 'hype' around the educational computer is noticeably at odds with the actual but often ignored impact of computing technologies at home. As borne out in responses to our questionnaire on the BBC Microcomputer. Respondents rarely talked about the educational value of the machine but did talk of their excitement about doing things with computers at home, particularly playing (and sometimes writing) games:

Both Star Trek and Blake 7 made me interested in science fiction and the art of technology. Games arcades with Space invaders in 1977 and watching Pong played on Crackerjack intrigued me as to how they actually worked.
Ian, Respondent 1760350457

Aside from the BBC Micro - I had also a Sinclair spectrum which I played various games on - and later used as a rudimentary line plotting device as part of a camera dolly set up.
Philip, Respondent 1747647713

After the wooden-box-hex-keypad computer, dad bought a TRS-80, then a BBC. I was allowed to go on the BBC. I typed in programs and fiddled with commercial games.
John, Respondent 1749765016

The BBC Micro was probably the most important machine I owned and worked with… It was very open, encouraging exploration and tinkering. Its limitations were a challenge, forcing economical and ingenious approaches that have stood me in good stead ever since. Its community of users and programmers were welcoming and shared their knowledge and passion freely. I owe my career to the BBC Micro. I have founded successful startups in Silicon Valley and worked for Apple and Google. I earned a PhD in Artificial Intelligence. I was inspired by my BBC Micro.
Dan, Respondent 1744975368

BBC Micro was a complete machine because you could do scientific experiments and produce analysis with it. It established a culture for making computers do useful things other than play games or write letters. It gave me opportunity and was empowering. When used in the Domesday project it brought 'tomorrow's world' to the present.
Bill, Respondent 1745918926

These quotes show how the discourse of 1980s home computers is often about fiddling, tinkering and playing at home, rather than any real sense of use or educational value. The machines facilitated social interaction and gave respondents to the questionnaire a sense of freedom and empowerment:

> *Freedom to do whatever I want to do. As a child, able to make my own games. Use a vessel for self expression.*

Memo, Respondent 1752019607

> *In 1982 my mum bought a ZX81 with her first paycheque from a part-time job. I was 13 at the time. As soon as she switched it on, I was fascinated. Here was a machine that could do anything and, even more impressive, I could make it do things.*

Dan, Respondent 1744975368

The open and accessible nature of early home computers invited consumers to explore and experiment through activities such as programming and gaming. The machines themselves did not have specific use – it was the pleasure of making them do something by programming and bringing people together through the technology that defined the meaning of the BBC Microcomputer for users in Britain's homes.

6 Constructing Computer Enthusiasm Today

Today there is a renewed interest in Britain developing a fresh generation of computer programmers. Unlike the late 1970s and early 1980s this is not because of a fear of what may happen to the structure of work in Britain in the future, but because of a real concern now about the drop in the number of students studying computer science and an anxiety that current ICT education in schools is failing students (Royal Society, 2012). Our education system is perceived to concentrate too heavily on using commercial software packages such as Microsoft Office and not enough on how computers actually work. The lack of programming experience is also cited as a problem with modern education courses.

New technological initiatives, such as Scratch, a programming language developed at MIT that makes it easy for children to create interactive stories, animations, games and share their creations on the web, and the UK's Raspberry Pi, that offers a credit-card sized computer that plugs into a TV and keyboard at the cost of only $25 plus tax, are hoped to present some solutions. Others hope that computer clubs, such as Young Rewired State which aims to get young children teaching themselves to code, and Code Club, where volunteers run programming after-school clubs, will present more social environments to enthuse children in programming. But most of there isitatives are still focused around the school and formal education. Even after school clubs happen in a education environment, be it after formal lessons, rather than in the home.

Selwyn (1991) has highlighted how during the 1980s the concept of the 'educational computer' was enthusiastically pursued by the government, the IT industry and the media, often with the ambition to sell more machines, support industrial policy and modernise schools. However such analysis fails to acknowledge the wider role of the BBC's Computer Literacy Project, and the construction of the BBC

Microcomputer as a home machine where families could share programming as an activity and reflect their own social and cultural values in their consumption of the machine.

Today's renewed interest in computer literacy presents Britain with another opportunity to enthuse a new generation in the creative uses of computing and gives us an opportunity to reflect on initiatives from the past. Few programming initiatives today focus on the importance of social interaction in the home or parents' contribution to learning. Instead they look to online social networks or schools to recreate the environment of the BBC Computer Literacy Project and BBC Microcomputer. None of the current initiatives are working with a broadcaster to reach into adult education, or aim to significantly challenge the cultural perception of programming in the home. Given the ambitions and the perceived success of the original BBC Computer Literacy Project, there is obviously a need to construct our notions of computer literacy well beyond the utility of the machines and their educational value. Computing has real social and cultural value, and by failing to acknowledge the role of the home we are missing an opportunity to reinvigorate the culture of programming in Britain.

References

CET, Microelectronics: their implications for education and training: a statement. vol. 2, p. 16. The Council for Educational Technology, London

Goldwyn, E.: Now the Chips are Down. Horizon, BBC 1, tx (March 31, 1978)

Haddon, L.: The Home Computer: The Making of a Consumer Electronic. Science as Culture (2), 7–51 (1988)

Linn, P.: Microcomputers in Education. In: Mackay, H. (ed.) Understanding Technology in Education, pp. 193–233 (1991)

OECD, Information Computer Communications Policy, 5. Microelectronics, Productivity and Employment (1981)

NEDO, The Microelectronics Industry: A Report by the Electronic Components Sector Working Party, SWP (1978)

Selwyn, N.: Learning to Love the Micro: the discursive construction of 'educational' computing in the UK, 1979–1989. British Journal of Sociology of Education 23(3), 427–442 (2002)

Radcliffe, J.: BBC Continuing Education TV and Robert Salkeld, Ass. Senior Education Officer (eds.) Towards Computer Literacy: The BBC Computer Project 1979-1983, BBC Education (1983)

Royal Society, Computing in Schools: Shutdown or Restart? (2012), http://royalsociety.org/education/policy/computing-in-schools/report/

Woolgar, S.: Configuring the user: the case of usability trials. In: Law, J. (ed.) A Sociology of Monsters: Essays on Power, Technology and Domination. Routledge, London (1991)

Reflections on the History of Computer Education in Schools in Victoria

Arthur Tatnall[1] and Bill Davey[2]

[1] School of Management and Information Systems, Victoria University, Melbourne, Australia
Arthur.Tatnall@vu.edu.au
[2] School of Business IT and Logistics, RMIT University, Melbourne, Australia
Bill.Davey@rmit.edu.au

Abstract. This paper traces the introduction of computing into schools in the Australian State of Victoria. Told from the point of view of two active participants, the story exposes a number of themes that resonate with experiences in other countries. From its beginnings in the 1970s on borrowed or shared minicomputers and the use of punched cards for teaching programming in conjunction with facilities at local universities, progress in the 1980s was rapid after the advent of the relatively low cost microcomputer. This article tells the story of how computer education developed in Victoria in the 1970s and 1980s, leaving discussion of more recent history for another time.

Keywords: Computer Education, Schools, Victoria, Microcomputers, Curriculum, Professional Development.

1 Introduction

Australia is an interesting country in which to study the development of computing. One of the very first computers (CSIRAC) was developed by the Commonwealth Science and Industry Research Organisation (CSIRO) and exists as the only remaining complete first generation computer today. (It is on display at Museum Victoria in Melbourne.) Australia is also well known for early adoption of new technologies and their rapid incorporation into the education system. In the State of Victoria in 1974 Monash University provided the specially created MONECS system to allow schools to teach programming using mark sense-cards. Victoria embraced computers from a very early stage and so this story relates some of the history of the introduction of computers into education in an environment where no global exemplars existed. The story is clearly seen through the eyes of two people who were present at the centre of developments and it is their reflection that flavours the choice of reported events. It involves many different human and non-human actors (Callon 1986; Latour 1991; Latour 1996). The human actors are readily apparent, but it is also important to consider the influence of non-human actors such as incompatible microcomputer systems, available software and professional development.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 243–264, 2012.

2 Background of the Authors

Both of the authors completed university science degrees going on to become Secondary School Maths/Science teachers in the 1970s. Arthur Tatnall majored in physics at the University of Melbourne. This course contained one unit, in its third year, on computing that involved FORTRAN 4 programming using punched cards on an IBM mainframe. After completing the science degree and a Graduate Diploma in Education he then went on to become a school teacher and did not touch computing again until teaching at Watsonia High School in the mid-1970s. In the late 1970s he studies for a Graduate Diploma in Computer Science at Latrobe University. Bill Davey took a science degree majoring in pure and applied mathematics at Monash University and completed a unit of computing in that University – an important institution for the development of computing in the Australian context. He started teaching at Bendigo High School and undertook a post graduate computing course, at Bendigo Institute of Technology, that included languages such as FORTRAN, ALGOL and COBOL as well as data processing subjects. At the Bendigo High he taught programming to year 10 students in 1970, 1971 and 1972, initially using punched cards sent to Monash University and subsequently using punched cards run at the local Institute of Technology. Bill and Arthur met while teaching at Watsonia High, after which Bill became a member of the *Mathematics Standing Committee* for the Victorian Education Department at a time when the State Government was moving quickly to investigate the introduction of computers into schools. This put him at the centre of activities at the very start of the growth of educational computing in the State of Victoria.

3 Schools in Victoria in the 1970s and 1980s

The Commonwealth of Australia is a federation of six states and two territories each having a considerable degree of independence. Constitutionally, state governments have responsibility for school education, but in matters of perceived national importance the Commonwealth Government adopts a policy position and supplies funding for specific education projects (Tatnall, 1992). It is, nevertheless, the State Government Education Ministries that determine school curriculum, and how it is supported and delivered. In the 1970s and 1980s Arthur and Bill were both employed as secondary school teachers by the Victorian Ministry of Education.

In the 1970s and 1980s the State of Victoria had several different types of schools. Firstly there was the divide between Government and Non-Government (Independent) Schools with Non-Government schools making up around 30% of the total. As the name suggests, Government Schools were controlled directly by, and received their funding from the Victorian Ministry of Education. Their teachers were appointed, their conditions determined and their overall curriculum guidelines set by the Ministry. Non-Government schools were of two main types: Catholic Schools and Independent Schools. In each case these schools then received no funding from the State, but did receive some funding and support from the Commonwealth Government. They employed their own teachers and determined their own curriculum within broad guidelines laid down by the Victorian Ministry of Education (Tatnall & Davey, 2008).

In the Government School sector students up to the age of about twelve attended Primary Schools and then moved to either High (Secondary) Schools or Technical Schools. The idea of dividing students into two streams for their post-primary education was to allow that some students were more academically inclined, while others needed a more practical education. In the late 1980s these two divisions were merged into what amounted to comprehensive Post-Primary schools. Administratively, each of these school types was attached to its own division within the Ministry of Education. These divisions were often non-cooperative, each pursuing their own policies and directions. This was in evidence in the Computer Education area, particularly in the often apparent friction between the Secondary and the Technical Divisions.

4 School Computing Up to the Time of the Microcomputer

Prior to the 1970s, for most people the idea of introducing school students to the use of computers, or of a school owing its own computer was difficult to imagine. School computing had begun in Victorian schools when a small number of minicomputers started to appear in the early 1970s (Tatnall, 1992; Tatnall & Davey, 2004b). This typically resulted from the exposure of particular teachers to computing during their university studies. In 1972, for example, Burwood High School in Melbourne was loaned a PDP-8 computer by Digital Equipment (Salvas, 1985) and in 1973 McKinnon High School received a Government Innovations Grant to enable the purchase of an 8k Wang computer costing over $10,000 (AUS) and requiring an annual maintenance contract of 15% of the purchase price. These early computers were typically used by school mathematics departments almost exclusively for the teaching of programming (Salvas, 1985) and so had very little overall impact on other aspects of education.

A bigger early impact on schools however, was the introduction of the Monash Educational Computer System (MONECS) making it possible for an average school to provide students with hands-on access to a computer before the PC became available. In 1974 a group at Monash University produced a system using mark-sense cards that allowed a class of 30 students to each get two runs in a one-hour period (Monash Computing Museum, 2003). The MONECS system was used to teach programming in FORTRAN or BASIC. At this stage schools saw computing as a branch of mathematics concerned with algorithm design (Tatnall, 2006; Tatnall & Davey, 2004b) and the MONECS system provided a useful means of supporting this.

This situation began to change during the late 1970s and accelerated in the 1980s with the advent of the microcomputer. While microcomputers were much cheaper and more easily handled than minis and mainframes, in the late 1970s they were still quite foreign to most school teachers. In the late 1970s the number of microcomputers in Victorian schools began to grow rapidly but without any central direction from education authorities. In 1980 Anne McDougall was commissioned to report on how computers were then being used in schools and on the possibilities for their future use (McDougall, 1980). A few years later the use of Computers in Education in Schools began to be seen as an important national consideration and the Commonwealth Government noted the importance of introducing computers into schools. In 1973 the Commonwealth Schools Commission set up its *National Advisory Committee on Computers in Schools (NACCS)* to report on how it might help to provide funding for school computer education (Commonwealth Schools Commission, 1983; Tatnall & Davey, 2008). In its report *Teaching Learning and Computers in Schools* (Commonwealth Schools Commission, 1983) the Committee made comprehensive recommendations covering curriculum development, professional development, support services, software/courseware, hardware and organisation. The report indicated that priorities for curriculum development should be:

- The provision of 'Computer Awareness' activities for all students in the earlier years of secondary schooling.
- The integration of computing into the school curriculum: 'computers across the curriculum'.
- Optional, in-depth Computer Studies courses at the secondary level.
- Curricula which met the special needs of relevant disadvantaged groups.

As the first priority in curriculum development, Computer Awareness courses around the country were given a considerable boost. In the period 1984-1986, the Commonwealth Government provided $19m to support the program. One result of the Commonwealth involvement was funds to set up State Computer Education Centres in those states not already having them. These Centres aimed to provide both support and professional development to teachers involved in computer education.

4.1 School Timetabling

The energy around the concept of computers was evident in many ways. One example is the use that was made at Watsonia High School of the computing facilities at the nearby Latrobe University. Watsonia High had implemented a timetable that allowed students to study outside their age groups. The timetable was created from blocks of classes offered to students across multiple age levels. These blocks were created by first asking students what classes they would like to do, overlaying information from teachers as to ability groups and then maximising the number of choices from students within the blocks. This was a monumental mathematical task and was not helped by unstable teaching assignments (- teachers were often moved or promoted to another school during the summer break with little notice.) Latrobe University offered to help with the timetabling task and a computer program was written that significantly improved the whole process.

4.2 The Teaching of Programming with Mark-Sense Cards

Much of the computing that went on in classrooms before 1980 was justified as being part of a new approach to mathematics education. This manifested itself mostly in the teaching of computer programming. Programming was easy to justify, suited the largely mathematics teacher-base of enthusiasts at the time and could be implemented without the school needing to own a computer. Students could write programs and have them run asynchronously using any facilities available either at the school or in conjunction with a local university. This tradition started with the MONECS system but was quickly used in schools with microcomputers as cheap card-readers were available and a class of student programs could be batch processed and give every student something to do. The MONECS system used cards with removable chads, but card-readers using the simple expedient of two wires for each field, connected by the graphite trace of a mark sense card were readily available.

4.3 The Acquisition of an Apple II at Watsonia High School

In 1977 Watsonia High School, in the northern suburbs of Melbourne, obtained an Apple II microcomputer with 16Kb RAM, a television monitor and a cassette tape drive as the result of a Curriculum Innovations grant submission to the Federal Government. As a member of the Secondary Mathematics Standing Committee, Bill had been encouraged to apply for this grant. The computer came with a number of pieces of software. The slightly later 'Apple Tapes Introductory Programs for the Apple][Plus' booklet (Apple Computer Inc., 1979) lists these as:

- Little Brick Out – an arcade game using the game controls
- Color Demosoft – a low resolution color programming demonstration
- Penny Arcade – an arcade game using the controls
- Lemonade – a business simulation asking you to manage a lemonade stand
- Hopalong Cassidy – a high resolution graphics game
- Phone List – an address book program using the cassette tape system for storage of up to 150 names and phone numbers
- Brian's Themes – high resolution Moire patterns demonstrating graphics programming
- Alignment Test Tone – a test tone to help align the read/write heads on your cassette tape recorder
- Renumber/Append – a programming utility to overcome the hassle of needing new line numbers in a BASIC program or the need to add some new lines to the end of a program

With such a 'plethora' of software it was possible to show other teachers just how useful (or otherwise) a computer could be. At Watsonia High this allowed a new Computer Awareness subject to be added to the year 10 curriculum (- this is discussed later). Of course, in practice, the programs were only ever used for demonstrating what might be possible and the computer was used mostly for programming classes until more computers became available at the school.

5 Support from the Ministry of Education

During the period 1978-1980 the discretionary fund of the Director of Secondary Education was used to support microcomputer purchase in a limited number of Victorian schools. The Director General's Computer Policy Sub-Committee also commissioned Anne McDougall, from Melbourne University Education Faculty, to undertake a study of the potential uses of computers in Victorian schools (Tatnall, 1992; Tatnall & Davey, 2004b). Her report's recommendations (McDougall, 1980) included a major commitment to in-service education (professional development) of teachers, and suggested that adequate numbers of Computer Education Consultants should be made available to assist schools. The recommendations included support for the introduction of Computer Awareness and Computer Science courses, professional development for teachers, the standardisation of computer equipment, a resource library of computer education materials and the development of a range of courseware programs.

In 1978-1979 the Computer Policy Subcommittee of the Victorian Education Ministry Director General's Policy Committee produced a plan for the introduction of computers to schools. A key feature of the plan was that:

> *"There is an immediate need for post-primary divisions to appoint full-time/part-time regional consultants for 1980 to establish clearing houses and assist in school program development and to co-ordinate the development of appropriate skills within each region"* (Bainbridge, 1979).

The Computer Policy Sub-Committee agreed in the need to separate Computer Education from mathematics, and influenced the formation of three (divisional) Computer Education Curriculum Committees (Tatnall, 1992). After representations from one of the Regional Directors of Education and a group of Inspectors of Schools it also approved the appointment of three Computer Education Consultants with state-wide responsibilities. These (one year tenure) secondments were to commence in February 1980.

5.1 The Computer Travelling Road Show

In 1978 the Education Department's Secondary Mathematics Committee, after recognised the potential of computers in mathematics and other areas of education. set up its own 'Computer Education' subcommittee. Bill was appointed as a member of this subcommittee which then set up a 'Computer Travelling Road Show' which in 1979 commenced visits to schools around the State to promote the use of computers in mathematics education as well as in other subject areas. Members of the group, often including Bill, would travel in twos or threes, normally bringing a 16k Apple II microcomputer with cassette tape drive (on loan from Computerland in Sydney) to demonstrate computer applications involving graphics, mathematics, commerce and word processing (rather than just programming) to teachers at curriculum days and staff meetings.

A typical meeting with a school would involve the whole teaching staff of the school in a hall followed by demonstrations in the staff room. A mortgage calculator, lemonade and similar simulations were used to show the wide variety of potential applications possible with these simple machines. While some work was done with the mathematics departments on programming, this was generally due to the presence of a 'computer champion' in the school: a teacher with an interest in computing. The Road Show was almost always invited to schools by the School Principal or a member of the School Council.

Of course ‘little’ details were glossed over in demonstrating to teachers the immense potential of general software in the classroom. These little details included:

- The need to constantly tune the tape recorder when transport caused the heads to move.
- The skill of being able to listen to the sound of the digital recording playing back through the speakers and to be able to align the tape so that the input of programs started with the program and not some hiss before the program.
- Being able to remember the machine code command in order to do soft reboots when the inevitable crash happened. Apple owners knew the operating system resided at memory location 3D0 and so knew that 3D0G would cause a restart.
- Since the monitor was just a TV set using the RF output from the computer a demonstrator needed to be able to handle interference, poor performance of the RF unit and normal TV maintenance.
- The expensive RAM chips (about $500 for a 16k chip, or several weeks’ pay for a teacher at the time) were seated in chip sockets that moved with heat, a demonstrator learned to wear a woollen jumper to polish the pins before reseating them after a hot session.

Of course these details were never part of the talks. In the very first years of use of the computer voice synthesis a version of voice recognition, graphics programs including simulations and such were all able to be shown. Some physics and chemistry teachers used the computer’s games port to connect various scientific instruments, and all these applications became part of the show.

These Road Shows represented the Victorian Education Department’s first official recognition of the importance of computers in education (Salvas, 1985; Tatnall, 1985).

5.2 The Secondary Computer Education Committee

In 1980, the Secondary Computer Education Curriculum Committee was formed with a membership made up of the original members of the Computer Education subcommittee of the Secondary Mathematics committee, several members of the Board of Inspectors of Secondary Schools, the three seconded computer consultants and a number of practising teachers (including Arthur). The brief of this committee was the production of Computer Awareness course guidelines, the investigation of Computer Science as a discipline, the publication of computer education articles, the collection and propagation of public domain software and the provision of in-service education (Tatnall, 1992).

5.3 Computer Education Consultancy

An important school curriculum support mechanism used by the Victorian Ministry of Education in the late 1970s and 1980s was the Regional Subject Consultant. This was a time of decentralisation and the Ministry had set up twelve Regional Offices around the state and in Melbourne, where many administrative matters were dealt with at a local level. The regions were not in any way really independent, but were used as a means of localising policies and decisions made by the Ministry of Education. An important function of the Regional Offices was to act as a base for the Regional Subject Consultants who spent much of their time serving the curriculum needs of local schools (Tatnall & Davey, 2008).

The Consultants were practicing school teachers who were seconded from their schools, usually on a part-time basis, to work from their local Regional Education Office. They were chosen for their subject expertise, teaching ability, willingness to adapt to and lead educational change, and ability to get on with and work with other teachers. They were mostly subject specialists (in the case of secondary schools) and were appointed only for a period of twelve months at a time. The idea was that although they could be re-appointed for following years, they should not become permanent advisors who might then lose contact with the school classroom. Arthur and Bill were both part-time Computer Education Consultants in the early 1980s. Having to spend half of their time doing their normal teaching job in a school meant that there was little chance that a consultant could forget what it was like to be a classroom teacher! In the curriculum consultancy part of their job, however, consultants rarely had any interaction with school students, working instead with teachers and school principals. As some Commonwealth Government money went into funding the Regional Subject Consultants, although they themselves were Government School teachers their brief extended to servicing both Government and Non-Government schools.

Unlike most other subject consultants, Computer Education Consultants were pioneering a new area of education and had little in the way of established precedent, techniques or materials to assist them. A common starting point for teachers in thinking out how best to present subject matter to their students is to remember how they themselves were taught, but as the use of computers in education was an entirely new area and few teachers had any experience with it, most had little idea of where to begin. The task of the Computer Education consultants was thus to introduce, and offer suggestions on the use of computers in schools. Their work had various different forms including (Tatnall & Davey, 2008):

- Professional Development activities run at the Regional Office and open to teachers from any school – often based around use of a particular software product.
- Professional Development activities within a given school that involved discussion of how some aspects of computing could be taught or how computers could be used in particular subject areas.
- Demonstration of educational software, and discussion of how it could be used in the classroom.
- Consultations with individual teachers on curriculum related matters.
- Investigation of possible school administrative uses of a computer.
- Individual consultations with school principals.

Consultancy was a very hectic life. Teachers considered the computing consultant to be their first port of call when computers broke down, when the Principal needed convincing, when classes required teaching ideas and when in-service education was required. In an era before mobile phones a day would start at the school or regional office answering machine significantly before school opening time. Calls that came in after hours would be returned to at least the school office (which opened early in most cases). A day would usually involve a morning and afternoon visit that had been pre-planned and two or three emergency stops at schools where a problem could be solved with a known remedy or the dropping off of materials. Many evenings would see a school receiving either a general staff talk after classes had finished for the day or a meeting with the Ministry of Education or the Year 12 Computer Science group. To try to control this impossible workload several tactics were put in place at a very early stage. Teacher self-help groups (- VITTA and CEGV are mentioned elsewhere in this article) were created. Large and small conferences were used to try to create champions around the State. A typical small conference was held in July 1979. The day commenced with a panel discussion at 9:15am where 'experienced' teachers from the Northern Region detailed their use of computers. At 11:00 am a session of workshops was used to introduce new teachers to writing computer related activities into the general school curriculum. After lunch another workshop showed teachers how to use the available software to put these curriculum ideas into action. This conference involved 27 teachers representing 16 schools, and others were repeated all over the State.

In the mid-1980s several Regional Offices experimented with the concept of a General Curriculum Consultant, rather than Subject Consultants. These teachers were typically seconded full-time and were chosen for their broad view of the school curriculum. They would still do subject consultancy work, but would also work on other more general curriculum tasks. In 1983 and 1984 Arthur became a General Curriculum Consultant before moving on to a position at the State Computer Education Centre in 1985.

5.4 Subject Teacher Associations and Conferences

Several Subject Associations showed an early interest in the use of computers in schools, particularly the Mathematics, Science and Commercial Teachers' Associations. A new subject association: the Computer Education Group of Victoria (CEGV) was set up in the early 1980s. It is interesting to see that the names of a quite small number of individuals keep coming up in the history of educational computing in Victoria, often in connection with these subject associations.

5.4.1 The Computer Education Group of Victoria (CEGV)

The CEGV was formed in the late 1970s as an association of academics (mainly from education faculties), computer salespeople, some teachers and others interested in the use of computers in education. It came into prominence in 1979 when it launched the first Computer Education Conference in Australia. The CEGV, and its counterparts in other states, exerted a considerable influence on computer education through professional development activities, annual conferences, journals and the provision of other publications and resources. Both authors were members of the CEGV during this period and Arthur was elected CEGV President in the late 1980s. The CEGV ran a quite influential annual Computer Education conference during this period. In the mid-1980s, the various state Computer Education Groups formed the Australian Council for Computers in Education (ACCE) which then began to run an annual

Australian Computers and Education Conference (which was held in a different State each year) and to lobby for computers in education at a federal level.

5.4.2 Victorian Apple Computers in Education (VACE)

Another type of group to emerge in the early 1980s was the 'user group'. In the early 1980s computer education had not progressed to the stage of being, to any degree, hardware independent. Schools using Apples had little to discuss with those using BBC, Cromenco or Commodore computers and their software and applications had too little in common. In September 1982 the authors, along with a secondary school principal, formed the Apple users group MACE (Melbourne Apple Computers in Education) which soon expanded its reach to the whole state and became VACE (Victorian Apple Computers in Education. These groups were formed at the grass roots by teachers to share knowledge between those using Apple II computers. From small beginnings VACE grew to have a membership in excess of 200 schools (Tatnall, 1985). Unlike the CEGV, VACE was very much a 'grass roots' organisation involving mainly practicing teachers. VACE conducted about two meetings per school term and fulfilled an in-service function. It had its own services section and software library and the VACE charter listed aims covering in-service education, helping with hardware problems and opportunities, software swapping, providing libraries of books and software and bulk buying schemes. The organisation was furiously busy for about 5 years then became irrelevant as the number of teachers with self-sufficient skills reached critical mass and hardware support ceased to a critical issue.

Although the Apple II was by far the most common computer used in secondary schools, the TRS-80, Acorn BBC computer, locally made Microbee and various 'industrial' computers used by Technical Schools were common enough to also need their own user groups. Many then delivered hours of useful in-service training and technical support. The importance of most of these groups sharply declined with the reduced emphasis on specific computer hardware of the later 1980s.

5.4.3 The Victorian Information Technology Teachers Association (VITTA)

In 1988 the Course Management Committee for HSC (Higher School Certificate) Computer Science (see later) saw a need for more support for teachers of the discipline of computing than was then available. In other subject areas teachers were principally supported by strong teacher subject associations, such as the Mathematics Association of Victoria, the Commercial Teachers Association of Victoria and so on. The much more general CEGV was so broadly based, covering teacher education and primary teaching so strongly that the specifics of computer science were largely neglected. On approach from the organisers of HSC Computer Science the CEGV agreed to call a meeting of Computer Science teachers at Melbourne High School. This was attended by teachers from all over the State and these people voted to create a new organisation dedicated to support of Computer Science teachers. The Victorian Information Technology Teachers Association (VITTA) was born from this meeting. (Bill and Arthur were leaders in this development). The aim of the organisation was to represent IT teachers at the various levels within the Education Ministry, to provide support materials for those preparing students for HSC Computer Science and the new VCE (Victorian Certificate of Education) ICT subjects (see later) and to provide general in-service education of teachers. A particular interest was to expose teachers to the industrial setting of computing through site visits. Many teachers could appreciate that much 'real computing' was done in a business setting, and that they themselves did not have sufficient knowledge of the way that business operates or what it does. The answer was to arrange a series of monthly meetings of VITTA, many of which would involve site visits. These meetings would typically begin at 7.30pm and go for 2 hours. The program of delivery of these visits for 1989 is instructive of the intentions of the group (Tatnall & Davey, 1989):

- February – Kambrook factory to see how a modern factory operates with process-control technology and computer-assisted management.
- March – Discussion session of the new VCE Information Technology Field of Study, at Parade College. This focused closely on the 'Common Assessment Tasks' of the new course which were a range of assessment methods, all of which would be used to measure the progress of students in the new course.
- April – Hewlett Packard: 'Micros and Mainframes'. Tour of HP's Australian Headquarters and a series of talks on topics such as HP's New Wave software, RISC technology and computer peripherals.
- May – Ford Motor Company's Broadmeadows production line: 'Robots and Production'. Tour of the Ford plant with special attention to the use of robots for welding and sub-assembly. Comparison with the other 'human operated' parts of the production line. Discussion of the social issues of robots replacing workers, quality control with human workers vs. robots, boredom on a production line and training of workers.
- June – Space Time Research: Supermap CD. The use of compact disk optical storage technology.

Monthly meetings each attracted about 35 teachers for the first half of 1989 but by 1990 the group had grown in strength to provide a large stream in the annual conference of the CEGV. As VITTA became recognised, both officially by the Education Ministry and by the teachers of the new Information Technology subjects, another theme emerged from the massive restructure of the high school years dominated by University Entrance Examinations.

Over the decades form the 1950s to the 1990s university entrance had moved from domination by the universities to control by the Education Ministry. The controlling body moved through name and constitution from the Victorian Universities and Schools Examination Board, which set a curriculum and final examination for every subject recognised for university entrance through to (in 1989) the Victorian Curriculum and Assessment Board. By the end of the 1980s a different philosophy was guiding the final year curricula and issues such as gender equity and accessibility were seen as more important than entrance standards.

These forces culminated in a complete re-engineering of the curriculum of the final years of high school. The issue that affected VITTA was the demise of one area of study – Secretarial Studies. A whole cohort of teachers in schools had been devoted to the area to be scrapped by the new VCE. These teachers were typically members of the Victorian Commerce Teachers Association (VCTA), a large and powerful support group. This group intended to move their members from secretarial studies into the new information technology subjects (as both typewriters and computers had keyboards, and computers were used to replace what the typists previously taught). For a short time both VITTA and the VCTA provided support for teachers of the new subjects, but the VCTA soon found they had few resources and that teachers were deciding if their area was either commercial or computing. Senior members of the VCTA approached the leaders of VITTA offering support in the form of accommodation and administrative support if VITTA were to become jointly sponsored by the CEGV and the VCTA. This was agreed to and VITTA moved into the headquarters of the VCTA and proceeded to support both the outgoing Computer Science and Secretarial Studies Teachers as the new Information Technology subjects began in schools.

5.4.4 The Australian Computer Society (ACS)

The ACS is the organisation representing computer professionals in Australia and maintains an active interest in the use of computers in education. In the 1980s it worked with the State Computer Education groups to achieve this end. Its direct influence at the school level was not large, but was significant particularly in the area of career education. Prominent ACS members frequently visited schools, on request, to speak to students about the computer industry and how computers were used in the business world. The ACS also ran an annual conference. In 1985 it ran the First Pan Pacific Computer Congress (PPCC) and Arthur Chaired its Schools Congress, attended by almost 1,000 students from Victorian schools. Arthur also chaired an ACS Schools Congress at the 1987 Australian Computer Conference.

6 The Victorian State Computer Education Centre (SCEC)

In Victorian schools, the early development of Computer Education was ‘bottom up’, beginning with the efforts of a small number of teachers and it took some time for the Education Department itself to become sufficiently interested to set up any form of central involvement. When the Computer Education explosion began in 1983 and the Education Department saw the need for some form of ‘top down’ planning and control, it became clear that there was a need for some central focus for computer education in the state. Policies of devolution and school-based decision making not-withstanding, in the early to mid-1980s the Education Department of Victoria still retained a quite strong central administration which facilitated the speedy and widespread adoption of computers in education while promoting a fairly common view of how computers could be used in education. There were, we believe, several key aspects of the computer education situation in Victoria that distinguished it from other curriculum areas, and that have benefited from a central operation. The first of these was the small number of people with expertise in the area: when a human resource is limited, it is a common response of any controlling group to centralise it, and this is what happened in the case of computer education (Tatnall, 1992).

Formation of the State Computer Education Centre (SCEC) of Victoria in 1984 was an important result. SCEC was set up in temporary premises at the old Moorabbin High School with 10 seconded staff, along with twelve Regional

Computer Education Centres staffed by seconded teachers. In 1985, all positions were advertised and staffing at SCEC was formalised with the centre headed by the Senior Computer Education Officer with a Software Co-ordinator, Professional Development Co-ordinator, Curriculum Co-ordinator, Educational Computer Systems Analyst and Equal Opportunity Officer holding Vice-Principal Positions. Seventeen Senior Teacher positions (five at SCEC and twelve in the regions), and four Assistant Teacher positions made up a total staff of twenty-seven professional officers. Arthur Tatnall was appointed to the position of Educational Computer Systems Analyst at SCEC.

SCEC played a significant role in setting the direction of educational computing in Victoria for the next three years. It developed policy, produced curriculum documents, evaluated and distributed educational software, evaluated computer hardware and systems and produced the 'recommended list' of computer systems for use in schools, facilitated interstate contacts and the sharing of resources, conducted professional development activities, and generally co-ordinated computer education in the state.

6.1 Recommendation of Computer Systems for Schools

There was little software compatibility between the early types of PC used in schools and so it made a big difference to a school's computer education curriculum whether it used Apple //, BBC, Microbee, IBM or Macintosh computers (Tatnall & Davey, 2004b). In every Australian state in the 1980s it was policy to recommend specific computer hardware for use in schools and so to comply with Government tender, offset and preferred supplier requirements. Government offset policy was designed to encourage local manufacture of computing equipment by requiring that *foreign* companies re-invest, in the state, 30% of the profits they made as the result of being nominated as a *preferred supplier*. The process of evaluating computing systems and recommending that preferred supplier status be conferred on a particular company is a task that needs to be done centrally. It would not be economically possible for such a function to be regionalised, let alone left to individual schools, and so it was a task performed by all State Computer Education Centres. This task alone could be used as a justification by Governments for setting up these Centres. One of the functions of the Victorian State Computer Education Centres was to control the proliferation of these brands by supporting only a limited number on a 'recommended list' so as to make realistic Education Department support possible. Evaluating computer systems and drawing up this list was Arthur's responsibility as Educational Computer Systems Analyst at SCEC.

6.2 The Australian Educational Computer That Never Was

Like several other countries (including the UK, New Zealand, Canada and Sweden), for both educational and industrial reasons Australia even commenced a project to design its own Australian Educational Computer (Tatnall, 1990), which fortunately (in retrospect) did not proceed past the design stage. Arthur had a direct involvement in much of this project as a member of the committee drawing up the Educational

Technical Requirements (Tatnall, 2012 (forthcoming)). By the late-1980s the rise to dominance in schools of the IBM-compatible PC (MS-DOS and Windows) computer and the Apple Macintosh made this project unnecessary.

7 Curriculum

One of the early curriculum directions was ‘computers across the curriculum’ and Apple II software like: Lemonade, Hammurabi and the First Fleet (convict) database showed the possibilities here. Several streams of school computer education however, soon emerged:

7.1 Programming in Mathematics Classes

It is interesting that, while computing was often introduced to schools by a member of the school’s mathematics department, computers per se quickly moved out of the mathematics curriculum (in favour of hand held electronic calculators). The introduction of Computer Science as a Matriculation (year 12) subject in 1981 was the death knell for computers in mathematics. This may be because of the prime place of mathematics as a prerequisite subject for science courses at universities. Teachers wanting their students to get the best possible university entrance score were reluctant to overload an already difficult mathematics curriculum with yet another topic. There had already been massive reforms involving set theory, known as ‘The New Maths’ and more change seemed like just too much. Eventually elaborate electronic calculators, capable of showing graphs, were used where a computer might have been relevant. Two very large projects did recommend the use of computers in mathematics. One was a complete overhaul of the mathematics curriculum expressed as a set of guidelines. The other was the Reality in Maths Education (RIME) project. In both cases computers were incorporated not as programming devices but as flexible means of displaying or calculating in other aspects of the curriculum and schools were given a very wide range of programs that could be run to demonstrate some mathematical concept more clearly on a computer screen. At the same time a significant number of teachers had been using LOGO to try to incorporate the ideas of Piaget and Papert into Mathematics classrooms, but these never became an official part of the mathematics curriculum of the State system.

7.2 Computer Awareness

Today there is no need to introduce secondary school students to computer technology, but in the early 1980s the situation was quite different. In Australia in the late 1970s and early 1980s considerable importance was put on the introduction of Computer Awareness courses in secondary schools when the first microcomputers started to appear. These courses were typically quite practical and involved a good deal of computer *use* rather than being wholly theoretical (Tatnall & Davey, 2006, 2008). In Victoria in 1980 the McDougall report to the Education Department Computer Policy Committee noted that:

> *"The case for computer education in schools is based largely on the need for children to be prepared for living in a society which is fast becoming dependant on the widespread application of computer technology."* (McDougall, 1980 :3)

It went on to suggest that:

> *"Computers have been called electronic 'brains' and there is no doubt that in the popular view, they are surrounded by an aura of mystery and are credited with powers they do not possess. The result is that most people outside the computing profession have attitudes of awe and fear towards computers and feel helpless and powerless in a highly computerised society."* (McDougall, 1980 :3)

The report saw a need for computer education to begin by imparting an:

> *"informed understanding of the power and capabilities of computers and also of their limitations … to every school pupil as part of a general education for modern living."* (McDougall, 1980 :3).

The Secondary Computer Education Committee put an early priority on the introduction of Computer Awareness in the middle secondary school years and in 1980 proposed guidelines for a Year 10 Computer Awareness subject strongly stressing the interdisciplinary nature of this subject matter and that Computer Awareness should not be equated with Computer Programming. The Committee noted that although computers had become indispensable in the operations of science, business and government, they did not currently play a significant role in Victorian secondary education (Tatnall & Davey, 2006). To justify the introduction of computers and related technology into the secondary school curriculum the Committee argued that as computers were beginning to exercise an important and growing influence on society, part of the school curriculum concerned with preparation for living in society should contain at least some elements of computer education (Secondary Computer Education Committee, 1980). The Committee's Year 10 curriculum guidelines noted that:

> *"… we define Computer Education in terms of computer 'awareness' – the possession of skills and knowledge to enable informed judgments to be made on the basis of what is seen or heard about computers. … the future citizen, ignorant of computers, will be functionally disadvantaged in a computer oriented society. In terms of 'social obligation' therefore, a strong case can be made for Computer Education. Since computers have significant social, political and economic consequences, an awareness of these consequences is essential to informed decision-making and to the democratic process."* (Secondary Computer Education Committee, 1980 :1).

The document specified that such a course should have the following content (Secondary Computer Education Committee, 1980 :4-14):

- Section 1 (15% of available time):
  i. Historical development of the computer: from the abacus, Pascal's adding machine, Babbage's Difference Engine and early electronic computers through to the microcomputer.

   ii. Structure of the computer: Analogue and digital computers. Input and output, processing, backing store.
- Section 2 (25% of available time):
   i. Hands-on experience in operating a microcomputer (booting the system, loading programs, running programs, creation and use of files, word processing, general computer usage).
   ii. Algorithms – the concept of an algorithm, simple flowcharting.
   iii. Elementary programming in BASIC.
- Section 3 (60% of available time):
   i. Use of computers in Government, industry/commerce, science/research, the arts, at home.
   ii. Implications of computer use for society. Political, economic and social implications.

7.2.1 A Computer Awareness Subject at Watsonia High School

In 1979 Watsonia High introduced into the Year 10 curriculum what was to be one of the first Computer Awareness subjects introduced in Victoria. This was to be a core subject taken by all 150 Year 10 students at the school. The new subject came mainly from the initiative of a science teacher (Arthur Tatnall) who was then also a member of the Education Department's Secondary Computer Education Committee. The idea was for the subject to run for the whole year and consist of three parts, each of one term's duration delivered by a teacher who understood and could relate to this area (Tatnall & Davey, 2004a). The teaching team consisted of this science teacher, a commerce teacher and a teacher of social science. Together they then set out to determine the requirements for the new subject which consisted of the following units:

- How a computer works, computer programming, history of computer technology.
- Business and commercial uses of information technology.
- The social implications of increased use of computers.

As secondary school curriculum in Victoria was school-based at this time and a matter to be determined by the whole teaching staff of the school, the next step was to convince the remainder of the teaching staff to vote for this change to the Year 10 curriculum, but given the interest in computers at the time convincing other teachers did not prove to be too difficult. In its first year, with only one microcomputer and some access to a mark-sense card system at a nearby university, the new subject involved a good deal more theory than practical use of computers. With additional hardware however, the subject became much more practical in subsequent years. It was immediately popular with the students, most of who were intrigued by the new computer. It was also seen as worthwhile by their parents, many of whom saw the possibility of better jobs for their children if they learned how to use these new machines. The subject remained in place at the school until the late 1980s by which time it had become redundant (Tatnall & Davey, 2006).

7.3 Subject Use across the Curriculum

Apart from general use in Primary Schools, many Secondary School teachers saw a place for computers in their own subject areas and a good deal of computer software was used in a wide range of subjects from History and English to Science and Secretarial Studies. As Computer Education Consultants, Bill and Arthur gave a good deal of attention to this topic in school visits.

In the mid-1980s a group within SCEC came to the view that the teaching of the discipline of computing in schools conflicted with the use of computers 'across the curriculum'. Their argument began with the view that since you did not need to know what was under the bonnet of a car in order to use a car, we should not be teaching students what goes on inside a computer as it was *using computers* that was all important. Arguments of equity were also raised. The counter argument was that it was important that Australian schools turn out students who were not just passive users of other people's technology and that future Australians should be able to understand the technology to the extent that they know its limitations and can adapt it to suit their own needs (Tatnall & Davey, 1989).

7.4 LOGO

The 1981 conference of the CEGV involved Seymour Papert as keynote speaker. LOGO had been an integral part of computer education in Tasmania after its introduction by Scott Brownell in 1975 and the energetic work of Sandra Wills in transporting the equipment all around that State. Following Seymour Papert's use of a turtle in conjunction with Logo programming, in 1979 a Tasmanian company produced the Tasman Turtle, designed for use with Logo and an Apple II computer (Denning Branch International, 2009; Powerhouse Museum, 2012). This proved popular both in making programming in Logo more concrete and as a demonstration of the power of robotics (Tatnall & Tatnall, 1987). Later a connection was established between Lego and Logo (Carter, 1988). A committed group in Victoria produced versions to run on the common microcomputers available, wrote books expanding on the possibilities and educational theory and generally served as missionaries for LOGO (Jones, McDougall, & Murnane, 2004). Logo was used at all school levels and in articles for the 1983 Australian Computers in Education Conference, Terry Malone discusses using Logo in Primary Schools (Malone, 1983) while Anne McDougall and Tony Adams advocated its use with senior secondary school students (McDougall & Adam, 1983). In the proceedings of the 1986 conference Tony Gilding and Jon Pearce investigate use of Logo to teach science concepts (Gilding & Pearce, 1986) and Chris Bigum offers some critical reflections (Bigum, 1986). Following the turtle, Logo was also used with robotics and control technology (Tatnall & Davey, 1986; Tatnall & Tatnall, 1987)

7.5 Computer Science (Higher School Certificate)

In 1981, as a result of many years of effort by a group university academics, Computer Science was first offered as a Higher School Certificate (HSC, Year 12) subject in Victoria, although personnel from the Education Department had little involvement in determining the nature and content of this subject. The public: parents, students and employers readily accepted HSC Computer Science and student numbers increased rapidly. In its first year 120 students from 10 schools studied Computer Science, but this number increased by over 50% in each of the next five years before making a slower increase to over 2200 in 1991 with virtually every secondary school in the state offering the subject.

The content of this subject included the following topics: computer structure and data representation, algorithms and modelling, programming languages, data structures, input/output devices, file structures, system software, and social implications (Victorian Institute for Secondary Education, 1984). In addition, students undertook one of the following two optional units: computers in science and engineering, or computers in business and government. Practical work was seen as an important part of the course. Arthur and Bill were both members of the HSC Computer Science Subject Committee and Computer Science examiners, and Arthur later became Chief Examiner in HSC Computer Science.

Not all teachers, however, were in favour of the new subject and from the mid-1980s when Computer Science was still in a rapid growth phase a number began to question its place. Their arguments had several stands: firstly some claimed that Computer Science was an elitist academic subject and was too difficult for some students and so should not be supported. Others noted that the ratio of girls to boys taking Computer Science was almost as low as that for physics and expressed concern that it was becoming a *boys' subject*, a concern which the subject committee shared and tried to address. Perhaps, however, the most damaging criticism came from those teachers who claimed that the presence of a specialist subject detracted from the move to encourage the use of *computers across the curriculum* as the demands made on school computing facilities by Computer Science classes made it difficult for others to obtain adequate access to the machines. They saw Computer Science and computers *across the curriculum* as adversaries and it took some time before they could be convinced that a school was more likely to have a large number of computers if it offered Computer Science, and these points of view were reconciled.

One of the curriculum offerings in Victorian Technical Schools was Year 11 Computer Studies (McCluskey, Adair, Feil, & Smulders, 1983), but this was taught rather differently than similar named subjects in High Schools as Technical School courses were rather more related to business training and employment than those in the High Schools. The Technical Division expressed a need for their schools to use 'industry standard' hardware and software with microcomputers based on the Z80 chip running the CP/M operating system (such as the Cromenco, Micromation and Pulsar) and business software such as WordStar and dBASE. Later, when the IBM PC appeared, there was a move to MS-DOS in these schools.

7.6 Information Technology Studies – VCE

In 1992 HSC Computer Science was replaced by the more general subject area '*Information Technology*' as part of the newly introduced Victorian Certificate of Education (VCE). At this time several subjects had been made redundant as they were seen as having died due to a perceived lack of interest. A particular case was *Secretarial Studies* which had occupied a comfortable niche as a vocational subject preparing (mainly) girls for entry into office and administrative jobs. The idea of teaching typing on typewriters was, however, now seen as anachronistic and the numerically large and influential group of Secretarial Studies teachers made redundant in this process produced a powerful pressure group which was influential in the development of new computing subjects (Tatnall & Davey, 2010), particularly as they might relate to word processing.

Computer Science was replaced with three Year 12 subjects: *Information Processing and Management*, *Information Systems* and *Information Technology in Society*. Each of these subjects comprised two units. An additional new subject: *Information Technology* was offered only at Year 11. The group designing these new subjects aimed to provide a new subject area with a much wider appeal. Bill was a member of this group.

8 Textbooks

By 1984 Computer Science had overcome the main problems of acceptance as a 'real' subject, mostly in that it was now able to contribute to a score for the competitive university entrance process. The growing number of students aroused the interest of all text book publishers and an Australian publisher approached us to provide for the largest market segment – year 11 students. The first text produced was indicative of the climate at this time: *Computer Science for Year 11* (Tatnall & Davey, 1985). The text was clearly aimed at preparing students for the year 12 Computer Science subject and it quickly became the top selling text for year 11 students in Victoria as it had been produced by two practicing teachers who were also part of the central process of curriculum development. By 1990 the climate had changed and our next text was called *Information Technology Studies* (Tatnall & Davey, 1990) , a reflection of the move away from the more technical name 'Computer Science' and reflecting the advancing values of those who wanted to make computing in schools available to all. One of the largest selling text for computing at year 12 by 1991 had the revealing title: *Information Technology: Theory, Application and Impact* (Woodhouse & Tatnall, 1991)

9 Conclusion

Computer education became a mainstream activity in Victorian schools in a period of less than ten years from the late 1970s to the late 1980s. This adoption reflected a climate of intense enthusiasm and vision for the microcomputer, and of the success of attempts to support and train teachers in a completely new area. Factors in this success were:

- the Travelling Computer Road Show,
- the State Computer Education Centre,
- Computer Subject Associations, and in particular
- the Computer Education Consultants.

These Consultants were taken from the classroom on the basis of their expertise and enthusiasm so that they could help their colleagues by using their full knowledge of the working conditions in schools. They were also extended trust by the people they helped as they came from the same background. The measure of their success is the speed with which they did away with the need for their services.

Those of us involved at the beginning of the Computer Education revolution in Australia had great hopes that the introduction of computers would make a profound difference, and significantly improve the quality of school education. In the late 1980s Arthur and Bill each moved into university Departments of Information Systems and largely out of school computing, bring to an end this history of computers in schools. A lot happened in school computing in the 1990s and 2000s, particularly with the use of the Internet, but that is not a part of our story.

References

Apple Computer Inc., The Apple Tapes Introductory Programs for the Apple][Plus: Apple Computer Inc. (1979)

Bainbridge, W.J.: Internal memo to all policy committees (November 1979)

Bigum, C.: Logo, Some Critical Reflections. Paper Presented at the Australian Computer Education Conference - Computers in Education: On the Crest of a Wave, Melbourne (1986)

Callon, M.: Some Elements of a Sociology of Translation: Domestication of the Scallops and the Fishermen of St Brieuc Bay. In: Power, Action & Belief. A New Sociology of Knowledge?, pp. 196–229. Routledge & Kegan Paul, J. Law. London (1986)

Carter, P.: Lego and Logo: the Bits Between. Paper Presented at the Australian Computer Education Conference - Golden Opportunities, Adelaide (1988)

Commonwealth Schools Commission, Teaching, Learning and Computers. Report of the National Advisory Committee on Computers in Schools. Commonwealth Schools Commission, Canberra (1983)

Denning Branch International (2009). The Old Robots Web Site - the First Wave of RoboticsRetrieved (May 2012), `http://www.theoldrobots.com/turtle1.html`

Gilding, A., Pearce, J.: Logo and the Exploration of Some Concepts in Science. Paper Presented at the Australian Computer Education Conference - Computers in Education: On the Crest of a Wave, Melbourne (1986)

Jones, A., McDougall, A., Murnane, J.: What did we think we were doing? In: Impagliazzo, J., Lee, J.A.N. (eds.) History of Computing in Education, pp. 83–90. Kluwer Academic Publishers/IFIP, Assinippi Park, Massachusetts (2004)

Latour, B.: Technology is society made durable. In: A Sociology of Monsters. Essays on Power, Technology and Domination. Routledge, J. Law. London (1991)

Latour, B.: Aramis or the Love of Technology. Harvard University Press, Cambridge (1996)

Malone, T.: Strategies for Using Logo in the Classroom. Paper Presented at the Australian Computer Education Conference - Could You Use a Computer?, Melbourne (1983)

McCluskey, D., Adair, I., Feil, C., Smulders, M.: Teaching Year 11 Computer Studies. Paper Presented at the Australian Computer Education Conference - Could You Use a Computer?, Melbourne (1983)

McDougall, A.: Computers and Post-Primary Education in Victoria: a Study of Needs. Education Department of Victoria, Computer Policy Committee, Melbourne (1980)

McDougall, A., Adam, T.: Logo for Senior Secondary School Students. Paper Presented at the Australian Computer Education Conference - Could You Use a Computer?, Melbourne (1983)

Monash Computing Museum. MONECS Deamon Educational Computer system (2003), `http://www.csse.monash.edu.au/museum/` (retrieved February 2004)

Powerhouse Museum, Powerhouse Museum Collection Search - Tasman Turtle (2012), `http://www.powerhousemuseum.com/collection/database/?irn=262955` (retrieved May 2012)

Salvas, A.D.: Personal communication (1985)

Secondary Computer Education Committee, Year 10 Computer Education: Guidelines for Secondary Schools. Education Department of Victoria, Melbourne (1980)

Tatnall, A.: The Role of the State Computer Education Centre of Victoria, Minor. Deakin University, Geelong (1985)

Tatnall, A.: Designing the Australian Educational Computer. Education 110(4), 453–456 (1990)

Tatnall, A.: The Growth of Educational Computing in Australia. In: Goodson, I.F., Mangan, J.M. (eds.) History, Context, and Qualitative Methods in the Study of Education, London, Ontario, vol. 3, pp. 207–248. University of Western Ontario, Canada (1992)

Tatnall, A.: Curriculum Cycles in the History of Information Systems in Australia. Heidelberg Press, Melbourne (2006)

Tatnall, A.: The Australian Educational Computer That Never Was. IEEE-Annals of the History of Computing (forthcoming, 2012)

Tatnall, A., Davey, W.: Computer Science for Year 11. Jacaranda Press, Brisbane (1985)

Tatnall, A., Davey, W.: Conceptual Development Through Robotics: Information Technology in Practice. Paper Presented at the Australian Computer Education Conference - Computers in Education: On the Crest of a Wave, Melbourne (1986)

Tatnall, A., Davey, W.: Making Information Technology Real. Paper Presented at the ACEC 1989, Backup the Future, Canberra (1989)

Tatnall, A., Davey, W.: Computer Science for Year 11: Information Technology Studies. The Jacaranda Press, Milton (1990)

Tatnall, A., Davey, W.: Improving the Chances of Getting your IT Curriculum Innovation Successfully Adopted by the Application of an Ecological Approach to Innovation. Informing Science 7(1), 87–103 (2004a)

Tatnall, A., Davey, W.: Streams in the History of Computer Education in Australia. In: Impagliazzo, J., Lee, J.A.N. (eds.) History of Computing in Education, pp. 83–90. Kluwer Academic Publishers/IFIP, Assinippi Park, Massachusetts (2004b)

Tatnall, A., Davey, W.: Early Computer Awareness Courses in Australian Secondary Schools: Curricula from the late 1970s and early 1980s. In: Impagliazzo, J. (ed.) History of Computing and Education 2, pp. 107–116. Springer, New York (2006)

Tatnall, A., Davey, W.: Computer Education Support Structures in Victorian Schools in the 1980s. In: Impagliazzo, J. (ed.) History of Computing and Education 3, pp. 1–22. Springer, New York (2008)

Tatnall, A., Davey, B.: The Life and Growth of Year 12 Computing in Victoria: An Ecological Model. In: Tatnall, A. (ed.) HC 2010. IFIP AICT, vol. 325, pp. 124–133. Springer, Heidelberg (2010)

Tatnall, A., Tatnall, B.: Logo in Control: Control Technology in Education. Paper Presented at the Australian Computer Education Conference - Tomorrow's Technology Today, Adelaide (1987)

Victorian Institute for Secondary Education, Handbook for, Year 12 Curriculum and Assessment. Victorian Institute for Secondary Education, Melbourne (1984)

Woodhouse, D., Tatnall, A.: Information Technology: Theory, Application and Impact. Dellasta, Melbourne (1991)

From the History of Russian Computer Science

Yakov Fet

Institute of Computational Mathematics and Mathematical Geophysics of the Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russia
fet@ssd.sscc.ru

Abstract. This article includes a few passages from the History of Computer Science relating to the events of the middle of the last century. The first part of this article contains a short biography of Norbert Wiener, who is considered to be one of the fathers of modern computer science. This part pays particular attention to Norbert Wiener's visit to Moscow in the summer of 1960. The other parts of the article focus on life and work of Aleksey Lyapunov, Leonid Kantorovich and Andrey Ershov. The outstanding professional achievements of these Russian scientists, as well as their moral perfection, can set an example for modern students and young professionals.

Keywords: Computer Science, Norbert Wiener, Aleksey Lyapunov, Leonid Kantorovich, Andrey Ershov, human quality.

1 Introduction

The works of modern humanists are characterized by a view of *human qualities,* which we consider to be very important. Individual qualities and characteristics of one person can hardly be measured or compared with the analogous qualities and characteristics of the other. Nevertheless, all of us use an intuitive (and, of course, individual!) scale to evaluate behavior, actions and statements of other people.

When speaking about prominent scientists, we often use the phrase '*A Man with a capital letter*'**.** And we think that, in this high estimation, the civic characteristics that find expression in behavior, actions, attitude to other people and response to their actions and statements etc. are probably even more appreciated than just scientific achievements. A Man with a capital letter is a high class professional and one of great moral excellence at the same time. In 1968, Aurelio Peccei, a recognized expert in the field of management, founded and headed one of the most powerful democratic movements of the XX century – The Club of Rome. The scientists who formed the Club believed that our society could be improved by the way of the growth and development of *human qualities* which are the main reserve and a powerful driving force behind progress.

In 1977, Aurelio Peccei wrote in his book '*The Human Quality*' [1]:

> *"Until our so called technological society becomes as well a human society, violence will proceed its triumphal demonstration and we will continue to fight special cases of this general phenomenon without understanding the origins of the violence."*

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 265–288, 2012.

We are lucky to find in our life those special, wonderful, charismatic people that emerge from time to time in the history of science as well as in the history of Computer Science. Except for their brilliant intellectual capabilities, they possess the treasure of personal characteristics that are invaluable to history and society. Moral perfection of such scientists can be a great example for modern students and teachers.

This article is dedicated to the outstanding scientists – humanists who lived among us, who worked and attained most important achievements of their lives before our very eyes. They really are a living example to learn from! And we believe that their contributions to the development of human qualities in our society are their greatest achievements.

2 Norbert Wiener

This article deals with the history of the Russian Computer Science, but still I decided to locate the part called 'Norbert Wiener' at the very beginning of it, because Wiener's '*Cybernetics*' really is, as a matter of fact, the beginning of the history of Computer Science – especially taking into account that '*Cybernetics*' was published in 1948, when the first successful computers were built.

***

There are some exceptional circumstances in the biography of Norbert Wiener that must have had a great impact on his personality at the very beginning of his life, affecting both his human qualities and his later scientific work. Norbert Wiener was born to a Jewish family on November 26, 1894, in the USA. His father, Leo Wiener (1862–1939), immigrated to America from Bialystock (Russia) in 1880.

The uncommon personality of Leo Wiener and his special role in the process of raising his children had an important place in the life and work of Norbert Wiener. When young, Leo Wiener was a strong follower of the philosophy of Leo Tolstoy. In 1880, when he was 18, Leo Wiener decided to go to Central America with his friend who was a Tolstoyan, too. They intended to found there a humanistic community. They swore they would never drink alcohol, smoke and eat meat. The friend was lost somewhere along the way, and Leo went through many adventures and at last arrived in Kansas City, Missouri, without a penny in his pocket. Having faced many obstacles to overcome, he managed to get the job of a teacher in foreign languages (later, he was a Professor of Slavic Languages at the famous Harvard University in Cambridge, Massachusetts). Leo got married in 1893. One year later, his son Norbert was born in the city of Columbia, Missouri.

Those who read the biographies of Norbert Wiener and his father often ask themselves a question that has been discussed a lot by both historians and geneticists: is it possible to inherit intellectual qualities? Really, it happens sometimes in the history of science and culture that several famous people come from the same family. However, I don't believe in the fantastic ideas of eugenics. Development and manifestation of mental abilities depends mostly on the influence of social environment, traditions, upbringing and education.

Norbert Wiener's talent manifested itself very early. He had a good reason to choose the title '*Ex-Prodigy: My Childhood and Youth*' [2] for the book about his childhood. Leo Wiener never concealed his intention to bring up child prodigies.

Norbert's father took complete control over his education. He was a very energetic and exigent man and maybe sometimes he went too far – but the facts speak for themselves.

We know many other examples of attaining high intellectual achievements in result of combination of natural abilities, favorable environment, upbringing and education. A striking example is the family of Aleksey Lyapunov and the story of his life and work. Aleksey Lyapunov was a brilliant scientist, teacher and humanist, he played a decisive role in formation and development of Computer Science in Russia. It is appropriate to explain here the importance of establishing specialized schools and the system of admission of the pupils on a competitive basis. In the 60s, a good example of the rational and effective method to organize school education was the Physico-Mathematical School (PMS) founded in Novosibirsk *Akademgorodok* (the Novosibirsk Academic Village) by Aleksey Lyapunov and other Siberian scientists.

***

So, Wiener's father took the boy away from the school and educated him at home for several years; the main subjects were mathematics and foreign languages. In 1903, the Wiener family moved to Harvard, Massachusetts. Norbert was admitted to high school and studied there for three years. After graduating from this school in 1906 (at 11 years of age) Norbert Wiener entered Tuft's College. At the same time, he was continuing home education.

He was awarded a bachelor in mathematics in 1909 (at the age of 14), whereupon he began graduate studies at Harvard. He studied mathematical logic and philosophy of mathematics, and was awarded a PhD degree in 1913. By the time he was under 19 years old, which was about 6 years less than the average age of a person who attained a doctorate.

***

His working life began in 1915. During 1915–16 he was a teaching assistant at the Philosophy department of Harvard University. After that, he tried many jobs. He worked for several months as a journalist for *The Boston Herald* and even wanted to join the military.

At last, with the assistance of mathematician W.F. Osgood who was a friend of Leo Wiener, Norbert became an instructor of mathematics at the Massachusetts Institute of Technology, where he worked for the rest of his live. His talent blossomed out at MIT. Maybe he finally found his real calling in mathematics; maybe his permanent work gave him a feeling of safety and self-esteem; maybe at the age of 24 the former prodigy attained stable equilibrium and was ready to become a genius. In any case, during the first ten years of working at MIT Norbert Wiener made his most striking contributions to abstract mathematics.

***

Wiener invented the word 'cybernetics' which comes from the Greek χυβερνήτης, meaning 'steersman'. In the Webster Dictionary the word '*cybernetics'* is defined as: *the study of human control functions and of mechanical and electronic systems designed to replace them, involving the application of statistical mechanics to communication engineering*.

In his book '*I am a Mathematician*' [3] Wiener says that *cybernetics* is the most appropriate word he could find "*to define the art and the science of control in the wide range of applications of this notion*".

Like many other great things, this book appeared as the result of a concurrence of casual circumstances. In the summer of 1947, Wiener traveled to France to attend a mathematics conference that was taking place in Nancy. There he met M. Freyman, who was a mathematician and worked for the publishing house '*Hermann & Co.*'. Wiener received from Freyman a proposition to write a book dealing with the main ideas of his works in the field of control and communication.

Wiener willingly signed the contract and continued his journey. The book was written in Mexico, in the late autumn of the same year, and dedicated to Arturo Rosenblueth. And the first lines of 'Cybernetics' told:

> *"This book represents the outcome, after more than a decade, of a program of work undertaken jointly with Dr. Arturo Rosenblueth, then of the Harvard Medical School and now of the Instituto Nacional de Cardiologia of Mexico" [4].*

The friendship and co-operation with Rosenblueth opened to Wiener, who was mainly a mathematician, the world of biology and medicine. The idea of a universal methodological approach to all fields of science emerged and rooted in his mind. Remarkable generalizations made by Wiener in 1948, rested upon a solid foundation.

Wiener's book appeared in 1948. It was published simultaneously in France and in the USA under the title 'Cybernetics or Control and Communication in the Animal and the Machine'. This book instantly turned Wiener into a kind of scientific superstar.

Since then, the name of Wiener and the word 'cybernetics' became inseparable from the new science and its numerous applications that by now are covered by the terms *informatics* and *information technologies*.

Talking about cybernetics, we have to pay special attention to the importance of *communication* or *information transmission* for the functioning of any natural organisms and technical devices. Now we say that informatics, which stems from cybernetics, is a group of scientific fields dealing with *general characteristics of information* and the ways and means of *creating*, *storing* and *transferring* information. It is hard to overestimate the significance of informatics to the modern society.

At present, informatics in all its applications depends upon computers. The first modern computers and Wiener's 'Cybernetics' appeared approximately at the same time.

***

In the Summer of 1960 (from June 27 to July 02) the First International Congress of the International Federation of Automatic Control (IFAC) took place in Moscow. More than 1,200 scientists from 30 countries arrived in Moscow. Actually, it was the first congress on such a large scale conducted in the Soviet Union. On one hand, this event could only be possible in the time of Khrushchev's '*thaw*' – and it was the first attempt to put an end to the isolation of Soviet science. On the other, it clearly showed a general interest in the achievements of Soviet science, and in particular, in the field of automatic control and cybernetics.

1,200 delegates – that was impressive! One of the active participants of the Congress 1960, Eric Nappelbaum (who was just an interpreter during the public speeches held by Wiener in Moscow), remembered that:

> *…many of the people who had come to the Congress were not intending to participate in the scientific event. They were very eager to know more about our country and took the unique opportunity to come and see everything with their own eyes. That was the reason why there were so many people not directly connected to the subject of Automatic Control [5].*

No wonder, that Norbert Wiener was greeted in Moscow with great enthusiasm. He participated in the Opening Ceremony of the Congress which took place at the State University of Moscow on the 27th of June. On June 28, he gave a lecture on '*Brain Waves and Self-Organizing Systems'* in the Big Lecture Hall of the Polytechnic Museum. As it often happens when a public speech held by popular scientist or poet take place, the Big Hall couldn't hold all the people who attended, so some had to sit on the stairs or in the gangways. The lecture had to be repeated on the 1st of July.

Of course, the interest was mutual. The main Soviet newspapers and magazines did their best to organize and publish discussions with Norbert Wiener – it is not every day you get the chance to meet such a great person in Moscow!

The magazine *'Voprosy Filosofii' ('Problems of Philosophy')* invited Norbert Wiener to visit its editorial office. A meeting between Wiener and Soviet philosophers was organized there on July 5, 1960. A shorthand record of this discussion was published in the Issue 9 of this magazine.

The magazine 'Priroda' ('Nature') published in its Issue 8, 1960 an interview entitled: *'Cybernetics and Humans. A Conversation with Professor N. Wiener'.* (The same interview was later included in the second Russian edition of Wiener's 'Cybernetics' as one of the Appendices) [6].

On June 30, 1960, the newspaper 'Literaturnaya Gazeta' ('Literary Newspaper') published materials of an interview with Wiener under the title *'Cybernetics and the Literature'* [7]:

Wiener:

> *I would like to tell you about my first Moscow impressions. I like the city and I am very impressed with the young Soviet scientists I met here. They are polite, proper and very nice people and excellent professionals. I would be pleased to have such colleagues.*
>
> *The work that I have been doing over the last 30 years has much in common with the work of the Soviet scientists. When I'm reading the papers of Academician Kolmogorov, I feel like I am sometimes reading my own thoughts – it's just what I wanted to say. And I know that he feels the same reading my papers. Both of us profit enormously from such collaboration.*

LG:

> *Could you tell us what are you working on right now?*

Wiener:

> *Right now I'm working on a second edition of 'Cybernetics'…*
> *At the same time, I'm writing a new novel. In my free time, I'm a writer. It's not just a way of relaxation. It's always interesting to me how the circumstances of the life affect a person's character and what destiny awaits him or her. My first novel 'The Tempter' [8] was published in November 1959. It deals with a conflict between a scientist's ideals and*

> *his wish to make a good career, which is very typical for America. The epigraph of the novel says: "To those scientists who prefer to look for the truth and not for earthly blessings". At the center of the narrative is the destiny of the engineer Gregory James who was born in Russia, in the city of Odessa, and immigrated to America before the First World War. My choice of the main hero was, in fact, not accidental. He is a Russian by birth – and Russia is very special to me. My father, Leo Wiener, was born in Białystok and immigrated to America in 1880. During all his life he was a passionate disseminator of Russian culture.*

(However, I have digressed.)

> *So, I'm finishing my new novel, 'What Lies Under the Stone', that will be published in December of this year. It deals with corruption in the TV production sector of the USA and with exploitation of talented children.*

LG:

> *Why have you chosen this title for your novel?*

Wiener:

> *Have you seen what happens when you turn over a big stone? You see the various things that accumulated there over the years. Something like that happens in my novel: during a legal process, the TV business shows its seamy side…*
>
> *In this novel, I wanted to explore the formation of a gifted child's character. This subject is close to me. After all, I was also a kind of 'Child Prodigy'...*

At the end of the conversation, Norbert Wiener said some words to the readers of the LG:

> *It is a great honor for me to communicate personally with my colleagues from other countries and to see a great interest and understanding of the questions of automatics. I am glad that me and my wife have come to this beautiful and friendly city.*

During the lecture at the Polytechnic Museum he was asked about his coming plans in literature. His answer was:

> *Last year I published my first novel and I intend to write another one together with my colleague, Doctor Asimov from the University of Boston. I never think about how many books I'm going to write. I find satisfaction in writing novels when I have some free time.*

Being a great mathematician and the founder of cybernetics, Norbert Wiener took his work in literature very seriously.

We know nothing about what happened to the novel *'What Lies Under the Stone'* and to the Wiener's plans of co-operation with Isaac Asimov …

Most of Wiener's writings are non-fiction books. We know that he created two autobiographies: *'Ex-Prodigy, My Childhood and Youth'* and *'I Am a Mathematician. The Later Life of a Prodigy'*. The two of his most popular works in cybernetics are *'The Human Use of Human Beings'* [9] and *'God and Golem, Inc.'* [10]. In these works, it can be clearly seen that, compared to his contemporaries, Wiener had a better understanding of the influence of technical progress on our society and that he believed in the power of human intellect, and human qualities.

In 1993, the MIT Press published one of his unknown works: *'Invention: The Care and Feeding of Ideas'* [11]. This novel about the life of an inventor in the pitiless modern society has a remarkable dedication:

> *To the Massachusetts Institute of Technology, a home for the creative intellect.*

One of Wiener's readers was the famous astronomer, President of the Armenian Academy of Sciences, Victor Ambartsumian.

P. Masani, who was a pupil of Norbert Wiener and his biographer, published in 1990 a remarkable biography entitled *'Norbert Wiener. 1894–1964'* [12], where he presents a letter from Ambartsumian to Wiener (dated Janurary 25, 1963):

> *Dear Professor Wiener,*
>
> *Many thanks for the copy of your novel 'The Tempter'. I have just finished reading the book and should like to say that it interested me very much. As you know we have here almost forgotten the times when there were private companies in our country. Therefore a novel describing the activities of companies as regards to application of new scientific ideas and technical inventions and the moral conflicts arising from these activities has opened to me an unknown aspect of the life of your country. Perhaps it will be useful for our young generation also to be acquainted with these problems.*
>
> *Therefore I think that it is worthwhile to publish here a translation of this book and I suppose to try to do this through State Publishing House. Of course, this is only my intention and I write this to you only to show how much I was impressed by your novel.*
>
> *Being an astronomer almost completely devoted to my science I have still some interest in mathematical problems and therefore I take this opportunity to send you my deep appreciation of your research work.*
>
> *With kindest regards,*
>
> *Yours sincerely,*
>
> *V. Ambartsumian.*

This project was never carried out, but after a while Ambartsumian received from Wiener a very cordial letter of gratitude…

Wiener's interest in literature was so great that he even tried to write a screenplay. In 1952 he sent a screenplay project to Alfred Hitchcock.

In the covering letter he wrote:

> *I have recently been in Mexico working in a scientific laboratory where I have run into a combination of characters and even of possible situations lending themselves ideally to a suspense and horror movie of the type in which you are expert. Together with my daughter, Miss Peggy Wiener, and an American doctor, Dr. Morris E. Chafetz, we have written a synopsis of the proposed movie. Because we are without experience in screen technique, we have not attempted to go*

> *further and write it up as a scenario. This synopsis has been registered with The Authors' Guild in New York, and we should be delighted to pass it on to you for examination if and when it should be possible. [14].*

We don't know what happened to this screenplay …

3 Aleksey Lyapunov

Aleksey Lyapunov was a descendant of an old noble family which originated numerous outstanding figures of Russian science and culture who lived and worked in the 19th and 20th centuries. Aleksey Lyapunov (1911–1973) began to work in the field of cybernetics in the early 50s. By that time he was already a well-established scientist, famous for his works in the field of descriptive set theory, mathematical statistics and geophysics. His erudition combined with extended scientific interests allowed him to become the leader of a new science. Aleksey Lyapunov did an enormous work trying to understand the fundamentals of cybernetics, to define the subject of this science, to classify its main directions and problems.

In the early 50s, the official ideological authorities of the USSR and many Soviet philosophers condemned cybernetics as a 'bourgeois pseudo-science'. It's quite enough just to mention the article: *'Whom Does Cybernetics Serve?'* that was published in the Issue 5 of the magazine *'Voprosy Filosofii' ('Problems of Philosophy'),* 1953. It was placed in the section *'Critique of Bourgeois Philosophy'*. The author of this libel who hid himself behind the pseudonym *'Materialist'* wrote:

> *"The theory of cybernetics which endeavors to apply the functioning principles of the brand-new computers to diverse natural and social phenomena… is a pure reductionism transformed into idealism. It is a barren flower on the tree of knowledge" [15].*

In 1954, in the 'Short Philosophical Dictionary' ('Kratkii Filosofskii Slovar') cybernetics was defined as

> *"A reactionary pseudo-science that emerged in the USA after the World War II and became widespread in capitalist countries" [16].*

It is easy to imagine what it was like in the Soviet Union to disseminate the *'reactionary pseudo-science'* in those hard times. But Aleksey Lyapunov practiced this dissemination on a regular and professional basis.

Lyapunov realized at once that mathematical theory of control demands a detailed analysis of different control systems, deep research of the processes of emergence, transmission, storage and processing of information in engineering, nature and economics. This could create a new approach that was named 'cybernetic' by Norbert Wiener. Encyclopedic erudition so characteristic of Lyapunov perfectly suited for integration of facts and theories from different fields of natural science with the aim of generation and development of theoretical cybernetics.

Lyapunov never limited himself to mathematics in his scientific work – he was a real encyclopedic scientist who worked diligently and effectively in different branches of science. But in spite of Lyapunov's extended scientific interests he always remained highly professional in his scientific works. Biologists believed him to be a biologist, geophysicists – a geophysicist, philosophers – a philosopher. His great erudition in combination with a holistic, integrated approach to natural science and to the whole complex of scientific knowledge have become a kind of fruitful ground that was just perfect for growing the ideas of cybernetics.

1n 1954–1956, Lyapunov gave lectures on cybernetics to many different audiences – he spoke to mathematicians, engineers, biologists, philosophers and linguists, explaining the essence of cybernetics and its true purpose.

Aleksey Lyapunov possessed wonderful human qualities and remarkable pedagogical abilities. He was a responsive and friendly person; he never limited himself to a concrete subject he was teaching, but always drew the audience's attention to diverse connections between the taught subject and the general system of scientific knowledge and pointed out the unexplored fields of science.

Naturally, it was impossible to organize active promotion and development of the new science without appropriate literature. So, Lyapunov concentrated on creating and publishing of the necessary materials. He achieved a success in publication of the translation into Russian of Wiener's *'Cybernetics'* and other foreign publications, the publication of Igor Poletaev's *'Signal'* [17] (it was the first original book on cybernetics written in our country!), made agreement with *'Fizmatgiz'* to launch his collections *'Problemy Kibernetiki' ('Problems of Cybernetics').* The friends and pupils of Aleksey Lyapunov helped him to fulfill these noble and bold enlightenment projects.

At the beginning of 1954, Anatoly Kitov, who was one of Lyapunov's pupils, prepared a voluminous report on the essence of cybernetics for a seminar held at one of the research institutes. Materials of this report were developed and amplified by Academician Sergey Sobolev and Aleksey Lyapunov, whereupon the paper was published with the title *'The Main Features of Cybernetics'* in the Issue 4 of *'Voprosy Filosofii'*, 1955 [18]. This was the first positive article on cybernetics that received the official approval of Soviet authorities.

But it was not enough for further development of cybernetic research and formation of cybernetics as an integral field of knowledge. It was necessary to create a consolidated team of scientists who would be capable of organizing and conducting cybernetic research programs. The other goals were to get familiar with the process and results of similar studies performed abroad, to formulate the problematic and find effective and unified ways of collaboration for heterogeneous teams of scientists who were ready to work in this field of science. All these goals were achieved in the seminar on cybernetics organized by Lyapunov at the State University of Moscow in 1954. This seminar had become the first coordinating center for cybernetic research in the Soviet Union. It attracted interest across a broad range of specialists (mathematicians, engineers, biologists, philosophers, physicists etc.) and talented young people. Very soon it had become a city-wide and even a nation-wide event. Almost every team formed afterwards with the aim of conducting cybernetic research was somehow connected with this Seminar. It was named a *'Big Lyapunov's'* seminar on cybernetics.

This seminar existed for 10 years (1954–1964). 120 meetings (with the same number of reports and discussions) were conducted during this period of time. Surely it is enough to look at the list of these reports [19] to understand that the research area and the level of reports and lecturers met all requirements of the head of the Seminar and corresponded to the needs of cybernetic science in our country.

Among the goals of the Seminar were explanatory work with the aim of vast dissemination of cybernetic ideas and active struggle against negative attitude towards cybernetics, which was not uncommon in the Soviet press. Over the years, several hundreds of popular lectures and reports concerning cybernetics, its subject-matter, essence and goals were presented by the head of this Seminar Professor Lyapunov and its participants, who severely criticized the unprofessional articles on cybernetics that often appeared in the Soviet magazines and newspapers.

The Seminar of Aleksey Lyapunov became a kind of nucleus of the cybernetic research in the USSR. It had given rise to many local *'small'* cybernetic seminars headed by participants of the *'Big'* Seminar.

This success naturally led to the idea of establishing an Institute of Cybernetics of the USSR Academy of Sciences. Creation of an academic institute for researches in a new progressive field of science was a standard practice in every scientific community.

After the official 'rehabilitation' of cybernetics, numerous academic institutes of this kind were established in different republics of the USSR, such as the Institute of Cybernetics of the Academy of Sciences of Ukraine, the Institute of Engineering Cybernetics of the Academy of Sciences of Belarus. Other Institutes of Cybernetics were established in Georgia, Uzbekistan, Azerbaijan, Estonia etc.

A project to establish such an Institute in Moscow, at the Department of Physical and Mathematical Sciences of the USSR Academy of Sciences was seriously considered during the period of 1959–1961. In the archives numerous documents are preserved concerning this project. A lot of these materials and documents were published in 1998 in the book *'Essays on the History of Computer Science in Russia' (Ocherki Istorii Informatiki v Rossii*) [20].

But why was not this Institute established by that time, when everyone was fascinated by cybernetics?

It is known that there was some disagreement among participants of the consultations concerning the main research directions of the future institute. Most of the participants were eminent scientists already possessing a big academic experience in different fields. Accordingly, they had different views of cybernetics and of the scientific problems the future institute had to deal with.

There were also conflicting opinions concerning the staff structure. Who was able to head the Institute of Cybernetics? The fact that Lyapunov decided to move to Novosibirsk in early 60s may have influenced the situation, too.

With the aim of coordinating research activities, the '*Scientific Council on the Complex Problem of Cybernetics'* was established at the USSR Academy of Sciences in 1959. Academician Aksel Berg, an outstanding scientist and organizer of science, was appointed as the first chairman of the newly-formed Council. Under his chairmanship, the Council became the national center of cybernetic research. Aleksey Lyapunov, Igor Poletaev and other scientists made major contribution to creation and development of this Council.

In 1958, Lyapunov launched his famous series of collections *'Problemy Kibernetiki'* (which is known abroad as *'Systems Analysis'*). He remained the editor of these collections until the end of his life. Under his editorship, 29 issues were published. Altogether, 41 issues of *'Problemy Kibernetiki'* appeared during the period of 1958–1984.

Thus, the situation changed. Due to the heroic efforts made by Lyapunov and his associates, cybernetic science took its deserved place in the Soviet Union. As a matter of fact, exactly at this point began the active development of this science in our country.

Lyapunov contributed a huge amount to formation and development of the national cybernetics. His merits are very great indeed, and the first 10–12 years of the Russian cybernetics is fairly called *'The Lyapunov Period'*.

The work performed by Aleksey Lyapunov obtained international recognition. In 1996, the IEEE Computer Society honored Lyapunov (posthumously) as a *'Computer Pioneer'* who *'Founded Soviet Cybernetics and Programming'*.

Here are some reminiscences of Lyapunov's contemporaries:

Julius Schreider:

> *What enabled cybernetics to unite people who were completely different? Why did the Lyapunov's seminars become the main center consolidating people of different professions and scientific views?*
>
> *What exactly happened in the early period of cybernetics? I suppose, it was the process of consolidation of scientists who were united by cybernetics as a field of scientific activity allowing to detect natural origins in the world of organizations up to the intellect itself. The idea to find a rational explanation of how the human intellect works seemed to be just fascinating… [21].*

Modest Haase-Rapoport:

> *Aleksey Andreevich devoted his whole life to science and served his country unselfish. The scope of his scientific interest was so vast that he can truly be called a person of encyclopedic knowledge.*
>
> *His attitude to cybernetics resembled the attitude of a priest to his religion. It was sacred to him. Understanding living organisms by means of natural sciences is a goal of utmost significance… It was definitely very important to him [22].*

In November 1971, Vladimir Uspensky wrote in his birthday greeting to Aleksey Andreevich:

> *Over many years I admiringly observed you working. I consider you to be one of the heroic figures of Russian science. For the younger generation, it's hard to believe how much fearlessness, persistence and commitment to principles one needs to give birth to a new science, cybernetics [23].*

At the beginning of 1962, Aleksey Lyapunov moved to Novosibirsk upon invitation of the administration of the Siberian Division of the USSR Academy of Sciences and remained there for the rest of his life. While living in Novosibirsk *'Akademgorodok'* (the *Novosibirsk Academic Village*), he worked in the field of theoretical and applied cybernetics with all his inherent energy and great brilliance. It would hardly be an exaggeration to say that when Lyapunov moved to Novosibirsk, the main center of Soviet cybernetic research moved here along with him.

There, he took part in forming of scientific teams. He played an important role in creating of the Division of Cybernetics at the Institute of Mathematics. He organized the Department of Mathematical Analysis and the Department of Theoretical Cybernetics in Novosibirsk State University.

Lyapunov was an outstanding teacher and a remarkable disseminator of scientific knowledge. He was engaged in the teaching process at all stages of education, from the university down to the primary school level. He never limited himself to teaching mathematics. His interests extended to the whole range of natural sciences. He also occupied himself with the problems of education and upbringing.

Lyapunov's pedagogic activity reached its highest point at the Novosibirsk Academic Village. He took part in the organization of Siberian Mathematical Olympiads. Together with Academician Lavrentiev, he initiated the establishment of the famous Physico-Mathematical School (PMS) at the Novosibirsk University. It was the first school of this type in our country. Lyapunov had a great influence on the process of formation and development of this new type of school.

Lyapunov's work in the PMS left a general imprint on school education. It was not just a well-organized process of teaching mathematics. It was also living example of selfless devotion to science and society.

Lyapunov was a remarkably charismatic person, loved by successive waves of pupils and students. His sociability and friendliness, his breadth of interest and generosity of scientific communication were valued by people working in different siences. In the 60s and in the early 70s he was regularly visited by mathematicians and physicists, biologists and linguists who came from Moscow and from other cities of the country. The Academic Village hosted many national and international conferences on cybernetics and programming.

In 1973, Igor Poletaev wrote:

> *The scientific truth was his sacrificial altar, and the search for the truth was his religion. His attitude to cybernetics resembled the attitude of a priest to his religion. His self-denying and knightly serving the truth was supplemented with his fascinating personality and his ability to be precise and understandable at the same time. … Even disputable opinions sounded attractive and almost convincing, when he was talking. Every conversation with him was an intellectual event and an aesthetic experience [24].*

Today's reader will surely feel that Aleksey Andreevich Lyapunov was not only an outstanding mathematician. He will remain in history as a remarkable humanist possessing precious human qualities.

4 Leonid Kantorovich

The name of Academician Leonid Kantorovich (1912–1986), his life, work and struggle for his principles hold a special place in the history of the 20th century.

The early manifestation of his talent, mathematical discoveries and formulation (at the age of 27!) of the scientific methods of control over economics and planning, an extraordinary breadth of interests, uncompromising of a fighter on one hand, modesty and nobleness on the other. All this forms the unique phenomenon of Kantorovich.

The mathematical discoveries of Kantorovich have become the basis of major new directions in mathematics. He is fairly considered to be one of the most striking mathematicians of the last century. Already in the 1930s, he made a big contribution to theoretical mathematics. Functional analysis holds a special place in the mathematic works of Kantorovich. With his classical works, Kantorovich turned functional analysis into the natural language of computational mathematics.

At the same time, he believed economical cybernetics to be the work of all his life. The linear programming discovered by Kantorovich constitutes the nucleus of economical cybernetics. This is a new and very significant conception. It allows turning economics into an objective science which leads to the most effective results of economic activity.

One of the closest associates of Kantorovich, Academician Valery Makarov, wrote:

> *He achieved high-class results in functional analysis, function theory, computational mathematics. He has written a number of major works on set theory, theory of computer programming etc. He has written (alone and in joint authorship) about ten considerable monographs on mathematics. One would think it is quite clear that Leonid Kantorovich is a mathematician to the core … As a matter of fact, it is not the whole truth. His phenomenon is characterized by the fact that at the same time he is an outstanding economist who has fundamentally changed the economic thinking and the understanding of economic events; he has become the founder of a new economic school [25].*

Another pupil and adherent of Kantorovich, Professor Simon Kutateladze observed:

> *In the 20th century, two eminent mathematicians, John von Neumann and Leonid Kantorovich, devoted themselves to the economic subject matter. The first one developed game theory as an instrument of study of economic behavior, the second one elaborated linear programming as an instrument of taking optimal decisions about optimal use of limited resources. The significance of the studies performed by von Neumann and Kantorovich goes far beyond the borders of their outstanding technical results. Their achievements have clearly shown that the modern mathematics possesses large-scale possibilities to apply economic analysis to practical problems. The distance between economics and mathematics became shorter. Economics remains humanity, but it becomes mathematized…[26].*

Academician Israel Gelfand said:

> *Why do I consider Leonid Kantoorovich to be a genius? It is very simple – he combines in himself two cultures: the first one pertaining to humanities and the second one pertaining to mathematics… In the 20th century, very few people were capable of such a synthesis… What we see is an integral inner spirituality that equally affects all areas of his work [27].*

Leonid Kantorovich was a child prodigy. He was good at counting already at the age of two. As a child he received a special scholarship that was established for talented children.

In 1926, at the age of 14, Leonid Kantorovich was admitted to the Physic-mathematical Faculty of the University of St. Petersburg. There were not many students at the faculty, but among the classmates of Leonid Kantorovich were some with great talent: the future Academicians S. Sobolev and S. Khristianovich, corresponding member D. Fadeev, Professor S. Mikhlin. The latter wrote in his memoirs about the young Kantorovich:

> *The first impression I got of him was that he is a shortish boy… a childish boy with rosy cheeks. What was he doing here, I wondered, such a little boy at the University? I was almost 19 and considered myself to be an adult man, and he wasn't even 15 yet… He was exceptionally talented, you could feel it at once. I remember how astonished we were when his first works had been published, less than a year after he was admitted to the University. My classmates and I were really shocked. We were in the third year at the University, and he was in the second, but the idea of publishing student's works seemed to us unrealizable [28].*

His first research, Leonid conducted under the direction of Gregory Fikhtenholts, who was a prominent mathematician and a Professor at the University of St. Petersburg (Leningrad). Already as a student Kantorovich became famous for his works on the descriptive theory of functions. At the age of 18, he successfully graduated from the University. Professor Fikhtenholts wrote about his young graduate:

> *During a period of 4 years we were in regular communication with each other, so I can certainly state that this young man (now at the age of 18) possesses an extraordinary mathematical talent [29].*

After graduating from the University, Leonid Kantorovich pursued postgraduate studies. He also began his teaching career at the University and at the newly opened Leningrad Institute of Engineers of Industrial Construction, where he became a Professor two years later. The Department of Mechanical Engineering at this Institute was of a very high level. The problems of designing large objects like turbines, airplanes and complex constructions raised his interest. He invented a range of new computational approaches, and in 1936 he published his work (written in joint authorship with V. Krylov) '*Methods for the Approximate Solution of Partial Differential Equations*'. This monograph was the starting point of computational mathematics as an autonomous science.

Leonid Kantorovich is the author of more than 300 scientific studies which are characterized by a strikingly large scope of research directions. They are united not only by his personality, but also by the integrity of his work and the interpenetration of methods and ideas used in the process of resolving the most different problems.

The same inner integrity of Kantorovich determined his life and civic engagement. He was honest and daring both in science and in life. *"It is the scientist's duty and right to tell the truth"* said Kantorovich in his last interview. It is not just a declaration, but a moral attitude confirmed by his whole life and his actions as a scientist and a citizen. As a scientist, Kantorovich did not differentiate among fundamental problems or applied problems; problems related to mathematics or to humanities; problems or not prestigious. He devoted himself to any challenging problem with equal interest and endeavor and proved by his works that a real scientist could uncover the true depth of a minor problem that would bring it to the highest level of significance. As a citizen, he defended the truth irrespective of the predominant opinions and fearlessly struggled for his ideas for humanity's sake.

The efficiency of Kantorovich's approach showed up most clearly in the case of the 'plywood trust'.

In 1938, Kantorovich received a request from the employees of a Central Laboratory of the Leningrad plywood trust. They asked him to recommend a numerical method of calculating a rational plan of machine utilization. It was necessary to organize a complex of five technological processes involving five peelers of eight types. The machines had different productivity, so it seemed to be impossible to calculate the outcome which presumably depended only on casual circumstances.

To solve this problem, some totally new ideas were needed. The central part of the Kantorovich's invention is constituted by optimality criteria. The new method allowed proposing different schemes of examination of options.

A little booklet with the title *'Mathematical Methods in the Organization and Planning of Production'* appeared in 1939. In this study he developed mathematical formulation of the industrial optimization problems, effective techniques of problem solving and methods of economic analysis of these problems. Thus, Leonid Kantorovich formulated the mathematical technique known as *linear programming*, which had a great influence on the development of economic science.

On the basis of this study, Kantorovich created a whole number of further works. Years after, in 1975, these works have brought him a Nobel Prize in Economics.

Unfortunately, the invention of Kantorovich did not receive recognition in the USSR. Application of mathematical methods to economics (which belongs to humanities) was received with hostility by the traditional economists and official philosophers: they considered it a breach of the purity of Marxist theory that condemned mathematical economics (together with genetics and cybernetics) as a 'bourgeois pseudo-science'.

During the war period, Kantorovich engaged more and more in economics. He created the first version of a big book titled '*The Best Use of Economic Resources*', where he formulated the design principles for the systems of economic indicators stimulating fulfillment of the set tasks. This book was not published until 1959. For over 20 years he had been waiting for publication of his papers in the field of economics that made him world-famous!

Overcoming the Iron Curtain and other limitations, Western scientists gradually learn more about the discoveries made by Kantorovich. Having studied his first publications, they unconditionally acknowledged his priority. They've even made great efforts to translate his pioneer work into English and to get it published.

The first-rate American authority in the field of linear programming G. Dantzig wrote that Kantorovich's paper dated 1939 contains nearly all applications that have become known in 1969. Subsequently, a similar technique has been created independently of Kantorovich by the American economist Tjalling Koopmans and other scientists.

In 1975, the Nobel Committee decided to award the Nobel Prize in Economics jointly to Leonid Kantorovich and Tjalling Koopmans *"for their contributions to the theory of optimal use of resources"*. The outstanding achievements of the scientists from the two countries have been recognized and appreciated.

There exists an opinion that the scientific life of Leonid Kantorovich was quite trouble-free. Indeed, he was a child prodigy, admitted to the University at the age of 14. At the moment of graduation, he already was a prominent mathematician and the author of about 10 papers; he became a Professor at the age of 20, a Doctor of Sciences at the age of 23 (he was given this degree without thesis defense). At 26 years he was the head of a scientific school, the winner of National Contest for Young Scientists.

But if you look closer at the facts of his biography, you will see that it was not so rosy.

Archive materials shed new light upon the events related to his struggle for recognition and application of the scientific techniques of the organization and control of economic activity.

Leonid Kantorovich constantly received invitations to different international conferences on computational mathematics, operations research, mathematical economics etc. Being a member of Program Committees of different international congresses, and an honorary member of several international scientific societies, he had no possibility to attend these events because he was not permitted to travel abroad. It was called in the Soviet Union *'nevyezdnoi' ('non-exit')!*

In his Biography Card filled out by Kantorovich in February, 1986 are specified the Universities having awarded him an honorary doctorate (*Honoris Causa*):

The University of Glasgow (1966, Great Britain)
The University of Grenoble (1967, France)
The Warsaw School of Economics) (1967, Poland)
The University of Nice (1968, France)
The University of Munich (1970, FRG)
The University of Helsinki (1971, Finland)
Yale University (1971, USA)
The University of Paris 1 Pantheon-Sorbonne (1975, France)
The University of Cambridge (1976, Great Britain)
The University of Pennsylvania (1976, USA)
Indian Statistical Institute (ISI), Calcutta (1977, India)
The University of Economics, Prague (1981, Czechoslovakia)
Martin Luther University, Halle-Wittenberg (1984, GDR)

In the private archives of Kantorovich there are many documents concerning his visits abroad.

After Khrushchev's so-called *'thaw'*, he was sometimes allowed to go beyond the *'Iron Curtain'*. In June of 1966 he received permission to go to Great Britain, to attend his award ceremony – The University of Glasgow had awarded him the degree of Doctor of Laws. In 1967, he was sent to France where he was awarded the degree of Doctor of Laws by the University of Grenoble.

Thereupon the Academician Kantorovich had no possibility to participate even in the most significant scientific forums that took place abroad. He was prevented by the authorities from travelling to other countries to accept the awards that not only the scientist but the Soviet Union itself could be proud of!

Thus, Leonid Kantorovich was named in absentia as the recipient of the honorary degrees in Nice (1968), Munich (1970), Helsinki (1971) and even in Paris (Sorbonne, 1975). The award ceremony at Yale University did not take place – according to the University Charter, the degree couldn't be awarded in absentia.

In those days, the receipt and transfer of correspondence for the Soviet scientists was carried out through the Foreign Department of the Presidium of the Academy of Sciences. When a letter of invitation came – even the most honorary one – it was sufficient just to store the letter in some office of the Presidium and give it to the addressee after the expiry of the set term. It was also possible to conceal the receipt of such letter.

So, in 1971 the Foreign Department simply did not inform Leonid Kantorovich that he was awarded an honorary doctorate by the University of Helsinki. Actually, Kantorovich recognized it by chance, when he received (to his home address) a letter from a Finnish tailor (!) asking him about his clothing sizes, in order to make a ceremonial mantle and a hat for the award ceremony.

In 1976, after a long period of writing letters and procrastination, Kantorovich was at least allowed to go to Cambridge personally to receive an honorary doctorate.

Here, I would like to add something else concerning this high award.

Later, on July 6, 1983, there was a short item in the newspaper titled *'Vechernaya Moskva' ('Evening Moscow'),* placed in the section *'Information Desk',* which said:

> ***Question:*** *What scientists from Russia and the Soviet Union were awarded honorary membership of the Cambridge University?*
>
> ***Answer:*** *The honorary doctorate of the Cambridge University (Great Britain) was awarded to K.A. Timiryazev in 1909. Along with Timiryazev, this high degree was awarded to the biologist I.I. Mechnikov (1891), the chemist D.I. Mendeleev (1894), the physiologist I.P. Pavlov (1912), the biochemist V.A. Engelgardt (1970), the mathematician and economist L.V. Kantorovich (1976).*
>
> *The physicists P.L. Kapitsa (1925), L.D. Landau and I.M. Lifshitz (1962) were awarded honorary membership of Trinity College, Cambridge.*
>
> *This information was provided by the Institute of History of Natural Science and Technology of the Academy of Sciences of the USSR [30].*

It seems to me that the above short item, which maybe remained unnoticed by the historians of science, is a real condemnation to the Soviet Academy of Sciences and to the whole Soviet system that has, for many decades (!), kept the great scientists like Leonid Kantorovich on the black list of persons who were not permitted to travel abroad…

In 1957, a government decision was made to establish a big new scientific center in the eastern part of the Soviet Union, the Siberian Branch of the USSR Academy of Sciences. Leonid Kantorovich was one of the first scientists who were invited to work in the Siberian Branch. In 1960, a team of Leningrad mathematicians headed by

Kantorovich moved to Novosibirsk and joined the Institute of Mathematics of the Siberian Branch as the Department of Mathematical Economics.

During the period of 1960–1970 Kantorovich was the deputy director of the Novosibirsk Institute of Mathematics and the Dean of the Department of Computational Mathematics of Novosibirsk State University. Since the founding of the '*Siberian Mathematical Journal*' and until the last days of his life, Kantorovich remained a member of the Editorial Board determining the scientific image of the Journal in applied functional analysis and mathematical economics. In 1958, he was appointed a Corresponding member of the Department of Economics of USSR Academy of Science, and in 1964 – a Member of the Department of Mathematics.

The Department of Mathematical Economics created by Kantorovich in Siberia was the first one to use a complex approach to the problems of application of mathematical methods in economics.

During this period, Kantorovich was engaged in the scientific organizational work. The conduction of many several national and international conferences and meetings on the problems of application of mathematical methods in economics was entirely due to his initiative. He organized the training of specialists in the area of economical cybernetics.

In 1971, Leonid Kantorovich moved to Moscow where he was appointed the head of the Department of System Modelling of scientific and technological advancements at the Scientific Research Institute for System Studies of the USSR Academy of Sciences.

Research work on computer architecture and programming hold a special place in the intellectual legacy of Kantorovich. He was one of the first scientists to understand the great significance of computers and informatics. During the period of 1953–1956, Leonid Kantorovich was engaged in the popularization of computer science.

In his articles Kantorovich wrote:

> *In the history of cultural and technological development of human society a special place is held by the discoveries and inventions having affected not just one but nearly all the fields of the human activity. Among these discoveries and inventions of major importance we can name book-printing, discovery of the New World, steam engine, weaver's loom, railway, electricity, bacteriology, radio, aeronautics, television and, finally, atomic energy [31].*

General ideas of Leonid Kantorovich concerning the complex development of machine mathematics (methods, algorithms, programming, computer architecture) also deserve consideration.

The scientific school of Professor Kantorovich, be it in the area of mathematics or economics, is not just tens of his immediate pupils. There are lots of his followers whose scientific thinking and activities were greatly affected by the works of Kantorovich and by personal communication with this outstanding scientist.

Leonid Vitaliyevich set for his pupils and followers a brilliant example of honesty, uncompromising and firmness in science, of objectivity and friendliness. Throughout his life, Kantorovich attracted and drew about him people by force of his fascinating personality and wealth of ideas, remarkable kindness, modesty and sociability, his human quality. It was a pleasure for him to work with the younger generation, and he was always surrounded by young people.

Many authors try to compare the life and work of the two outstanding mathematicians of the 20^{th} century – Leonid Kantorovich and John von Neumann. There is really some similarity between the figure of Kantorovich who was characterized by extended scientific interests and made a great change not only in mathematics and mathematical economics, but also in computer science, and the figure of von Neumann.

5 Andrey Ershov

We have already mentioned the fact that, in the beginning of the 1960s, when many of the leading Soviet mathematicians moved to Siberia, the Novosibirsk Academic Village became, in essence, the national center of cybernetic science.

Andrey Petrovich Ershov (1931–1988) graduated from the State University of Moscow in 1954. This was the first group of graduates in the specialty 'Programming'. Then, Ershov pursued postgraduate study under the direction of Aleksey Lyapunov.

Subsequently, Ershov wrote:

> *From the very first lectures Aleksey Andreevich completely captured the minds of the students. He was an ideal guide of new ideas. The magnetic influence of his bright appearance and an exceptional eloquence, his selfless enthusiasm and merry passion, his complete availability to the students without any familiarity – all this at once made him the most popular lecturer… [32].*

Already in the late 50s, due to his brilliant studies Ershov became one of the leading Soviet programmers. After the establishment of the Siberian Branch, Ershov received an invitation from the Director of the Novosibirsk Institute of Mathematics, Academician Sergey Sobolev. He was proposed to organize and head *the Department of Programming* of this Institute. In 1960, Ershov moved to Novosibirsk.

Many authors have written about Andrey Ershov. It suffices to name the interesting voluminous book [33] containing reminiscences and photographs, that appeared in Novosibirsk in 2006. In this book and in many other publications you can find detailed information about Ershov's life, his personal achievements and the achievements of his team, concerning theoretical and applied works in informatics. Surely, programming holds one of the central places in Computer Science. And it happened so, that at the beginning of this new branch of science stood such a talented, energetic and honest person as Andrey Ershov.

One of the closest friends and associates of Ershov, Svyatoslav Lavrov, said about him:

> *Andrey Petrovich was the leader of the native programming. Any of his undertakings, even if it didn't seem significant at the beginning, would draw the attention of many other programmers and set them in motion. A wide range of his popular science lectures found a broad response among programmers from the whole world [34].*

Indeed, Andrey Ershov made a considerable contribution to the organization of international cooperation. This side of his activity had a great and productive influence on the development of programming science in general and the recognition of the achievements made by the Soviet scientists.

It appears that Andrey Ershov, as opposed to Leonid Kantorovich, was allowed to travel abroad. Below, we will site a few opinions of Ershov's contemporaries about his international contacts:

Boris Trakhtenbrot:

Andrey Petrovich maintained friendly relations with most of the outstanding programmer scientists and collaborated with the leading scientific centers from the whole world. It happened by the time when a trip abroad, especially to the 'capitalist countries', was a rare privilege in the Soviet Union. Fortunately, Andrey Petrovich felt very obliged to have this privilege and did anything possible to relieve the isolation of those who were… restricted in their contacts with their foreign colleagues [35].

Andrey Bers:

Andrey Petrovich thoroughly organized and maintained the relations with numerous foreign and native colleagues and groups of scientists. He received an immense quantity of letters from the whole world and punctually registered all materials, incoming and outgoing letters. On the base of these papers he created a unique data library containing reports, preprints and publications.

Many foreign colleagues came to the Academic Village to visit Andrey Petrovich, and we had the possibility not just to see and hear them, but (again, due to Andrey Petrovich's efforts) also to discuss our results and to talk, along with Ershov, to the most eminent programmer scientists of the world: John McCarthy, Donald Knuth, Edsger Dijkstra, Tony Hoare, Aad van Wijngaarden and many others [36].

Heinz Zemanek:

Andrey Ershov was a prominent and highly distinguished person in the international community of specialists in the area of information processing…

My American colleagues often spoke of Andrey with respect …

After the establishment of IFIP in 1962, Andrey was appointed a representative of the USSR Academy of Science in the Technical Committee 2 (TC 2).

He endeavored to organize the participation of the USSR in IFIP, and used every possibility to present the works of his pupils and colleagues and to enable them to take part in the events conducted by IFIP. We all know that it was a hard task [37].

In 1974 Ershov was made a Distinguished Fellow of the British Computer Society. This grade is awarded for *'Outstanding Contributions to the Advancement of Computing'*. Earlier than Ershov, only five specialists (among them, two English scientists) were awarded this honorable grade: Maurice Wilkes, Edsger Dijkstra, Grace Hopper, Christopher Strachey and James Wilkinson.

In my opinion, the major achievement of Andrey Ershov's activity in international scientific co-operation was the Symposium *'Algorithms in Modern Mathematics and its Applications'* which was dedicated to Al-Khorezmi, the outstanding ninth-century mathematician who gave his name to the word *'algorithm'*. This event took place in September of 1979, in the city of Urgench, Uzbekistan. Andrey Ershov and Donald Knuth acted as co-chairmen of the Symposium.

In the private Ershov's archives *hundreds* of letters are preserved concerning the preparation and conduct of this Symposium. Let us cite some extracts from these papers:

In his letter to Donald Knuth of April 7th, 1977, Ershov is planning:

> *Concerning the scientific pilgrimage to the place of birth and youth of Al-Khorezmi.*
>
> *I propose you to consider the following idea. The conducting of the colloquium with the title 'Algorithm in Modern Mathematics' in Khorezm or somewhere else in the region, beginning from Tashkent. Scientific organizers: D. Knuth and A. Ershov. Promoter: the Uzbek SSR Academy of Sciences. Number of participants: 10 persons from foreign countries and 10 persons from the USSR. Timeframe: May of 1978. Naturally, any counter-offers are welcome [38].*

At first, Donald Knuth and Andrey Ershov prepared and coordinated the number and the membership of participants. It included the most eminent scientists from the Soviet Union and other countries who were engaged in computational science.

So, Ershov sent invitation letters to the participants and received letters of response. Mostly, the invitees expressed their gratitude and informed Ershov about their intention to come to Urgench.

At the same time, Ershov realized engaged in correspondence with the academic authorities: he had to obtain entry permits for the foreign guests.

Moreover, he managed to obtain for some of them the status of *'a guest of the Academy of Sciences'*, which facilitates their financial problems.

The Urgench Symposium was the first scientific attempt to gather all material on the subject of Al-Khorezmi and his works. The Austrian scientist Heinz Zemanek played a special role in the process of preparation and conducting of the Symposium. Zemanek was quite interested in the history of science, and in particular, in the mathematics of the ancient Middle East.

The Urgench Symposium took place on September 16–22, 1979. It gathered 39 participants, among them 26 scientists from the USSR and 13 from other countries. The first speaker at the first morning meeting (September 17) was H. Zemanek. He delivered a great and remarkable report entitled *'Al-Khorezmi: his Background, his Personality, his Work and his Influence'*.

After returning to Wienna, Zemanek wrote:

> *From the international point of view, the Urgench Symposium 1979 has become one of the most impressive scientific events. In collaboration with Donald Knuth, Andrey Ershov, acting in the name of the USSR Academy of Sciences, organized the Symposium 'Algorithms in Modern Mathematics and its Applications' that gathered many outstanding participants, such as S. Kleene, F. Bauer, G. Kaufmann and A. van*

Wijngaarden. It was a crucial experience for the development of the theory of algorithms [39].

Donald Knuth said:

… It was an unforgettable event. Instead of an 'ordinary' conference where the participants just read out their ready reports, we conducted a whole number of discussions on fundamental problems of mathematics and computational science. The desert landscape surrounding the Khorezm Oasis and the breath of history distracted us from the flood of thoughts going through our minds in the circumstances of our daily lives. It helped us to concentrate on profound ideas and philosophical thoughts turning to the future. …

This Symposium – a scientific pilgrimage to Khorezm, to the birthplace of the notion 'algorithm' – has become a dream-come-true for me. Although both of us were on the list of co-chairman, Andrey carried out about 99% of the work. So I had the possibility to relax and just enjoy what was going on around me, learning many important things from the people I met there. It really was a once-in-a-lifetime experience, and I hope that many other specialists working in computational science will be able to participate in events like this one, provided that Andrey's example inspires someone else.

During this week I got to know him much better, and was especially surprised at how brilliantly he performed his manifold duties: Head of the Conference, organizer, philosopher, public speaker, interpreter and editor [40].

In the early 1980s, one of the participants of the Symposium, the English scientist M. Paterson, who was the President of the European Computer Society, published in the Bulletin of this Society a report on his journey to Urgench. In particular, he said:

This informal Symposium offered its participants exceptional opportunities to establish new academic and social contacts and to renew the old ones. It is particularly important when we need to overcome the linguistic, political and geographical gaps which so often hinder the development of our science.
I hope we shall all retain the spirit of community and collaboration which enveloped us during this delightful and unforgettable week. [41].

6 Conclusion

We have described in brief outline the biography of Norbert Wiener, the scientist who introduced in our scientific and everyday use the proud word *Cybernetics.* Special attention was given to Wiener's visit to Moscow in the Summer of 1960, at the very beginning of the 'cybernetic era'.

We have traced the main events of the lives and activities of prominent Russian scientists Aleksey Lyapunov, Leonid Kantorovich, and Andrey Ershov. They lived and worked in the middle of the last century in Novosibirsk Scientific Center.

Evidently, they possessed, as well as a number of their disciples, remarkable human qualities.

Unfortunately, we live in difficult times. Financing of science and education is gradually reduced. The mass media are used for systematic declining of the mental and moral level of the public. The honest people attempt to resist these unfavorable trends. In our opinion, the main weapon on this difficult road is *education.*

Contemporary young people typically are not familiar with the remarkable scientists of the past. We hope that the presented materials in this article's narratives will help the young readers to choose a worthy way of their life.

References

1. Peccei, A.: The Human Quality. Pergamon Press, Oxford (1977)
2. Wiener, N.: Ex-Prodigy, My Childhood and Youth. Simon & Schuster, New York (1953)
3. Wiener, N.: I Am a Mathematician. The Later Life of a Prodigy. Doubleday, Garden City-New York (1956)
4. Wiener, N.: Cybernetics or Control and Communication in the Animal and the Machine, 2nd edn. MIT Press and John Wiley, New York-London (1961)
5. Cybernetics. The Expectations and the Results. Politechnic Readings (2), p. 64. Znanie (Knowledge), Moscow (2002) (in Russian)
6. Wiener, N.: Cybernetics or Control and Communication in the Animal and the Machine, 2nd edn. Sovietskoe Radio (Soviet Radio), Moscow (1968) (in Russian)
7. Cybernetics and the Literature. Literaturnaya Gazeta (Literary Newspaper), Moscow (June 30, 1960) (in Russian)
8. Wiener, N.: The Tempter. Random House, New York (1959)
9. Wiener, N.: The Human Use of Human Beings. Mifflin, Boston (1950)
10. Wiener, N.: God, Golem, Inc., a Comment on Certain Points Where Cybernetics Impinges the Religion. MIT Press, Cambridge (1964)
11. Wiener, N.: Invention: The Care and Feeding of Ideas. MIT Press, Cambridge (1993)
12. Masani, P.R.: Norbert Wiener. 1894–1964. Vita Mathematica, vol. 5. Birkhauser, Basel (1990)
13. Masani, P.R.: Norbert Wiener. 1894–1964. Vita Mathematica, vol. 5, p. 337. Birkhauser, Basel (1990)
14. Masani, P.R.: Norbert Wiener. 1894–1964. Vita Mathematica, vol. 5, p. 339. Birkhauser, Basel (1990)
15. Voprosy Filosofii (Problems of Philosophy), vol. (5), pp. 210–219 (1953) (in Russian)
16. Kratkiy Filosofskiy Slovar (The Brief Philosophical Dictionary). Gospolitizdat (State Polit. Publ.), Moscow. 4 edn., pp. 236–237 (1954) (in Russian)
17. Poletaev, I.A.: The Signal. Sovietskoe Radio (Soviet Radio), Moscow (1958) (in Russian)
18. Sobolev, S.L., Kitov, A.I., Lyapunov, A.A.: The Fundamental Features of Cybernetics. In: Voprosy Filosofii (Problems of Philosophy), vol. (4), pp. 136–148 (1955) (in Russian)
19. Haase-Rapoport, M.G.: The Emergence of Cybernetics in the USSR. In: Essays on the History of Informatics in Russia, pp. 245–249. Publishing Center of SB RAS, Novosibirsk (1998) (in Russian)
20. Essays on the History of Informatics in Russia. Publishing Center of SB RAS, Novosibirsk (1998) (in Russian)

21. Shreider, J.A.: A.A. Lyapunov as the Leader of Scientific Movement. In: Essays on the History of Informatics in Russia, p. 198. Publishing Center of SB RAS, Novosibirsk (1998) (in Russian)
22. Fet, Y.I.: Stories About Cybernetics, p. 39. Publishing Center of SB RAS, Novosibirsk (2007) (in Russian)
23. Lyapunova, N.A., Fet, Y.I. (eds.): Aleksey Andreevich Lyapunov, p. 497. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2001) (in Russian)
24. Fet, Y.I.: Stories About Cybernetics, p. 40. Publishing Center of SB RAS, Novosibirsk (2007) (in Russian)
25. Fet, Y.I.: Stories About Cybernetics, pp. 94–95. Publishing Center of SB RAS, Novosibirsk (2007) (in Russian)
26. Kutateladze, S.S.: Kantorovich, and the Mathematization of Science. In: Science and Its People, p. 59. Russian Academy of Sciences, Vladikavkas (2010) (in Russian)
27. Gelfand, I.M.: Leonid Kantorovich and the Synthesis of Two Cultures. In: Leonid Vital'evich Kantorovich: A Man and a Scientist, vol. 1, pp. 161–162. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2002) (in Russian)
28. Mikhlin, S.G.: Reminiscences. In: Leonid Vital'evich Kantorovich: A Man and a Scientist, vol. 1, p. 192. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2002) (in Russian)
29. Fikhtengolts, G.M.: A Review of the Articles by L.V. Kantorovich. In: Leonid Vital'evich Kantorovich: A Man and a Scientist, vol. 1, p. 259. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2002) (in Russian)
30. Information Desk: Vechernaya Moskva (Evening Moscow). Moscow (July 6, 1983) (in Russian)
31. Kantorovich, L.V.: Significance of Modern Computers in Human Culture. In: Leonid Vital'evich Kantorovich: A Man and a Scientist, vol. 2, p. 21. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2004) (in Russian)
32. Ershov, A.P.: Aleksey Andreevich Lyapunov. In: Lyapunova, N.A., Fet, Y.I. (eds.) Memory of Aleksey Andreevich Lyapunov, p. 82. 'GEO' Affiliate of the SB RAS Publ. House, Novosibirsk (2001) (in Russian)
33. Andrey Petrovich Ershov – The Scientist, and the Man, 504 p. SB RAS Publ. House, Novosibirsk (2006) (in Russian)
34. Programmirovanie (Programming). Nauka (Science), Moscow, vol. 1, p. 117 (1990) (in Russian)
35. Trakhtenbrot, B.A.: Memories of Academician Ershov. In: Andrey Petrovich Ershov – The Scientist, and the Man, p. 349. SB RAS Publ. House, Novosibirsk (2006) (in Russian)
36. Programmirovanie (Programming). Nauka (Science), Moscow, vol. 1, p. 105 (1990) (in Russian)
37. Programmirovanie (Programming). Nauka (Science), Moscow, vol. 1, pp. 110–111 (1990) (in Russian)
38. Fet, Y.I.: Stories About Cybernetics, p. 118. Publishing Center of SB RAS, Novosibirsk (2007) (in Russian)
39. Programmirovanie (Programming). Nauka (Science), Moscow, vol. 1, p. 110 (1990) (in Russian)
40. Fet, Y.I.: Stories About Cybernetics, p. 127. Publishing Center of SB RAS, Novosibirsk (2007) (in Russian)
41. Paterson, M.S.: International Meeting of Scientists at Khoresm. In: International Forum of Information and Documentation, vol. 5(1), p. 37. VINITI, Moscow (1980)

Hungarian Scientists in Information Technology

Győző Kovács

John von Neumann Computer Society, Hungary

Abstract. Studying Information Technology, the History of Science and Technology was very rich in Hungarian talents; those who designed 'clever' machines at the very early times of calculators. These calculators are the ancestors of the present-time ones that were called later on, in the 20^{th} century, computers. The computer historians may agree or disagree, but I think the first real-life, early 'calculator-like' machine was developed by *Farkas Kempelen* in the 18^{th} century. It was a real output device, a talking machine. Its input was an organ, a music instrument keyboard and the operator of the machine could enter the text and the output of the operation was a human-like speech.

I start the row of the Hungarian inventors with Kempelen and I finish it in the 20^{th} century with a talented mechanical engineer: *Marcell Jánosi*, who designed and patented the world's first floppy disk. Among the thirteen Hungarian inventors are engineers, mathematicians, priests etc. all developed machines for the information technology.

Keywords: History of science and technology, Hungarian inventors, fist cassette floppy disk.

Introduction

I am an electronic engineer and I was a member of the team that built the first Hungarian computer, the M-3 (see section 8) therefore I always had a bias in favour of hardware stories and not that of software.

Fig. 0.1. Dr hc. Győző Kovács (- when I was 70 years old)

Let me add one note yet. I was very fortune in my life, because I knew the majority of these famous persons. Of course, unfortunately, I did not know the inventors, who were living in previous centuries, such as e.g. Kempelen and Jedlik. I also did not meet István Juhász, but we were living at the same age. Neither have I met John von Neumann, as he died in the year I started my career in computer science. I am proud to say that I was a friend of many other fantastic specialists of the computer technology, whose stories are described in my current essay. I am happy that I have been living at the same age and I could meet them.

Finally, I would like to express my thanks to *Gábor Inokai*, who was my colleague and of course is my friend, for scrutinising my English manuscript. Many thanks again.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 289–319, 2012.

1 Farkas (Wolfgang) Kempelen (1734-1804)

He was a great Hungarian scientist, but he is honoured in two other countries too. The Slovaks call him as a great Slovakian scientist. He was born in Pozsony, in the Hungarian capital at the time. The city's present name is Bratislava and it is now the capital of Slovakia. The Austrians call him a great Austrian scientist, because he served in the emperor's court of Maria Teresia in Vienna. Originally his father, Engelbert, was an Irish citizen and he immigrated to Hungary (Fig 1.1).

Fig. 1.1. Farkas Kempelen, self-portrait

He was educated in Hungary, Pozsony, Győr and Vienna then he began to work for the Hungarian Queen and Austrian Empress Maria Teresia as a mechanical engineer in 1755.

His first construction was a special water pump to solve the water supply of the castle in Pozsony. Then the Empress appointed him the commissioner of the government in South Hungary (Bánát).
He visited in 1769 the Empress in Vienna and she invited him to attend the miracle presentation of magnetism by a French physicist, Pelletier. Kempelen understood the 'miracles' and he gave a plausible explanation to the Empress. He also gave her a short note offering another, more interesting presentation if required. The Empress, who loved such shows, asked him to prepare and present it to the Emperor's court as soon as possible.

Kempelen returned to Pozsony and he constructed a chess automat in 6 months' time and was ready to present it to the Emperor's court (Fig 1.2).

The automat was a closed-desk-like box, with a chess table on its top. Before the desk was sitting a Turkish puppet. In reality a chess champion was hiding inside the 'Turk' who won several tournaments of the court of count of Coblenz (Fig 1.3).

Fig. 1.2. Drawing about the Turks and the Chess-machine by Kempelen

Briefe
über den
Schachspieler
des
Herrn von Kempelen.
Von
K. G. v. Windisch.

Preßburg,
gedruckt und verlegt bey Anton Löwe.
1 7 8 3.

Fig. 1.3. Windisch's book about his idea of the Chess-player

Before the show Kempelen opened the doors of the desk and everybody could see the desk was empty. Of course this show was a trick. Kempelen has never unveiled the secret of the chess player. Nobody could expose the trick, because unfortunately the chess player was burned to death in a Philadelphia fire on 5th July 1854.

There are different versions of explanations about the secret of Kempelen's chess automat, but nobody was able to figure it out. Probably a gnome friend of Kempelen was hiding in the desk, (Fig. 1.4) but the several mechanical tricks of Kempelen are unknown, e.g. how could the gnome see the chess table from his closed place and how could the 'Turk' move the chess pieces with his hands (Fig. 1.5).

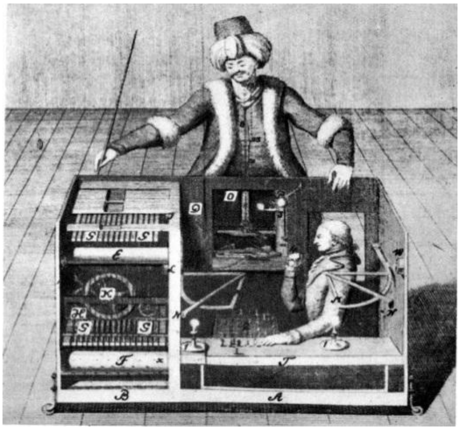


Fig. 1.4. A conception about the chess-machine by Raknitz

Fig. 1.5. A book about the chess-machine of Kempelen written by Rachnitz. 1789

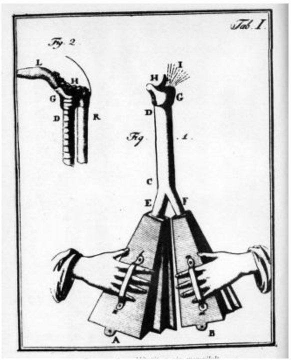


Fig. 1.6. The first model of Kempelen's speaking machine, right and left lungs (A,B), windpipe (CD), larynx (H), tongue (L), pharynx (R)

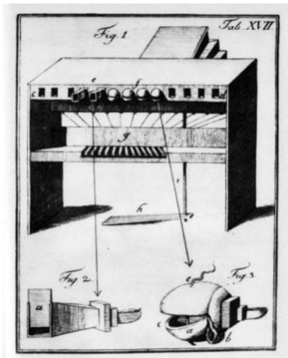


Fig. 1.7. A speaking-machine. Blow-pipe, air suction-pipe, and wind-box, then 13 pieces of mouth elements from wood, with ivory tongues

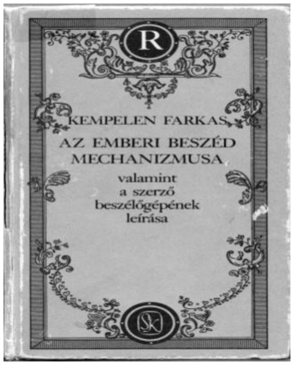


Fig. 1.8. Kempelen's book about the speaking machinme, The Mechanics of the Human Speech. (Orig. German, 1791)

Kempelen usually told his friends the automat player was not a very clever innovation, it was a toy. He had a real innovation too: a *talking machine* that was developed after long research. He conducted very long acoustic and phonetic studies before he started the construction of the talking machine (Fig 1.6).

The machine was ready in 1773 and it was able to say a long sentence: *Venez, Madame, avec moi a Paris.* The machine spoke French (Fig 1.7).

Kempelen wrote a book about the machine in 1791 (Fig 1.8).

The Kempelen talking machine was copied several times, e.g. by acoustical specialists: the Englishman Ch. Wheatstone, the German Posh, the Austrian J. Faber and the French J.S. Lienard, too. A copy of it is preserved in Budapest up to this day.

Another important innovation of Kempelen was a printing-press, which produced a convex script for blind people, including one of his blind friends too.

2 Ányos JEDLIK (1800-1895)

Fig. 2.1. Ányos Jedlik - the inventor-priest

Every secondary school student knows in Hungary that the *concept* of the *dynamo* was invented by **Ányos Jedlik** in the 19th century, in 1861. His results are not known in other parts of the world, because Jedlik did not patent the dynamo and the electrical engine. These machines were patented by Werner von Siemens, six years later, in 1867.

Ányos Jedlik was a professor of mathematics and physics. He was a monk of the Holy Benedictine Order, a priest (Fig 2.1). He finished his secondary studies at the schools of the Order and did his tertiary studies at the Budapest University of Sciences. He earned his PhD degree at the same university in 1839. Then he was an appointed professor of physics and mechanics at the Budapest University of Sciences. He became the Dean in 1848 and in 1863 the Chancellor of University. In 1858 he became an elected member of the Hungarian Academy of Sciences. He retired in 1879 and returned to Győr. He died in 1895.

In the centre of Jedlik's life was the physics. His first interest was mechanics, he wrote several books on this subject matter, but later his main interest was the ultimate science of the time: **electricity**.

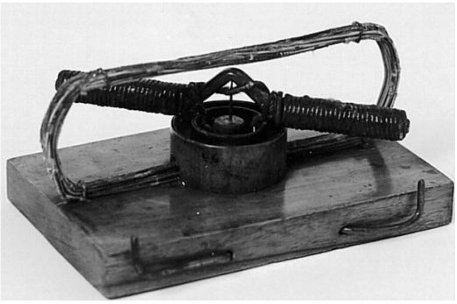


Fig. 2.2. The first electro-motor

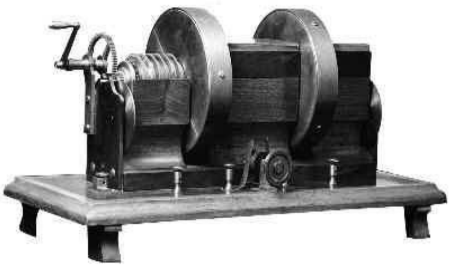


Fig. 2.3. The Jedlik's Dynamo

As I wrote above, he studied the different types of electrical motors, he published first an exact description of the different sorts of electrical motors (Fig 2.2) and he designed the first dynamo (Fig 2.3). He had other electrical inventions too, e.g. voltage multiplication (based on the Leyden jar), this design of Jedlik was awarded at the Vienna World Exhibition in 1873.

Jedlik constructed the first Hungarian **mechanical analogue graphical calculator** in 1878, which was not well-known among the scientists of the time. The Hungarian name of the equipment was: **rezgési készülék (**equipment of vibration). This piece of equipment (Fig 2.4) was able to add graphically two oscillations and one travelling motion, this way the Lissajous curves could be drawn. Bálint Laczik, a lecturer of the University of Technology Budapest, wrote a computer model of the concept of the Jedlik's equipment. He was astonished to see that there was no difference between the results of the more than 120 years old mechanical machine and the computer.

This equipment is now on display in the Hungarian Technical and Transport Museum.

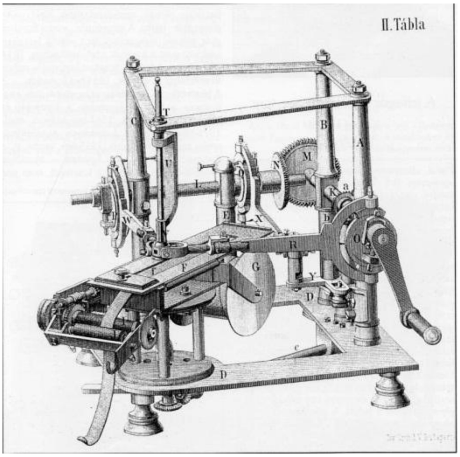


Fig. 2.4. Mechanical graphical analog calculator (Hung. rezgési készülék)

Fig. 2.5. The soda-water

Jedlik had a very well-known and world-wide used innovation, the **soda water** (Fig 2.5). Therefore the owners of Hungarian soda water and sparkling water factories celebrate Jedlik's birthday every year.

3 István Juhász (1894-1981)

He was an excellent mechanical engineer. Together with his younger brother he bought a bankrupted manufacturing plant: **GAMMA Technical Private Listed Company (GAMMA Műszaki Részvénytársaság).** István Juhász was not only an engineer (Fig 3.1), but a very good organiser and businessman too. Therefore his factory was very soon successful.

Fig. 3.1. István Juhász

Their main products were scientific types of measuring equipment for geodesy, but he reorganised the optical division of the company and they started to produce e.g. the Barabas-telescope, the light phone and the artillery periscope etc.

The Juhász brothers served in World War I as artillery men where they saw the first airplane military actions, which was a very useful experience to them. They decided their factory would produce a piece of equipment to help the artillery to shoot down airplanes. They were sure, if a new war broke that out the air force would be the most important division of the military. If the airplane was to be such an important weapon, then the army has to organise a very effective defence.

When WW I came to an end, István Juhász began to test an existing anti-aircraft gun director, designed by a retired artillery officer, *Sándor Szabó,* a reserve lieutenant. István Juhász was not satisfied with the result of the test; therefore he stopped the development of the equipment and designed a new, electro-mechanical one. He found the following problems: the GAMMA-Szabó equipment was inaccurate, it was difficult to transport and their operation was unstable.

He decided to change the main characteristics of the new GAMMA-Juhász anti-aircraft gun director. First of all, his equipment controlled 4 anti-aircraft guns that shot at the same time with parallel gun-barrels (Fig 3.2). He used a Hungarian patent, the self-exploding munitions. The time-controller of the munitions could be adjusted before shooting. The measuring telescope of the anti-aircraft gun director measured very quickly the height, the speed and the direction of the enemy airplane and then the equipment calculated very quickly the parameters of the 4 guns aiming for the target, i.e. the airplane. Then the guns fired, the munitions exploded near the airplane. If the airplane was between the 4 explosions, it was destroyed.

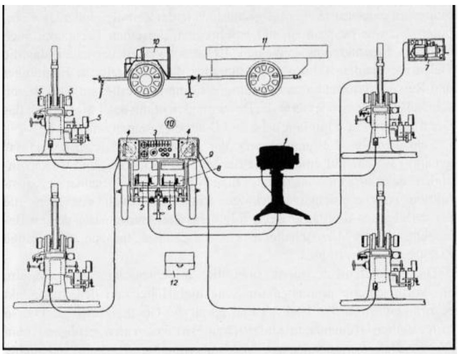


Fig. 3.2. The GAMMA-Juhász anti-aircraft gun-director controlled 4 guns simultaneously

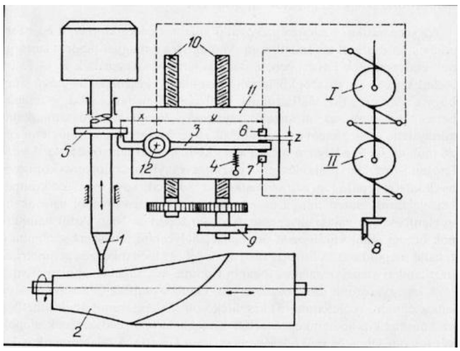


Fig. 3.3. The memory of the GAMMA-Juhász equipment, the ballistic body

The other great innovation in the equipment was a ROM (!), a *ballistic body* (Fig 3.3), which contained the possible ballistic paths of the given gun and munitions, which were selected by the anti-aircraft gun director depending on the measured parameters.

The first official international presentation of the GAMMA-Juhász equipment was in Sweden in 1932, where several companies and military representatives were invited (Fig 3.4). The results of GAMMA-Juhász equipment were the best one. The competition came from the factories of Germany, France, the Netherlands and

Belgium. They were not as successful as the Hungarian one. After the Swedish presentation the GAMMA Works received a lot of orders from the armies of Sweden, Norway, Argentina, China, Finland an Iran. It was a really great business success.

Fig. 3.4. The GAMMA-Juhász equipment in the Swedish army

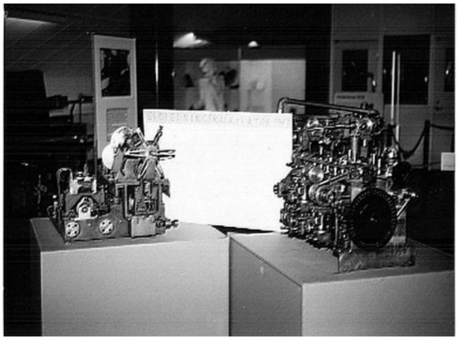


Fig. 3.5. The GAMMA-Juhász equipment - made in Sweden - in the Stockholm museum

The Swedish Industry bought the GAMMA-Juhász licence and organised a joint venture, called ARENCO AB, manufacturing the *GAMMA-Juhász anti-aircraft gun director* (Fig 3.5). This was a barter business: the Diósgyőr Steel Factory received the licence of the famous Swedish BOFORS 75 mm anti-aircraft Machine-Gun. From this time on the Swedish and the Hungarian companies delivered similar, complex anti-aircraft systems to military organisations world-wide (Fig 3.6).

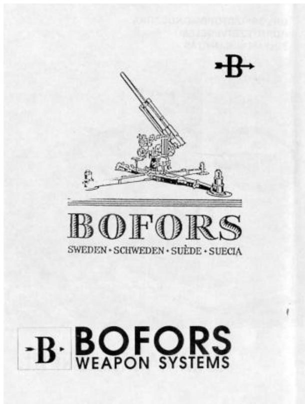


Fig. 3.6. The Swedish Bofors Weapon System

After WW II, when the Soviet army invaded Hungary, they took possession of the GAMMA Works too. The majority of the tool machinery was dismantled and transported to the Soviet Union, as a compensation for the war damages. István Juhász tried to resume the manufacturing with the remaining machines, but his factory was nationalized and the State representatives declared in 1945, he was unfit to manage the company. He retired and died in 1981.

The Soviet authorities instructed the GAMMA factory to manufacture a *Soviet anti-aircraft gun director instead of the Hungarian one: PUAZO.* It was not as modern and as successful to the Hungarian army as the GAMMA-Juhász equipment was. Therefore the leaders of the Hungarian army and the officials decided in 1953/54 that they would stop the manufacturing of the PUAZO and they would resume the development of the GAMMA-Juhász equipment. They renamed the equipment to *Model 'E2'.*

The new *Model 'E2'* was ready in 1956. It was used it in the six-day Arab-Israeli war in 1967, where its legendary achievement was to shoot down a *Mirage hunter-airplane* made in Israel.

4 Tihamér Nemes (1895-1960)

He was a very talented mechanical engineer, who earned his bachelor degree at the Budapest University of Technology in 1917. His first job was at the Lloyd Aircraft Factory. He was interested in the emerging telecommunication opportunities and in 1921 he joined the new 'Telephone Herald' company (telefonhírmondó). This system was a Hungarian patent invented by Tivadar Puskás. Tihamér Nemes was appointed as a chief-engineer. He was a very good organiser and therefore he changed his job. He was appointed the director of the Electrical and Precision Mechanics Public Limited Company (PLC).

He did not like the administration and tendered his resignation as director. He resumed his carrier at the research institute of the Hungarian PTT as a scientist.

His first task was the development of a new telephone set: CB 35 that was used in Hungary as well as abroad for a long time. Back then one of his tasks was to check and repair the telecommunications problems of the phone system, e.g. he measured and corrected the intelligibility of the communication through phone lines.

Fig. 4.1. Tihamér Nemes

He was one of the first specialists interested in the very early television technology. He had some patents and he was a member of the first team that carried out the first television broadcasting in Hungary.

Tihamér Nemes (Fig 4.1) was the first researcher, who was interested in cybernetic problems in the mid-thirties. However Norbert Wiener (1894-1964) founded the science of cybernetics only in 1948 in his book: *Cybernetics: Or Control and Communication in the Animal and the Machine.*

Tihamér Nemes decided to construct models to simulate human and animal activities. His first constructions were logical machines, animating human thought. He built a copy of the Jevons logical machine from wood (Fig 4.2, 4.3 and 4.4). His next construction was the pocket logical machine (Fig 4.5) that could be used for solving logical tasks. He constructed a genetic logical machine with relays (Fig 4.6) too.

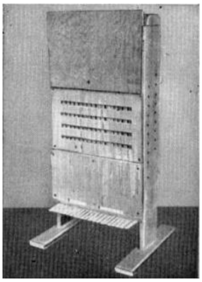


Fig. 4.2. Copy of the Jevons logical machine from wood

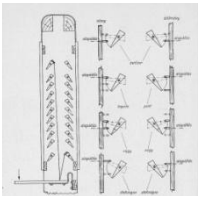


Fig. 4.3. The Jevons logical machine. Wood and String, by Nemes

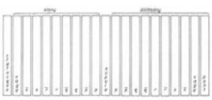


Fig. 4.4. A Jevon's logikai piano, keyboard

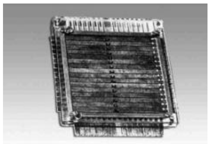


Fig. 4.5. The pocket logical machine

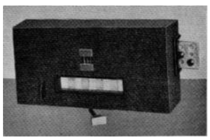


Fig. 4.6. The genetic logical machine (with relays)

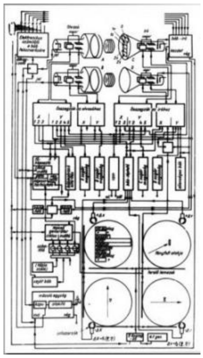


Fig. 4.7. Nemes' chess machine for solving a chess-task in two steps

Tihamér Nemes was a good chess player and designed in 1949 a chess playing machine. Unfortunately he could not build his machine because the technology was not developed enough at the time to construct such a machine. Unfortunately the majority of his ideas and patents remained only ideas and were not implemented (Fig 4.7). He patented many of his ideas. Some of them were: a letter-reading machine, artificial animals, a gaming machine, a reproducing machine, a walking machine, a talking machine etc.

He was very interested in the computer. Our team constructed the first Hungarian computer, the M-3 between 1957 and 1959. Tihamér Nemes visited me at least once a week in our laboratory where he asked me a lot of questions about the computer as he was interested in all of the details. He was ill at the time and died before he could see a real, working computer based on the von Neumann concept.

His posthumous book *Kibernetikai gépek* (Cybernetic Machines) was published in 1962 from his papers collected by his friends after his death (Fig 4.8).

Fig. 4.8. Nemes, Tihamér his posthumous book, Cybernetic Machines

5 László Kozma (1902-1983)

I am feeling a special affiliation with him, because he was my favourite professor at the University of Technology, Electrical Engineering Faculty (Fig 5.1).

Fig. 5.1. László Kozma, 1959 Aug. 3

His life was not very easy, his parents were not rich and he was a young Jewish man, when the anti-Jewish Law was legislated in Hungary. The law admitted only a limited number of Jewish students to the University. It was the first law against Jewish nationals in Hungary. One of his relatives offered him a job at the largest electric factory in Hungary: *Egyesült Izzó*; its name was changed later to *TUNGSRAM*. He became a phone operator and was working at the manually operated telephone-exchange.

His command of English was very good. Therefore the engineers of the factory gave him electric designs of the new automatic telephone exchange circuits that they wanted to manufacture at the factory. Kozma was very soon regarded an authority on automatic telephone exchanges at the factory.

When the engineers saw the talent of the young colleague, they gathered enough money at the factory and they granted him a scholarship to study at the Brno University, where such anti-Jewish Law did not exist.

He graduated from the University as an electrical engineer in 1930. Then the factory suggested to him to go to Antwerp and join the European plant of Bell Telephone Laboratories. This company was the world leader of automatic telephone-exchange products. The Hungarian Plant wanted to buy the telephone exchange licence from this company in Antwerp too.

He was very successful at the factory, he submitted alone or jointly with his colleagues 25 patents between 1934 and 1938.

The director of development tasked him to design calculators from telephone exchange parts manufactured in the factory (Fig 5.2). László Kozma did not know anything about the race in the United States to manufacture super-fast calculators. This effort was driven by the Americans' belief that the next world war would be won by a country that produced the fastest calculators.

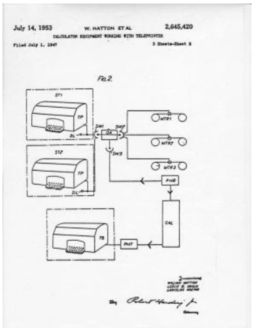


Fig. 5.2. A patent of László Kozma, in Antwerpen 1947

Fig. 5.3. George G. Stibitz

Fig. 5.4. BELL Relay Calculator, designed by Stibitz

Laszló Kozma commenced working on the calculator at the beginning 1938 and it was available before the end of the year. His second calculator was ready before the end of 1939. It was a very interesting design that was used by the whole company. Every department had a terminal that was connected to the calculator through a local telex network. A steel memory wire was connected to the telex-network, too. I believe it was the first calculator network in the world. A similar American network designed by George Robert Stibitz (Fig 5.3), and installed by the American Bell Laboratories was ready only in 1940.The centre of the network was one of the first American Relay calculators (Fig 5.4). The factory protected Kozma's calculators with 10 patents.

On 10th May 1940 the German Army occupied Belgium, but the factory was working continually. The director sent the Kozma calculator to the US, but he was told the machine never arrived there, because a German submarine sunk the ship. (Who knows?)

Kozma had to leave the factory and Belgium too, because the Germans started to collect the Jewish people and delivered them to the concentration camps. Kozma could have travelled to the UK, or US, but he believed in the Hungarian authorities and returned to Hungary in 1942. Soon after his arrival he was deported to Mauthausen, but he survived it (Fig 5.5). His wife did not.

He returned to Hungary and participated in the reconstruction of the demolished telephone exchanges in Budapest. He was one of the professors who established the new Faculty of Electrical Engineering at the Budapest University of Technology and he became one of the first professors. Then he was employed by the Ericsson telephone exchange factory in Budapest and was the technical director of the plant. Kozma was awarded the Kossuth Prize for his role in the reconstruction of the telephone exchange.

The communist government arranged a show trial against the factory at the end of 1949, because they wanted to nationalize the foreign company. They sentenced the leaders of the plant, including László Kozma, to a long imprisonment. The reason was: sabotage, which, of course, was untrue.

Kozma got out of the prison in 1954. He was rehabilitated and was reinstated in the professorship and the Kossuth Prize. He became a correspondent member of the Hungarian Academy of Sciences in 1961, then in 1976 a regular member.

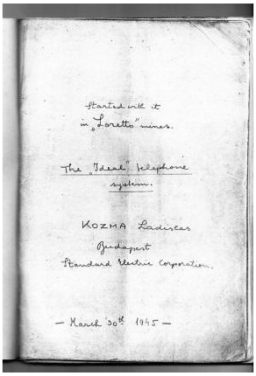


Fig. 5.5. An ideal phone system designed by Kozma in the concentration camp

Fig. 5.6. The MESz-1 computer constructed by László Kozma, in the museum

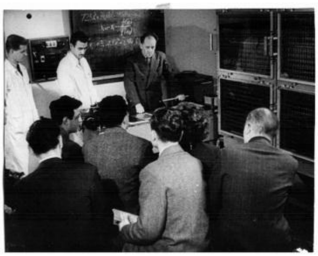


Fig. 5.7. Kozma László teaches before his computer

He constructed the first Hungarian relay computer, the MESz-1 (Fig 5.6), at the University of Technology. This computer operated with 8 digit decimal numbers, i.e. 27 digits binary numbers, its storage capacity was 12 decimal numbers and it could be controlled by instructions. The instruction input was a punched card with a set of 45 possible instructions. The output was a Mercedes typewriter, adapted by László Kozma. The machine was working with 2,000 relays (Fig 5.7).

He died on 9th November 1983.

6 John von Neumann (1903-1957)

Fig. 6.1. John von Neumann (1940)

Fig. 6.2. The Neumann family's children, (from left) Nicholas, Michael, John and their cousin, Lily

I think he is the best-known computer scientist in the world. His original Hungarian name was: Margittai Neumann János Lajos (Fig 6.1). His father: Margittai Neumann Miksa was ennobled by the Hungarian king and Austrian emperor Franz Joseph. This was an acknowledgement for his support to the development of the Hungarian economy as a banker at the beginning of the 20th century.

His oldest son was John, the middle one was Michael and the youngest brother was Nicholas (Fig 6.2). The three children graduated at the very famous secondary school in Budapest, the Lutheran Gimnazium, where several Nobel Prize winners studied.

John was very good at mathematics and physics, but he also learned different languages easily. He spoke fluently French, German and later English too. One of his favourite foreign languages was ancient-Greek.

His mathematics teacher was *László Rátz*. When he discovered the mathematical talent of John he allowed him to visit some famous mathematicians instead of the mathematics class. He visited very famous Hungarian mathematicians on a regular basis, discussing mathematic problems of the time.

When he completed his secondary education his teacher suggested to John's father that his son enrol at the University of Sciences to study mathematics. His father disagreed and decided that John would become a chemical engineer and commence his studies in Zurich. The Jewish families were very well organised; the decision of the father could not be questioned. John followed his father's instructions and travelled to Zurich to begin his studies at the *Eidgenössische Technische Hochschule*. Concurrently he also started his studies in the Budapest University of Sciences, where he learned mathematics, physics and chemistry. He was also a student of the University in Berlin, where he attended the lectures of Albert Einstein.

He graduated as a chemical engineer in Zurich, did his PhD in mathematics in Budapest and a little bit later he habilitated at the Berlin University.

His first job was in Göttingen, because he decided to work with David Hilbert, the famous German mathematician (Fig 6.3).

At that time the Nazi party gained dominance and a Nazi person was appointed the new chancellor of the Göttingen University. This was a strong incentive for the Jewish scientists to immigrate to America. Von Neumann accepted a guest lecturer invitation at Princeton University in the US in 1930. He was appointed very soon an ordinary professor of mathematics at the University, a little bit later the mathematics professor of the Institute for Advanced Study (Fig 6.4).

Fig. 6.3. David Hilbert

Fig. 6.4. The Hall of the IAS

Fig. 6.5. Hermann Heine Goldstine

When the Nazis started WW II, he offered his cooperation to the American Army. He worked in Aberdeen too, but he served most famously in Los Alamos, were the best scientists of the United States were developing the Atomic Bomb. He was an advisor of the Research Institute. His main invention was the *implosion*, a method to detonate the bomb.

He had to perform a lot of calculations to do his job, but the electro mechanic calculators were very slow at that time.

Von Neumann met Hermann Heine Goldstine (Fig 6.5) at the Aberdeen train station in the summer of 1944. Goldstine informed von Neumann about the ENIAC project, i.e. they were working on an electronic calculator that was several thousand times faster than the ordinary electro-mechanic calculators.

Von Neumann was enthusiastic to find a calculator that was needed for his project in Los Alamos. So he visited Moore School at the University of Pennsylvania (PENN) as soon as possible to gain first-hand experience with the ENIAC (Fig 6.6).

Von Neumann joined to the ENIAC team, but he did not maintain a good relationship with the inventors of the ENIAC: John Mauchly and Presper Eckert (Fig 6.7). He criticised the architecture of the ENIAC, because the programming of the machine was very cumbersome. The size of the memory was also not large enough. So von Neumann decided that he would design a new computer. Hermann Goldstine agreed.

Fig. 6.6. The ENIAC in operation

Fig. 6.7. John William Mauchly and John Presper Eckert

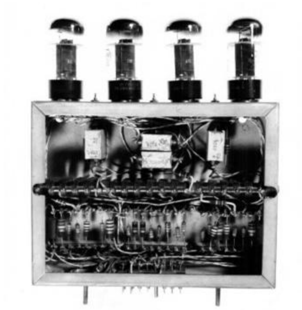


Fig. 6.8. The logic unit of EDVAC

Von Neumann returned to Los Alamos. On 30th June 1945 he travelled to PENN again taking a 101 page study with him: *First Draft of a Report on the EDVAC, by John von Neumann* (Fig 6.8).

It was the first description of the *von Neumann stored program concept*. The ENIAC team began to build the EDVAC, but von Neumann was dissatisfied with the new machine.

The war ended, Mauchly and Eckert founded the first computer factory: *Electronic Control Co***.** where they manufactured the successor of EDVAC, the BINAC and later the UNIVAC.

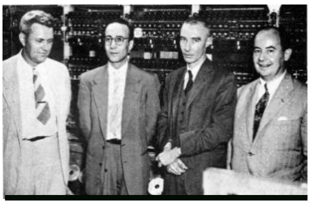


Fig. 6.9. photo before the IAS Computer, (from left), Julian Bigelow, Hermann Goldstine, Robert Oppenheimer and John von Neumann

Von Neumann decided he would design a new computer, a different one from the EDVAC. The Institute for Advanced Study (IAS) started a new computer programme. Some scientists from the earlier ENIAC programme joined to the team, e.g. Hermann Goldstine, Julien Bigelow, (Fig 6.9) Arthur Burks and some new members, e.g. James H. Pomerene and *dr. Jule G. Charney***.** He developed meteorology forecast programs following the von Neumann numerical meteorology concept.

Von Neumann fixed the problems of the EDVAC. In this new computer program, e.g. parallel operations were in the IAS machine, they used one-address instructions, a relatively large (1 k/word, 40 bits/word) operative memory with Williams tubes, a magnetic drum as a background memory and IBM punch card I/O. A brand new element also emerged: a graphical display transformed from an oscilloscope.

This was a very important event: Von Neumann and his colleagues declared that the concept of the IAS computer was freely available for everybody, i.e. they did not patent it. They distributed the design of the computer, not only in the US, but world-wide, making it possible that several IAS computer clones were built simultaneously all over the world. Some of the clone computers were: **ORDVAC** (Ordnance Variable Automatic Computer), **ILLIAC** (Illinois Automatic Computer), **AVIDAC** (Argonne National Laboratory), **ORACLE** (Oak Ridge National Laboratory), **MANIAC** (Mathematical Analyser Numerical Integrator, and Computer, 1952), and the **JOHNNIAC**, (Johnny Integrator and Automatic Computer) etc.

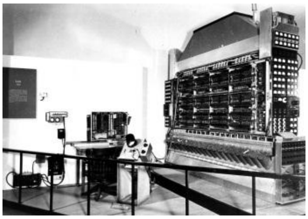


Fig. 6.10. The IAS Computer in the Smithsonian Museum

The IAS computer was ready and presented to the public in 10th June 1952.

Von Neumann died in 1957 and Hermann Goldstine became the head of the IAS Computer Programme. Hermann Goldstine and Robert Oppenheimer, the director of the IAS, foresaw that computers would be manufactured in factories and not in research institutions. They decided to terminate the research programme at the IAS.

The computer installed at IAS was relocated to the Princeton University in 1958, where it was used for some years. Today the IAS machine is in the store of the Smithsonian Institute (Fig 6.10).

7 László Kalmár (1905-1976)

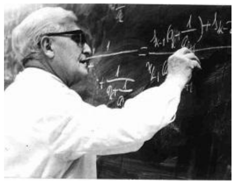


Fig. 7.1. László Kalmár, the university professor

Szeged is quite a large university city in South Hungary. The József Attila University of Sciences is the main university of the city, and at that time it was famous for several mathematical departments and a specialized one was the School of Cybernetics. The founder of the department was Professor László Kalmár (Fig 7.1) in the 1950s, a mathematician and academic of the Hungarian Academy of Sciences.

Professor Kalmár taught mathematics, logic, set theory and analysis. He developed a computer science-mathematics curriculum, the first one in Hungary.

He established the Cybernetic Laboratory in 1963, where a lot of young scientists started their careers: mathematicians, engineers, physicists and other specialists. Szeged was the centre of Cybernetics in Hungary.

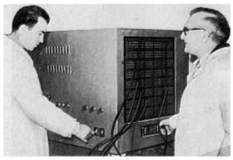


Fig. 7.2. László Kalmár, (right) and Dániel Muszka at the Kalmár Logic Machine

The university had no computer of its own at that time, Professor Kalmár designed a logic machine with relays that was built by Dr. Dániel Muszka, his assistant. Its official name was: *Kalmár's or Szeged logic machine* (Fig 7.2)

The machine was able to solve different logic tasks with a lot of variables and was ready in 1958. It could be controlled by a wired program (Fig 7.3). The control program was a physical net of cables connected to the machine.

Professor Kalmár tried to find real-life applications for the logical machine, e.g. a railway network controller, but the management of the Hungarian State Railways declined his suggestions.

The Cybernetic Laboratory designed several different cybernetic machines, e.g. an automatic traffic lights control system that was installed in Szeged, carried out by Győző Kovács and Dr. Dániel Muszka. Dr Muszka put a lot of efforts into cybernetics aided driver support in the automotive industry.

Dr Muszka designed the first artificial cybernetic animal: the *Szeged Electronic Ladybird*, (Fig 7.4) which was a demonstration of the conditioned reflex in the animals (Fig 7.5).

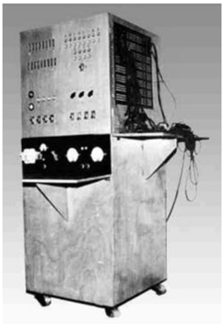


Fig. 7.3. The Kalmár Logic Machine

Fig. 7.4. Dr Dániel Muszka and the Ladybird Logic Machine

Fig. 7.5. The Szeged' Ladybird

The first Hungarian computer, M-3, was installed in 1959 in Budapest. In 1965 it was transported to Szeged, where the computer was in operation till 1968. It was the first computer of the country and the first computer centre established in a countryside town. This centre played a pioneering role in computer use in Hungary.

8 Rezső Tarján (1908-1978)

He was a Hungarian mathematician. During his teenage years he was interested in radio technology and was a passionate amateur radio operator. He graduated from the secondary school in 1925. He could not continue his studies at the Budapest University of Sciences because the 'Numerus Clausus Act' was legislated. This quota limited the percentage of Jewish students at universities.

Fig. 8.1. Rezső Tarján

He was employed as a blue-collar worker for a year and thereafter he enrolled to the University of Vienna. He studied physics, mathematics and philosophy. He earned his PhD degree in physics in the early 1930s, the subject of his thesis was light dispersion.

His first job was at 'Janus', later at 'Adria' Insurance Company, as an insurance mathematician, but he continued his activity as an amateur radio operator, too. He also designed a *signal generator*.

During WW II, in 1944, he had to perform forced labour in-lieu of conscription as young Jewish men were prohibited to serve in the armed forces. During this service he fell seriously ill, but he survived.

After the war he was left a scientist and as such he was appointed to several senior executive positions (Fig 8.1). He joined Tungsram (Egyesült Izzó) and later was promoted to managing director of the Telecommunication Industry. At the beginning of the 1950s he was appointed as a deputy director of the PTT Research Centre.

Tarján had personal connections with a lot of Western scientists from different backgrounds. The communist government of the time deemed it as a crime and he was sentenced for spying on his own country entrusted by Western powers in 1953. He was released from the prison after 2 years.

The Hungarian political police established a 'research institute' called *'KÖMI 401'* in the prison. Lots of scientists, engineers etc. who were sentenced to long imprisonment could continue their work in the prison at a very low 'salary'.

Tarján was in the prison with two of his friends: *József Hatvany*, a physicist and *László Edelényi, a* mechanical engineer. These three prisoners offered to the Mathematical Department of the Hungarian Academy of Sciences (HAS) that they were prepared to design a computer for the HAS. Their conditional offer was subject to the HAS's obligation to carry out and install their computer design. It is noteworthy to mention that the letter was sent via the political police and the names of the authors were omitted. The HAS rejected the offer.

When Tarjan was released in 1955 he contacted some academics of the HAS asking for their help to realize his idea, i.e. the installation of the first Hungarian designed computer. The HAS was in a very uncomfortable situation and took a chance by allowing Tarján to realize his dream of the computer. He headed up a department in a research institute of HAS, where he started building the first electronic computer, the B(udapest)-1. Hatvany, Edelényi and a few new colleagues comprised his team.

Tarján's original idea was to clone the EDVAC or EDSAC to enable him to visit Maurice Wilkes in Cambridge. Wilkes offered him a working installation of the mercury delay line, but Tarján told me it was too heavy to be transported to Budapest.

His team began to develop B-1, but they wanted to design a *nickel delay line*. Unfortunately they did not succeed.

Tarjan did not stop 'bombarding' the HAS, because he desired to found an independent research institute to conduct cybernetics and computer research. Finally, in 1956, the Presidency of the HAS decided to establish a research team and called it *Cybernetic Research Group of the HAS* (Fig 8.2).

The academic circles were enthusiastic, not so Tarján. A Soviet emigrant, Sándor Varga, was appointed as the director and Tarján was the Scientific Deputy Director. The goal of the institute was to conduct cybernetic research and to install *the first Hungarian computer,* under Tarján's leadership. The HAS permitted that the research team begin its work and new members joined as specialists, mathematicians, engineers, economists etc. Four young mathematicians and four young engineers were selected by Tarján, they were 1956-57 university graduates. I was very fortunate to be one of them (Fig 8.3).

The big problem was we had not seen any real-life electronic computer before. The engineers visited the MESz-1 relay computer installed by *László Kozma* at their University, the mathematicians read books, articles about computer programing, but it was not sufficient for designing and installing a computer. Unfortunately Tarján's computer knowledge was also insufficient to organise and control the research work (Fig 8.4). So we did our best without any specific goal and result (Fig 8.5).We had no such opportunity, as e.g. the English, Swedish, Danish or the US specialists, when they designed the *EDSAC, ILLIAC, MANIAC, JOHNNIAC* and the other EDVAC and IAS clone computers (Fig 8.6).

Fig. 8.2. The Nídor utca 7, where the first computer - the M-3, was constructed

Fig. 8.3. A study from the HAS Cybernetical Research Group, accepted by Varga and Tarjan

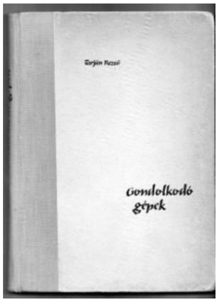


Fig. 8.4. The book of Rezső Tarján, the thinking machine

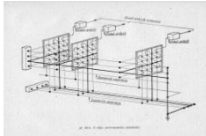


Fig. 8.5. A page from Tarján's book, the ferrite-memory

Fig. 8.6. The first publication possibility in Computing Science

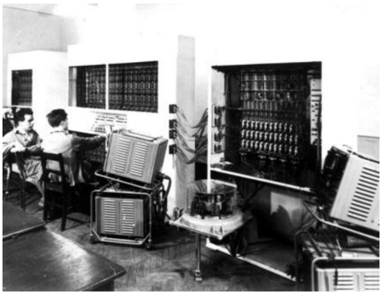


Fig. 8.7. The M-3, is ready, before the computer, Győző Kovács and Bálint Dömölki

Conducting research without any tangible result was not acceptable for Varga because he wanted to see a working computer within the shortest possible time. He stopped the development of B-1 and visited his earlier research institute in the Soviet Union: the *Institute of Energy Policy in Moscow*. One of the first Neumann concept computers, the M series, was being designed there at that time. Varga offered to undertake with him the design of their newest, medium-size computer, M-3, for the use of our team. They told him they were sure the Hungarian team would be unable to install the M-3 based on their blueprints and instructions. Four other countries, China, Estonia, Armenia and Belorussia, asked for a similar support at that time and were given the same blueprints.

The wooden box containing the blueprints and the necessary parts for the computer arrived soon and we started to build the M-3 computer. Varga reorganised the research team, Tarján was stood down, Bálint Dömölki was appointed as the head of the computer installation team, and me, as his technical deputy (Fig 8.7).

The task was very interesting. Dömölki was the first person in Hungary, who interpreted and understood the operation of a computer from blueprints. We, engineers, understood the operation of the circuits and finally the computer was turned on to start its first operation. The main memory of the computer was a magnetic drum, which was not too fast. The first version of the machine was able to execute only from 30 to 50 instructions/sec. It was a fantastically high speed for us, because in the nineteen-fifties our computer was the largest electronic equipment in our country and the fastest calculator (Fig 8.8).

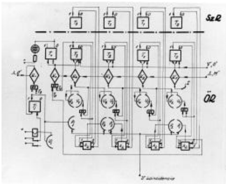


Fig. 8.8. A drawing about the logical circuits of the M-3


Esti Hírlap

Fig. 8.9. News from the Newspaper 21st January, 1959. The M-3 is ready

We changed the logic in some parts of the computer, because we felt, we were very ‘smart’. All together five M-3 computers existed in different countries as mentioned above, but they were incompatible with each other. We built in new instructions, our computer was able to play music, I connected not only one, but two magnetic drums via a new drum-controller to the central processor, as a background memory. We changed the original I/O units to an 8 channel paper-tape reader/punch unit etc (Fig 8.9).

The mathematicians solved a lot of interesting mathematic tasks, e.g. we controlled the stability of the longest new bridge across the Danube. The economists also developed several tasks, e.g. they calculated the planning tables, which would be called spreadsheets today, of the 5 year state plans of Hungary.

Varga renamed our team, once the computer was operational, the new name was: *The Computer Centre of the HAS.* (Magyar Tudományos Akadémia Számítástechnikai Központja).

HAS stood down Sándor Varga as managing director soon, because he allowed us to build a new, most modern, computer and he did not ask for the permission of the HAS. It was a very big offence, he had to leave and we had to disassemble the half manufactured second computer. The reasoning behind the HAS’s decision was that the *first M-3 was powerful enough for the HAS’s needs for at least 5 years’ time!* The decision was supported by the whole presidency of the HAS.

Tarján joined the Department of László Kozma at the University of Technology Budapest. A short time later he was invited to the National Committee of Technical Development. He worked in his scientific advisory capacity for the Unified Computer System of the Socialists countries. He passed away in 1978.

9 Árpád Klatsmányi (1923-2007)

He was an eminent student at secondary school and graduated in 1941 with high distinction. He excelled in mathematics and logic and possessed a powerful ability to comprehend new subject matters very quickly.

He was studying mechanical engineering from 1941 till 1947 at the University of Technology (Magyar Királyi József Nádor Műszaki és Gazdaságtudományi

Egyetem). He was a university student for eight years, because WW II interrupted his studies. The young university students were transported to Germany in 1944, where they self-studied engineering. They returned home in December 1945. He received his degree in March 1947.

His first job was at Tungrsam (Egyesült Izzó), where he was involved with the development of different subject matters. These were related to high-power vacuum tube production and a number of various measuring instruments. He also took part in new transistor development projects. He continued his career at the University of Technology Budapest from 1952 to 1957. He took an active role in the 1956 Revolution and therefore he was discharged from his educational duties.

Fig. 9.1. Árpád Klatsmányi

After leaving the university he joined to a team of measuring instruments developers. From 1st June 1959 he was employed at EMG (The Factory for Electronic Measuring Instruments i.e. Elektronikus Mérőkészülékek Gyára) (Fig 9.1).

I heard about him at the university, but we did not meet. I joined SzKI (Co-ordination Institute of Computer Science) in 1969 and our laboratory was located next door to the EMG. I visited him and we became good friends. We informed each other about our work and results as we met every day in the canteen of his factory.

The factory produced traditional analogue measuring instruments, but Klatsmányi realised that the measuring industry was heading towards the electronic, digital technology. His idea was accepted by the management of the factory and he was appointed as the chief designer of the Digital Division.

He recruited a very talented team around him and they produced a lot of digital instruments, e.g.:

- The EDS digital module family, which was a multi-purpose device, e.g. they synchronised the network of traffic lights on the Grand Boulevard in Budapest.
- They designed and manufactured the first 'electronic pocket calculator – the HUNOR series', which was not only technically, but also commercially very successful. Klatsmányi and EMG were the first world-wide manufacturers of this piece of equipment. They started the development in October 1964 and the product was on the market in February 1965 (Fig 9.2).

Klatsmányi decided in 1965 that he would develop a real-life computer using transistors. He and his colleagues designed a computer, which was not only up-to-date, but contained several novel solutions, e.g.

- a BUS system was first used to connect the components of the computer.
- Klatsmányi designed a modular system, because he was thinking about a centralised service to repair the faults of the client computers. If a client ran a test program that detected a problem and the test indicated the problem was in one of the modules, the central service engineer replaced the module and tested the computer again. Later the faulty module was returned to the central service for repair.
- The peripheral devices of the computer were bought abroad and he used a foreign manufactured magnetic tape, a FACIT tape reader and punch and an IBM typewriter.
- Klatsmányi organised the training of the clients, too.

The central maintenance was included in the price of the computer.

The EMG 830 computer (Fig 9.3) was introduced to the computer specialists in 1968 at the first ‘Computer Technology 68’ conference of the John von Neumann Computer Society.

Fig. 9.2. Hunor 131. Electronic table calculator

Fig. 9.3. The control desk of the EMG 830 computer

Fig. 9.4. Jínos Sebesty vice President of the OMFB

1968 was an unfortunate year for Hungarian computer development, because the Soviet Union decided, and the other COMECON countries accepted, that the computer was a strategic product, consequently it could only be produced in the Soviet Union. The other Socialist countries would buy the computers from the Soviet factories. The computer development had to stop in the satellite countries, also in Hungary. EMG contrived that the EMG 830 was not a computer, but it was a controller for tool making machines, therefore they could continue its production. The Hungarian authorities accepted this fake argument and EMG continued to manufacture the ‘computer/controller’.

In January 1968 there was an important event in relation to computer production. A. Kosigin, the Soviet prime minister, overrode the decision of the COMECON and asked the leaders of the COMECON countries to participate in the production of a computer product family called *Unified System.* It was also called by another name: *Rjad.* Parts of the System would be manufactured in different countries. The other decision was that the parts of the System would be cloned from the IBM 360 (later IBM 370 – Rjad-2) series.

Fig. 9.5. The R-10, made by the Videoton (EC1011)

Hungary was responsible for the smallest member of the Rjad computer family: R-10, a small computer that did not even exist in the IBM 360 series. A deputy minister of the Hungarian government, *János Sebestyén*, headed up the Hungarian delegation (Fig 9.4). He and *Dr. Zsolt Náray,* the chief designer of the Hungarian Rjad computer, decided that the Hungarian Rjad Team would not copy any computer. We would buy a licence, together with the computer manufacturing technology.

Finally the French company, CII, sold its licences to Hungary. The first one was the CII 10010, later the CII Mitra 10 computer that was made compatible with the Rjad System by Hungarian specialists. The new name of the computer was: R10.

At that time the biggest experiment of computer design was run in the EMG factory. The Hungarian government appointed EMG to adjust the French computer to the Rjad System and to manufacture the R-10 computers for the other COMECON countries. EMG was also instructed to finish the manufacture of their EMG 380 computers.

Fig. 9.6. Árpád Klatsmányi and Marcell Jánosi

In a socialist system they had no other option, but stopping their own manufacturing, of which a few tens of computers had been produced and EMG started to manufacture the French computer with the new name R-10.

EMG had already produced one R-10 computer, when the government changed their mind and the R-10 production was given to a politically stronger factory, a Radio, TV and military electronic equipment factory: VIDEOTON (Fig 9.5).

The computer production at EMG was in terminal agony for a while, then Klatsmányi and his team, 130 well-trained computer engineers, resigned and the main computer production was continued at VIDEOTON.

Klatsmányi returned to education and lectured at different universities, he published 25 books, articles etc. Árpád Klatsmányi retired in 1983. He was a diamond-level university educated engineer and an excellent computer designer. He passed away on 1st July 2007 (Fig 9.6).

10 The TPA Computer of the Central Physics Research Institute (KFKI)

The Central Physics Research Institute owned the Research Institute for Measurement and Computing Techniques (KFKI MSZKI) (Fig 10.1), where electronic measuring instruments were developed to satisfy the needs of their physics research programme. One of their products was a *multi-channel analyser* which was used for outer space radiation analysis. The first measuring instrument they developed had no memory. Their next idea was, of course, to fit out the analyser with a memory.

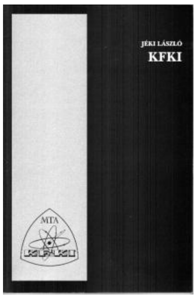


Fig. 10.1. A book about the KFKI by László Jéky

At that time the first URAL-1 computer arrived at KFKI and later the next computer, an ICL 1905. Probably the original idea that triggered buying a computer was to replace the memory equipped. They aimed for best possible computer, a PDP-8 from Digital Equipment Corporation.

They made enquiries about buying it, but DEC could not sell it to Hungary, because selling computers to a socialist country was embargoed by the US. (COCOM embargo)

When they studied the PDP-8, and they visited various computer exhibitions, they got a book distributed by DEC. Its title was: *Digital Small Computer Handbook.* It was also in the same computer exhibition, where I got a similar handbook. This handbook contained the logical descriptions and drawings of a computer. They thought they would be able to clone the PDP-8 computer from this handbook (Fig 10.2).

The handbook also contained another important piece of information, the description of the *instruction set.* Additionally they had a modern ICL computer at their disposal and they could copy its mechanical solutions which were used to reconstruct their own PDP-8 clone. They couldn't believe their luck: it was using the same instruction set.

Around the same time new transistors from Japan became available in Hungary, so the inability to buy active elements needed for computers did not exist anymore. When they were studying the application of the PDP-8 computer, they realised that it could be used for scientific analysis, which was necessary to the physicists of the KFKI (Fig 10.3).

At the end they built a copy of the PDP-8 in an ICL-like form and its new name was TPA 1001 (Fig 10.4), the abbreviation for stored program analyser in Hungarian. The computer was on display in 1968, together with the EMG 830, at the '*Computer-Technology '68'* conference of the John von Neumann Computer Society. This new name of the computer/analyser was important, because the COMECON countries decided: computers would be manufactured only in the Soviet Union, but *the TPA 1001 was not a computer, it was a stored program analyser (i.e. TPA)!*

Fig. 10.2. József Lukács, one of the leading constructor of the TPA computers

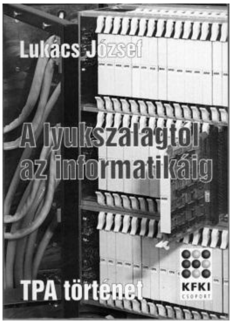


Fig. 10.3. A book by József Lukács about the story of TPA computer: *From the Punched Tape to Information Technology*

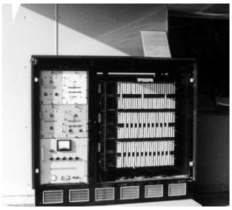


Fig. 10.4. KFKI TPA 1001, the clone of the DEC PDP-8

Fig. 10.5. KFKI TPA computers

The Hungarian PDP-8, i.e. TPA 1001, differed only in minor details from the original one. Because of the close match, all the original DEC PDP-8 programs were running on the TPA 1001 computer, that was demonstrated to the DEC software specialists.

DEC consented silently to the production of TPA computers, so, after cloning the PDP-8, every PDP computer was copied and reproduced at KFKI (Fig 10.5). The Institute became an official member of the DEC computer users group, *DECUS.* The last copied DEC model was the *Model Microvax 3000*. Only one KFKI model was not a copied one, the TPA 70 (Fig 10.6), it was the design of *János Bogdány* (Fig 10.7). The TPA 70 was used by other institutions in their computer controlled equipment too (Fig 10.8).

Fig. 10.6. TPA 70

After the political and economic changes in 1989, the KFKI 'computer factory' closed down; they stopped the production of the TPA computers. After the fall of the Iron Curtain and at the start of the free Hungarian market economy the original DEC computers were delivered to Hungary; the local DEC experts were available to assist. The Hungarian market was well equipped to use the original DEC computers thanks to the reliable TPA machines. DEC established its Hungarian subsidiary, mainly from ex-KFKI employees, and became soon the most successful Hungarian computer company.

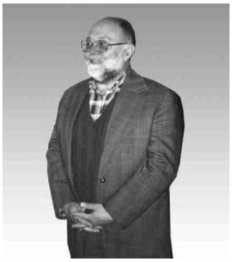


Fig. 10.7. János Bogdány, the constructor of the TPA 70

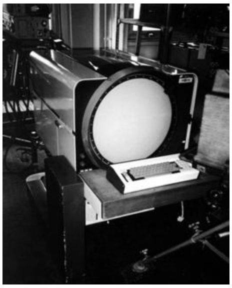


Fig. 10.8. The MTA SzTAKI, a GD 70 graphical display controlled by a TPA 70

11 Dr Náray Zsolt (1927-1995)

He was one of my favourite managers and a very good friend of mine. He graduated at the University of Technology (Magyar Királyi József Nádor Műszaki és Gazdaságtudományi Egyetem) in 1949 as a mechanical engineer. He worked as a physicist in the Physics Institute of the Budapest University of Technology (Fig 11.1).

Fig. 11.1. Dr Zsolt Náray

He joined the Central Physics Research Institute (KFKI) as a scientific research fellow and deputy head of the Cosmic Radiation Department in 1952. Then he was granted a scholarship in France in 1958-59. In 1959 he was appointed as the head of the Physics and Optics Research Laboratory and the Electronic Division of KFKI. From 1963 to 1969 he was the deputy managing director of the Central Physics Research Institute (KFKI).

His special scientific interest was the physics of light and he researched in depth its dual wave-particle nature. He conducted other researches too. He was the first scientist, who introduced the laser technology in Hungary. He designed and installed the first ruby-laser device at KFKI. He was the initiator and the head of the multi-channel analyser programme and later the TPA programme.

He released KFKI's development results to the factories of the Hungarian industry. Several companies started manufacturing KFKI's equipment, e.g. EMG, GAMMA etc. This activity made good economic sense and provided a sound stream of income to the institute. He was appointed as the chief designer of the Hungarian Rjad Computer

(Fig 11.2) in 1969 and he became the chief executive officer of the new Co-ordination Institute of Computer Science (SzKI) in1969. This institute was responsible for the development of the Hungarian Rjad computer system and the management of the co-operation among the Socialist countries in computer development (Fig 11.3).

Fig. 11.2. The SzKI's R10

Fig. 11.3. Computers of the Unified System

Fig. 11.4. The SzKI headquater (Donáti utca)

Fig. 11.5. Dr Náray in an exhibition (with microphone)

SzKI was one of the research institutes (Fig 11.4), which developed not only the Rjad R-10 computer and its software, but also operated a computer centre running mainly Siemens mainframes where SzKI's own departments, other Hungarian institutions and companies conducted their own software development (Fig 11.5). The departments' activities were e.g.:

- image analysis and processing
- computer-product technologies
- design of the next Hungarian computer, R10 and R15, in the series: Rjad-2
- the first micro-computers, the M-05X and M-08X (Fig 11.6).
- SzKI developed and manufactured the first IBM compatible personal computers: PROPER 8 (Fig 11.7) and 16 (IBM XT and AT)
- The SzKI Siemens mainframe computer was the first computer with a time-sharing operating system in Hungary, and was working with a quite large terminal network connected to the computer.
- SzKI's software departments developed several types of software products, such as:
    - o civil engineering programs
    - o static calculations program for the building industry
    - o hospital management systems and
    - o software development for several foreign companies all over the world (Siemens, Triumph-Adler, Ericsson, banks, publishing companies, different factories, airports etc.)
  - The institute developed software tools (software engineering): e.g. Softing,
  - character recognition software: RECOGNITA
  - artificial intelligence language compiler: M-PROLOG
  - one of the first certified ADA compiler in the world
  - a software quality assurance tool: SOMIKA

Fig. 11.6. SzKI made M08X computers donated to the Budapest University of Technologies

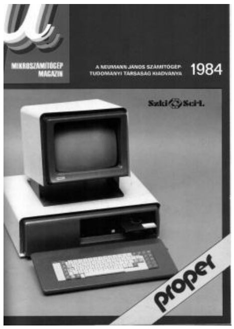


Fig. 11.7. The Proper 8 (IBM PC XT clone)

- production-control and data processing for different companies
- engineering construction sites support by computer (Fig 11.8)
- etc.

Fig. 11.8. Teletherm terminal developed by SzKI

Náray's great idea was to organise a large institute, where the different departments and the computer centre would work in close proximity. SzkI bought a large piece of land near the Buda castle, which was vacant because it was destroyed by bombing in WW II. In this area he built several buildings, where the different departments were working next to each other (Fig 11.9).

Unfortunately he was stood down in 1990 and his successor could not manage the institute as successfully as he did. He died in 1995 and the institute claimed bankruptcy. His spiritual heritage is still living in the loving memory of his more than one and half thousand colleagues (Fig 11.10).

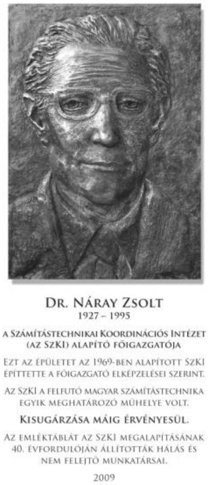


Fig. 11.10. Dr. Náray's Memorial Plaque (Donáti utca)

Fig. 11.9. The SzKI, Group of Application Laboratories (Iskola utca 10)

12 Mihály Kovács (1916-2006)

Fig. 12.1. Mihály Kovács as a military priest, 1941

It was usual for poor families in Hungary to have a lot of children. The older ones started working and earning money at young age. The eldest child supported the youngest one, who attended school and became an educated member of the family.

The family of Mihály Kovács was not rich, his father was a joiner and his mother was a dressmaker. The family had seven children, Mihály was the youngest one. His father died, when he was three years old. One sibling died at baby-age, his two older brothers became also joiners and his three sisters were trained dressmakers.

Mihály Kovács was an eminent student at the elementary school and therefore his mother enrolled him to the secondary school of the Jesuit order. His favourite subjects were mathematics and physics, but he was an enthusiastic traveller, mountain walker and a very good stenographer.

When graduating from secondary school he decided to be a teacher and a monk. He first continued his studies in the seminar of the Jesuit Order, then at the University of Sciences. He also studied sail making and was a category-A licenced pilot. He was consecrated as a priest in 1941 at the start of WW II in Hungary. He completed a military priest training in 1941 (Fig 12.1). He commenced teaching at the very secondary school in Szeged, where he had been a student. He relocated to the Jesuit Secondary School, called Piarist Gimnazium in Hungarian, Budapest as a teacher in the school.term 1943/44. He taught there until his death.

During WW II in 1944 Mihály Kovács and other three Jesuit teachers offered to go to Germany with the secondary and university students who were transported there by the German army. These young children were protected and taught by the priests in the military and later in English prison camps. At the end of the war the young Hungarian students and some other young soldiers returned home together with Mihály Kovács.

He continued his teachingr career, but the communist government in 1945/46 did not maintain a good relationship with church-run schools. Finally the government dissolved these schools and the teachers had to accept parish priest positions. Mihály Kovács also had to.

The government and the church started a dialogue about the nationalized schools in 1950 and as a result of it 8 schools were placed under church management again, the Piarist Gimnasium Budapest was one of these schools. Mihály Kovács was allowed to teach mathematics and physics again. The building that once was owned by the school was occupied by the Budapest University of Sciences. They had no choice, but starting the school-year in a new building, in a long-ago nunnery. The female order had been dissolved so the premises were available.

Fig. 12.2. The flotta of the Jesuist Gimnazium on the Balaton Lake

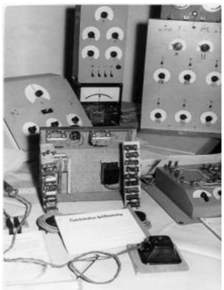


Fig. 12.3. Analog calculators, desig-ned by the students

The centre of Mihály Kovács's life were his students, he was fondly called 'Mr. Teacher Kovács' by his students. He organised all sorts of activities, e.g. he built sailing boats together with his students that were used at the Balaton-lake (Fig 12.2). He organised nuclear physics and physics study circles, where the students gave lectures and presented their own experiments in various subjects.

I first met him and his students, when we were developing the first Hungarian electronic computer at the Cybernetic Research Team in 1957/59. He gave me a call saying that he and his students would have liked to inspect our computer that was being built. Of course I was happy to show the computer, but I was very surprised that they knew a lot about computer technology. He told me his students were members of the Cybernetic Study Circle of the secondary school, where they designed and installed several cybernetic devices (Fig 12.3).

A few days later I visited the Cybernetic Study Circle at the school, where the students showed me their devices, working cybernetic models: e.g. the 'Logi card-playing machine' (Fig 12.4) with electro mechanical elements (*Zoltán Perjés)* 1960, the 'miracle mill', at first a simple version *(Jenő and Zoltán Ágost),* later on an automatic version *(Zoltán Perjés and György Vesztergombi)* 1961, the 'Shanon mouse and the maze', *(György and Ferenc Vesztergombi, with István Káli)* 1963 (Fig 12.5) etc. New students – Zoltán Fodor and Tivadar Lohner – designed new cybernetics equipment of: a well-known mechanical toy transformed to relay circuits. Its name was: Kombinett eithts (Fig 12.6). These all devices were the ideas of the students, because they were not only students of their teacher, Mihály Kovács, but they were also his friends. He assisted them, when they built these machines (Fig 12.7).

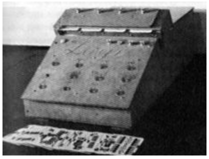


Fig. 12.4. The Logi, a card-playing machine

Fig. 12.5. The Shannon mouth

Fig. 12.6. Kombinett number eight

Fig. 12.7. A book by Mihály Kovács: *Cybernetic toys and models*

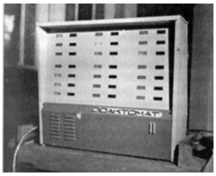


Fig. 12.8. The DIDACTOMAT

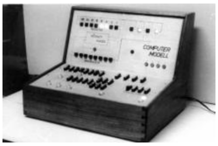


Fig. 12.9. The first computer model made by the students

Fig. 12.10. The students and the first computer of the school

The Cybernetic Study Circle developed modern tools for programmed teaching, too, the *DIDACTOMAT* (1964). (Fig 12.8) It was their teacher colleague's, *Lajos Terényi's*, idea, however the machine was built by the students. The DIDACTOMAT was able to check if the students in the classrooms really understood the lectures. Several DIDACTOMATs were used in the classrooms of the school. The school patented the DIDACTOMAT in 1966 and a company, the *Factory of School Equipment,* produced about 150-200 pieces from the machine.

The Cybernetic Study Circle had another famous device: the MICROMAT (1966). This tool supported the understanding of the concept of the computer. The prototype was built by a student: *Ferenc Woynarovits*. This piece of equipment was also produced by a *small cooperative.* The MICROMAT was a set of digital building elements that could be used for the compilation of about 100 different cybernetic games (Fig 12.9).

Mihály Kovács was very knowledgeable about the new computers. He and his students were regular visitors at the sites of these new computers (1960-1974). Sometimes they were given the opportunity to run their own programs.

When the age of the programmed calculators arrived (1974-79) some earlier students and foreign institutions supported the school and bought a number of programmed calculators for the students (Fig 12.10).

When the first microcomputers became available (after 1969), they started to use microcomputers. Their first real-life microcomputers were two TRS-80s. The students learned the BASIC programming language and wrote programs.

I recall that one of Mihály Kovács's students, *Zoltán Zsuffa,* was a participant in my Garay computer applications contest, where his software won the *first prize.* He analysed several Hungarian translations of the original Finnish national epic

'Kalevala'. His method was later acknowledged by many linguist scientists, when I showed them his work.

Fig. 12.11. A report with Mihály Kovács in the Microcomputer Magazine

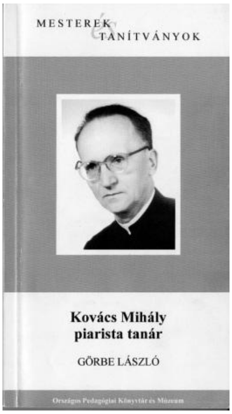


Fig. 12.13. The book of László Görbe: *Kovács Mihály, piarista tanár*

The number of computers in the school always increased, they bought e.g. Sinclair computers and the students were awarded computer prizes at various competitions, too. My computer journal, the *Microcomputer Magazine,* published 7 articles and reports with photos (Fig 12.11) about the achievements of the students of the Jesuit school (July 1986) (Fig 12.12).

The first IBM PCs (XT and AT) were donated to the school by CARITAS (Vienna), it was the real start of professional computers in the Jesuit secondary school (1988). Mihály Kovács was the first teacher in Hungarian secondary schools who studied computer programming and technology with his students. The use of the computers was free at the school and older students taught and instructed the younger ones (Fig 12.13).

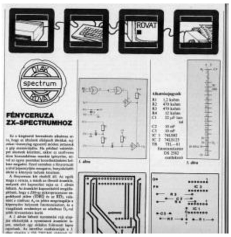


Fig. 12.12. A *light pen* construction of Csaba Káli, student of Mihály Kovács in the Microcomputer Magazine July 1986

Mihály Kovács became seriously ill in 1982 and retired in 1995, but he remained in touch with the order and his students. They frequently visited him. He gave occasional lectures occasionally at the school. He suffered a serious accident in December 2005 and he was unable to continue to support his beloved school and students. He died on 23 March 2006.

13 Marcell Jánosi (1931-2011)

Fig. 13.1. Marcell Jánosi

He was a talented precision mechanic. He started working in 1946. At the time of his university enrolment he was already qualified in joiner, welder, grinder and miller. He graduated as a mechanical engineer at the Budapest University of Technology Budapest in 1954. He made up his mind to become an excellent designer of mechanical instruments (Fig 13.1).

He started his career at the Safe Factory and then joined the technology development department of the Phone Factory. Later on he was employed by the Telecommunications Department of the Ministry of Metallurgy and Machinery. His main task was the establishment and elaboration of new technologies, e.g. printed circuits etc. He was discharged from the Ministry in

1957 and his next job was at the Budapest Factory of Radio Technology (BRG).

As a chief designer he recruited a very well trained designer team at the factory in 1960. While being employed at the factory he was also studying at the University of Economics and earned his second degree in 1965.

Jánosi designed and produced the first modern Hungarian made tape recorder that was marketed all over the country under the name CALYPSO (Fig 13.2). The factory produced and sold more than 100,000 devices. After such a resounding success he designed more than 30 different new tape recorders. The product lines of the tape recorder single product ensured steady revenue for the factory that was in excess of 150 billion HUF.

Fig. 13.2. The BRG Calypso tape recorder, designed by Marcell Jánosi

Fig. 13.3. The ABC 80 and the BRG Tape recorder

Fig. 13.4. The patent of the MC-1 Jánosi - cassette floppy

Fig. 13.5. The MCD-1 driver and floppy 2

Fig. 13.7. The MCD-1 in the Your Computer journal 1983, February

Fig. 13.9: The handing over ceremony of the von Neumann Society's life work prize by Gábor Péczeli President

Fig. 13.8. The MCD-1 cassette floppy

Fig. 13.6. The front-page of the Your Computer journal. February, 1983

His achievements were acknowledged and rewarded with a Hungarian State Prize of the second degree in 1975.

In 1970 he swapped his interest from tape recorder technology to computers. His first design was a piece of data preparation equipment which was using a cassette recorder instead of paper tape punch that was the usual peripheral device for this purpose at that time.

BRG manufactured the cassette recorders for several other computers, such as Commodore, Sinclair, Atari, Acorn and ABC 80. BRG and the Swedish company Luxor, the manufacturer of ABC 80 computers, signed a co-operation agreement.

They agreed that Luxor would use the BRD cassette recorders to their computers and BRG would manufacture the ABC 80 computers for the local market (Fig 13.3).

The success of the cassette recorder triggered Jánosi's new idea to design a 3" cassette-microfloppy as a more up-to-date replacement for the cassette recorder. It was the first cassette microfloppy in the world. Its name was: MCD-1 (Fig 13.4).

The computers were using 8" floppies at that time, the volume of this drive was 10 litres, MCD-1's volume was only 1 litre. The two floppy drives had the same capacity: 250 kBytes (Fig 13.5).

The management of BRG was not satisfied with the success of the MCD-1 cassette micro-floppy, because they preferred the development of a *magnetic bubble memory* that had never made it to be a product. Several managing directors of large foreign companies such as Commodore, Toshiba and Triumph visited Hungary in order to buy the patent and to attend business discussions with the director of BRG or with Jánosi (Fig 13.6). Unfortunately the management of the factory did not answer their inquiries and Jánosi was denied permission conduct negotiations (Fig 13.7). After all BRG did not renew the patent and it expired. Its competition commenced the production of the cassette micro-floppy. Hungary lost a big business opportunity and next year bought cassette floppies in the value of 1 billion HUF from Japan (Fig 13.8).

Marcell Jánosi's last design was a micro motor for the LEGO construction toys. He arranged the production of the motor with a small manufacturer. There was insufficient funding available for the joint venture and so the right of manufacturing the motor became the property of the LEGO Company. Their net yearly revenue from selling the motors was about 1 million USD annually.

Jánosi received a life-work prize from the John von Neumann Computer Society (Fig 13.9), and a university private professor title from the Óbuda University in 2010 (Fig 13.10). He died in 2004 (Fig 13.11).

Fig. 13.10. Jzsef Tik, vice principal of the Óbuda University gives the diploma of private university professor to Marcell Jánosi

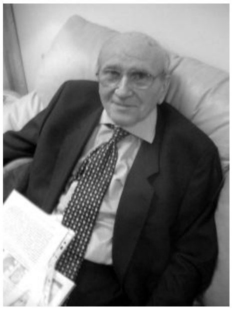


Fig. 13.11. Marcell Jánosi Titular university professor 2

References

Aspray, W.: John von Neumann and the Origins of Modern Computing (1990)

Burks, A.R.: Who Invented the Computer? (2003)

Goldstine, H.H.: The Computer from Pascal to von Neumann (1972)

Görbe, L.: Kovács Mihány piarista tanár. Mihály Kovács the Piarist Teacher (2007)

Ferenc, G.: Élhetetlen feltalálók, halhatatlan találmányok. Unpractical Inventors, Immortal Inventions (1985)

Juhász István hagyatéka. (The Legacy of István Juhász) (1994)

Kempelen, F.: The Mechanics of the Human Speech (1791) (Hung. Transl.)

Kovács, G.: 30 éves a magyar számítógép. 30 Years Old the Hungarian Computer, Exhibition in Nyíregyháza (1989)
Kovács, G., Tóth, E.: Az informatika története. The History of the Information Technolohy (1991)
Kovács, G.: Old Timers of Computer Technology. IOI (1996)
Kovács, G.: Neumann János. John von Neumann (1997)
Kovács, G.: Válogatott kalandozásaim Informatikában. My Selected Adventures in Information Technology (2002)
Kovács, G.: Ki volt igazából Neumann János? Who was really John von Neumann? (2003)
Kovács, G.: 40 éves a magyar számítástechnika és a Volán Elektronika. 40 Years Old the Hungarian Computer Science and the Volan Elektronika (1999)
Kovács, G., Gyula, S.: Neumann János emlékkönyv. A Book Published in Honour of John von Neumann (2003)
Kovács, G.: The Life and the Scientific Results of the World-famous Hungarian Born Mathematician. The Pioneer of Computer-Science (2004)
Kovács, G.: Epizódok az ős-számítástechnika történetéből. Episodes from the History of the Ancient Computer Science
Kovács, G.: 50 Years Ago We Constructed the First Hungarian Tube Computer the M-3 (2010)
Kovács, G.: Neumann János emlékkiállítás, egy géniusz ifjúkora. Memorial-exhibition of John von Neumann. The Young Years of a Genius (2011)
Kovács, M.: Atomfizika, kibernetika és számítástechnika budapesti iskolánkban. Nuclear Physics, Cybernetics and Computer Science in our Budapest' School (1993)
Kozma, L.: Egy Kossuth díjas börtönévei. Years in the Prison by a Kossuth Prize Winner (2001)
Kőszegi-Pap: Kempelen Farkas (1955)
Marx, G.: The Voice of the MARTIANS (1997)
Nagy, V.: Orsós magnó gyártása. The Production of the Reel Tape Recorder (2009)
Náray Zsolt emléktábla avatás. Unveiling of the Zsolt Náray's Memorial Plaque (2009)
Számítástechnikai Kislexikon (Small Lexicon of Computer Science) (1973)
Volt egyszer egy SzKI...(It was Once a SzKI...) (2011)

Information Technology in Italy: The Origins and the Early Years (1954 - 1965)

Corrado Bonfanti

Associazione Italiana per l'Informatica ed il Calcolo Automatico (AICA), Italy
corradobonfanti@hotmail.com

Abstract. Narration will be mainly focused on the origin, course and aftermath of four far-reaching initiatives that bloomed in Italy at almost the same time, in a few months encompassing 1954 and 1955; a choice which implies unfair omission of other worthwhile but less influential happenings.

Keywords: Italian computers, information technology, computer relics.

1 Introduction: 'Make' or 'Buy'

Until 1954, few Italian mathematicians and engineers – no more than a dozen – enjoyed the opportunity of some insight and acquaintance with stored-program electronic computers, on the occasion of study missions at the most renowned pioneering installations in England and especially in the United States; but none of such technological marvels was still running in Italy.

Two different approaches – 'make' or 'buy' – were viable to fill the gap and both were actually pursued[1].

The 'buy' Approach

- The Polytechnic University of Milano acquired, in USA, a CRC 102-A computer to equip the just then set up *Centro di Calcoli Numerici*. The machine was running by October 1954 and the Centre officially opened some months later.
- The *Consiglio Nazionale delle Ricerche* (CNR) bought in England a Ferranti Mark I* for the *Istituto Nazionale per le Applicazioni del Calcolo* (INAC). The FINAC (Ferranti-INAC) machine arrived at Roma in November 1954 and completed acceptance test on June 1955.

The 'make' Approach

- The University of Pisa established its *Centro Studi Calcolatrici Elettroniche* (CSCE) with the aim of designing and building the *Calcolatrice Elettronica Pisana* (CEP) scientific computer. CSCE activities started at the end of 1954.

[1] Ref. [1], [2].

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 320–347, 2012.

- The internationally renowned firm Olivetti decided to enter the emerging computer industry. The *Laboratorio Ricerche Elettroniche* (LRE) was then established in order to design and prototype the Elea commercial mainframe. R&D activities[2] begun at the middle of 1955.

The already mentioned simultaneity of the four initiatives – a circumstance that did not imply any underlying overall strategy – clearly indicates that the urgency to enter the computer era was widely perceived throughout the country. As a very fruitful consequence, the efforts went through distinct approaches which complemented each other, avoiding competitive or conflicting situations: although resting on mutually independent scopes and resources, several kinds of collaboration had been set up since the beginnings. As a matter of fact, computing centers at Milano and Roma together with CSCE and Olivetti laboratories became the incubators for the first generation of Italian informaticians[3].

The Polytechnic of Milano and INAC in Roma embraced the 'buy' approach in order to quickly have available an up-to-date powerful tool to afford computing applications that arose from the most varied needs of industry, engineering, physics, economics and mathematics itself; such was, after all, their institutional mission.

Worthy of mention that both acquisitions were almost entirely supported by ERP funds[4]. An even more important common trait of the two initiatives appears when appreciating what 'to buy a computer' really meant in 1954-55: far from being a ready-to-use product, the computer required in-house hardware maintenance and upgrades, often implying design and construction of original supplementary features.

Moreover, the machines were shipped 'nude', i.e. lacking of any software aid. Software development thus became the most impelling and demanding concern: service and utility programs (the forerunners of Operating Systems); algorithms for vectors and arrays handling; libraries of subroutines for frequently recurring problems like mathematical functions calculation. Until users themselves built up their loaders-interpreters and assembler-like symbolic languages, software had to be coded in machine language.

All these hardware and software tasks and duties concerned in a first moment the specifically owned machine but soon evolved into a generalized approach to every branch of computer science.

Final results of the 'make' approach, necessarily lasted several years even if the intermediate phases were full of scientific, technological and educational achievements.

---

[2] *Centro di Calcoli Numerici*: Numerical Computation Centre. / *Consiglio Nazionale delle Ricerche*: National Research Council. / *Istituto Nazionale per le Applicazioni del Calcolo*: National Institute for Applied Calculus. / *Centro Studi Calcolatrici Elettroniche*: Study Centre for Electronic Computers. / *Laboratorio Ricerche Elettroniche*: Electronic Research Laboratory.

[3] Let's at least mention other seminal initiatives. The Institute of Cybernetics at Napoli, founded by the theoretical physicist Eduardo Caianiello, mainly concerned with artificial and human neural networks from the Artificial Intelligence point of view [3]. DDA (Digital Differential Analyzer) installed at universities of Napoli, Bologna and Torino and the research that arose around them [4]. A small but effective enough electronic computer built at the University of Padova by Francesco Piva and others, expressly designed for teaching purposes [5].

[4] ERP: European Recovery Program, better known as 'Marshall Plan'. Apart from its political meaning within the cold war climate, ERP revealed highly effective in redeeming Western Europe from the disasters of the war.

The CSCE project at Pisa originated from rather fortuitous (as well as fortunate!) circumstances and culminated with the construction of the CEP scientific computer. While completion of the CEP lasted until December 1960, a Reduced Machine (*Macchina Ridotta*) was ready as early as 1957.

Olivetti's LRE was able to complete in 1957 two prototypes based on vacuum tube technology. The final product was a fully transistorized mainframe: the Elea 9003. The first machine was delivered to a customer in 1959.

The time elapsed between start-up and successful outcome of the 'make' projects revealed of the same order of analogous projects carried on by most advanced Countries. A fact which deserve appreciation because – as far as technological and theoretical indigenous know-how were concerned – they started from almost *tabula rasa*.

To give but some examples, it is widely known that American and English computer projects massively recruited former military radar engineers, notably acquainted with non-linear (i.e. 'pulse' or 'digital') electronics, the basic technology of computer circuitry[5]. Italy instead almost completely lacked such already skilled personnel, due to the blindness of military staff who, during the war, denied resources to radar development: only a handful of single scholars were then familiar with digital electronics (some of them will be mentioned later on). For the same reason, domestic industries did not receive any stimulus to produce components needed by the 'new' electronics, namely special quality vacuum and cathode-ray tubes.

A further drawback could be found in the lack of knowledge about the relatively recent results in theory of computability and mathematical or symbolic logic: revolutionary findings by Gödel, Church, Turing, Post, Carnap and others began to affect Italian academy only in the post-war years and lasted even more to be appreciated as founding principles of theoretical computer science[6]. This was but a consequence of a long lasting supremacy of the idealistic culture that caused, among other, decline and fall of the Giuseppe Peano's school that flourished and gained international reputation at the outset of twentieth century[7].

---

[5] To be reminded that 'linear electronics' originated from the invention of the triode vacuum tube (Lee DeForest, 1907). Its main concern was the undistorted (i.e. 'linear') amplification of continuous electric signals; it was then the core technology for radio communications.

[6] A single but outstanding exception was Corrado Böhm, whose brilliant career in the logic of computation began with his 1952 doctoral dissertation given at the Polytechnic of Zürich – with E. Stiefel and P. Bernays as rapporteurs – concerning the first compiler ever written in its own language; the dissertation appeared in [6]. He emigrated to Switzerland to escape fascist racial laws against the Jewish and returned to Italy (1953) as an INAC researcher. During the formative years in Switzerland he became familiar with IBM and Bull punch-card equipment and, most of all, with Konrad Zuse's Z4 relay computer and its *Plankalkül* programming language. We shall meet Böhm on several passages of this narration, but here is the occasion to anticipate the celebrated 1966 Böhm-Jacpini theorem [7] that assured a theoretical groundwork for the 'Go To-less crusade' by Edsger Dijkstra and subsequent Structured Programming methodologies.

[7] During the first half of twentieth century, Benedetto Croce has been the undisputed champion of idealistic philosophy, according to which, roughly speaking, conceptual and hard sciences – from mathematics to technology – were to be confined in the realm of instrumental activities, useful to mankind but deprived of cultural dignity; an approach that deeply biased public opinion and every degree of education. Croce particularly fought against the mathematisation of logic: logic, in his mind, had to be a hunting preserve for qualitative philosophical speculation [8].

2 Polytechnic of Milano

The Polytechnic of Milano was – and still now-days remains – the most renowned engineering university throughout the Country. It was necessarily concerned with numerical calculations associated to the most varied engineering problems. From its long lasting tradition in this domain, let's select but a few examples[8].

By 1928, Gino Cassinis – then at the beginning of his academic career – published a substantial seven hundred pages treatise on 'Numerical, Graphical and Mechanical Calculations'; Ercole Bottani, in 1935, issued an article titled 'Mathematics as seen by an Engineer', a sort of engineer's bible to approach mathematics; the same Bottani – very likely unaware of Bush's Differential Analyzer – extensively experimented with physical systems which could be converted into analogic calculators, notably the 'Electrolytic Vessel': in 1945, under CNR's auspices, he established and directed a *Centro Studi per i Modelli Elettrici*.

Not surprisingly Gino Cassinis – elected in the meantime as the rector of the Polytechnic (and as the Mayor of Milano as well) – issued in 1951 a request to buy an electronic stored-program digital computer, to be supported with ERP funds. The request was accepted and, upon a screening of the not numerous machines made commercially available by the then emerging USA computer industry, a CRC 102-A model was chosen[9]. Not a huge machine, it was very cleverly engineered and equipped with a relatively cheap magnetic drum main memory; price/performance rate resulted highly competitive[10].

Once the contract was signed, a young member of the Polytechnic was charged to carry-on the project: he was that Luigi Dadda who was to become one of the leading and driving personality for the development of informatics in Italy[11]. Dadda immediately left for Los Angeles, where he spent about four months at the CRC factory. He participated intensively in the making and testing of the machine, so gaining a full mastery of circuitry as well as of the new 'art' of programming. The computer arrived at Milano in October 1954; the installation resulted in a minor concern due to the compactness of the cleverly engineered machine that was shipped in fully finished units. The expressly constituted Numerical Computation Centre immediately started operations, while Dadda's first care was the training of colleagues (the first

---

[8] Ref. [9].

[9] The evaluation committee was composed by Cassinis, Bottani and Luigi Amerio, a mathematician formerly at Picone's INAC institute. Final screening benefited of knowledgeable suggestions by Samuel N. Alexander, the responsible of SEAC project at the National Bureau of Standards; he rendered himself friendly available while attending an international meeting at Milano.

[10] CRC (Computer Research Corporation) was born as a spinoff from Northrop Aircraft Corporation. Just in that period CRC was acquired by RCA (Radio Corporation of America) in order to establish its own entry into the computer industry.

[11] Dadda had worked with Bottani and, more important, was in the number of the few to have some previous experience with pulse electronics: the subject of his doctoral thesis (of 1945) concerned in fact a long-range pulsed radio link, a project he carried on under the supervision of Francesco Vecchiacchi, a distinguished professor of Electrical Communications. Among Vecchiacchi's achievements, the first statement (1939) of a formal theory for the bistable multivibrator (the popular 'flip-flop' electronic circuit) is perhaps the most commonly recognized [10].

programmers) and hardware technicians. After being housed in better suited premises, the Centre was officially opened a year later[12].

We have already considered the severe hardware and software tasks that were implied when buying a ‘nude’ machine in the years of infancy of electronic computing. All of them obviously occurred at the Polytechnic. Let’s mention but the case of arithmetic capabilities of CRC 102-A: the machine was natively able to operate only on integer numbers while floating-point arithmetic was a mandatory need. Development of reusable software algorithms to perform ‘floating-point by integer’ arithmetic initially helped to overcome the problem, even at the cost of considerable overhead. An original hardware – activated by newly implemented commands – was then promptly designed and built to directly exploit floating-point arithmetic. The capacity of the drum memory was doubled by a new design of the read/write heads. Also the I/O performance of the machine – natively limited to a teletype – was enhanced by a hardware interface connecting a fast photoelectric punched-tape reader. CRC 102-A productive life lasted about eight years, until Olivetti gifted the Polytechnic with an Elea 6001 produced by the same firm.

Details of the manifold research activities carried on by the Centro di Calcoli Numerici as well as an enumeration of industries and scientific institutes who submitted problems can be found on several historical accounts and personal recollections by Dadda[13].

Anyway, there is no doubt that the most enduring contribution of the Centre has been to intensively lend its personnel to teaching duties: single courses on computer-related disciplines started immediately and culminated into a formally stated specialization in Electronic Engineering (1960). Polytechnic of Milano fostered and licensed over the years a lot of highly skilled computer engineers; many of them undertook academic careers throughout the country, others went to industry and a considerable number worked lengthy abroad or migrated definitively.

3 INAC, ICC, FINAC, CINAC

The early story of INAC is intrinsically bound to the mathematician Mauro Picone, an outstanding character of this account.

Aged 29, he won a competition to fill a post of university full professor, but the outbreak of war (1914) caused enlistment as a second lieutenant and prevented him having a chair assigned. The ordnance staff became soon aware of the inadequacy of then available fire tables for heavy gunnery[14]. The urgent task of getting new tables ready was assigned to Picone. He worked intensively days and nights at a numerical solution of the quite intricate differential equations of exterior ballistic, only equipped with a small Brunsviga-like mechanical calculator. The new set of tables revealed timely and effective enough to get the author promoted at the rank of captain. Picone

[12] Later on (1958) the Centre was restructured on the base of two departments: Mathematical Section (Amerio) and Electronic Section (Dadda).

[13] Ref [10], [11].

[14] Available tables were in fact prepared for lowland operations, while the Italian war front implied a need to shoot in the middle of Alpine mountains, often with a considerable difference of altitude between guns and targets.

himself repeatedly quoted this war experience as the origin of his concern with numerical problem-solving and computing machinery.

By the end of the war, he taught analysis and calculus in several universities until arriving at Napoli, where, thanks to good offices of a colleague and a friend of him – the economist Luigi Amoroso – the local bank in 1927 granted funds to establish an *Istituto di Calcolo* within the Faculty of mathematics; Picone's approach towards 'concrete' mathematics got to a visible institutional status. Upon moving in 1932 to the University of Roma – his definitive stay – he successfully fought to get the Laboratory moved with him and rose to the rank of CNR Institute: the already mentioned INAC[15].

The number of mathematical consultancies requested of and fulfilled by INAC is really impressive: we know from official records that they were 1,492 between 1937 and 1964, i.e. more or less 5 per month. It is to be noted that INAC permanent staff always remained very limited, but over the years it was complemented by a huge number of temporary collaborators and visiting members, often from abroad.

Activities at INAC were far from resting on mere computing tasks: it was instead common practice that problems submitted to the Institute implied refinement of already known mathematics and, often enough, development of brand-new theories and methods[16]. This explains the number and the quality of scientific papers and reports issued by the Institute, often appearing on international journals and proceedings too: starting from early years at Napoli, they were 163 in the 1927-36 period and 663 followed up to 1968. Picone authored or co-authored about 130 of them.

INAC was obviously equipped with standard mechanical calculators and other simple aids to computation but Picone willfully looked about for any kind of machinery useful to increase computing power and to speed up computation. We shall mention but a rough selection of his efforts that culminated in a sort of 'rush to the computer'[17].

During the Thirties and early Forties, for instance, he was in touch with the already mentioned Bottani of Milano and came across equation solvers built – or merely devised – by Mallock in England, by Lorenzo Poggi and Lamberto Cesari at the University of Pisa and by Alessandro Boni, one of the INAC consultants[18].

---

15 Carrying on his initiative, Picone had to really fight against a long lasting tradition according to which academic research and teaching should be restricted to 'pure' mathematics. It is worthwhile mentioning that the authoritative mathematician Francesco Severi felt the need to establish in 1939 his own Institute of Higher Mathematics (Indam – *Istituto Nazionale di Alta Matematica*)), in order to emphasize the distance between higher (i.e. pure) mathematics and the computing practices of applied mathematics as well as the implicit – but fallacious – superiority of the former.

16 Moreover, it was mandatory for INAC to supplement numerical solutions with a careful evaluation of their reliability. Elaborate methods were then developed in order not only to reduce computing errors but also to certify an upper bound to the error that affected results.

17 We owe to Angelo Guerraggio, Maurizio Mattaliano and Pietro Nastasi a quite extensive account of Picone's rush to computer. Their excellent writings [12] and [13] are based on the huge documentation that is available at the IAC Historical Archive, in Roma.

18 The most recurring computing task at INAC was the solution of systems of linear algebraic equations, and the number of equations and variables was often so high to overwhelm capacity and accuracy of such otherwise ingenious contrivances.

As soon as the first notices about automatic computers arrived in Italy, Picone was dazzled: the miracle he expected seemed at hand![19] He immediately realized that, lacking of such a new resource, INAC computing capabilities and its same internationally renowned prestige incurred the risk of obsolescence: computing centers abroad and institutes in which an electronic computer could be available were fated to supremacy. Picone soon activated two lines of attack:

- to send study missions to USA, England and wherever appropriate in order to gain first hand insight into the rapidly evolving state of the art;
- to raise funds and partnerships to build an Italian computer.

Bruno de Finetti – a distinguished mathematician then acting as a CNR consultant – in 1950 visited coast-to-coast almost every significant US computer initiative and wrote an extensive account of the state of the art[20]. The same Picone together with his colleague Gaetano Fichera went to the United States for a quick survey. Other missions involved:

- Enzo Aparo and Dino Dainelli (they spent several weeks in Washington–DC at the Seac site, with the opportunity of coding and successfully running some mathematical program under the extremely kind supervision of Ida Rhodes);
- Michele Canepa (he was Olivetti engineer in charge of assisting Picone in view of a devised joint effort for an Italian computer; he was hosted for about eighteen month at Aiken's Harvard laboratory being there responsible for the design an set-up of a subsystem of the Mark IV computer);
- Giulio Rodinò and Mario Salvadori (who lengthily acted as Picone's watchers and 'ambassadors' across the States).

De Finetti, Kitz, and Rodinò went also to Teddington (London) where on March 1953 they attended an international symposium on automatic digital computing and wrote a careful report about it[21].

Raising funds and/or partnerships for a domestic computer soon appeared a most difficult task. Preliminary talks with domestic industries – namely Olivetti and Microlambda – were soon abandoned[22].

The proposal to make at INAC a clone of Harvard Mark IV – Picone suggested for it the name of Mark V – vanished as well, in spite of early friendly touches with Aiken, who also paid a visit to INAC.

For about six months INAC benefited of punch-card machines on free loan by IBM. The agreement was fostered by de Finetti, thanks to his long lasting familiarity with the managers of IBM's Italian branch. During those months de Finetti introduced

---

[19] Picone found an early account in a 1944 issue of *Stars and Stripes*, the popular magazine published by the U.S. occupation forces in Europe. Emphasis was on Howard Aiken's electromechanical Mark I, the IBM-Harvard University joint project. Later on, he became also aware of the ENIAC electronic computer. To be reminded that, although being technological prodigies, both machines were still not stored-program (*à la* von Neumann) computers.

[20] De Finetti's report [14] represented the early source of reliable and detailed information ever made available to the Italian scientific community.

[21] Ref. [15].

[22] We shall deal later on with Olivetti's ventures. Microlambda – established in 1951 near Napoli and leaded by Carlo Calosi – specialized in electronics and his flag product were radar systems produced under Raytheon licence. Microlambda later merged into Selenia, so becoming part of IRI (*Istituto per la Ricostruzione Industriale*: the huge and heterogeneous financial holding owned by the Government).

Picone and his staff to the secrets of punch-card machinery, uncovering its unsuspected ability to deal with mathematical problems, far beyond customary business applications. According to the gentleman's agreement, free loan had to be followed by the rental of a more powerful IBM CPC (Card Programmed Calculator) but the fee exceeded INAC budget and the loan simply came to an end.

Meantime, an unexpected event occurred which captured Picone's energies: it happened that UNESCO – United Nations Educational Scientific and Cultural Organization – promoted a consortium of nations to establish an ICC – International Computation Centre – whose site was to be placed somewhere in Western Europe.

Several Countries issued their application to host ICC and the most convincing came from Switzerland, the Netherlands and Italy. All of them stressed the existence of institutes deeply concerned with computing: ETH at Zürich, Mathematisch Centrum at Amsterdam and INAC at Roma[23].

Picone's will to win the competition became almost obsessive: he firmly thought that not only INAC's reputation but the national pride itself were implied. During 1950 and '51, besides soliciting solidarity by the estimators of his Institute, he urged the Ministry of foreign affairs and the Italian representatives at UNECSO with a stream of memoranda[24]. The Swiss retreated, and UNECSO diplomacy had to choose between Amsterdam and Roma. In order to escape lengthy and delicate negotiations, they decided to seek the authoritative and *super partes* arbitration by Hermann H. Goldstine, whose final advice (November 1951) privileged the Italian proposal[25].

But it soon appeared that Picone had fought for failure. Considering the international importance of ICC, he had taken for granted that the Institute would be promptly equipped with the most updated computing machinery, easily accessible by INAC. Things went quite differently: big countries such as USA and U.K. did not subscribe to the Convention and ICC lived in a sort of interim limb until the statutory quorum of ten member nations was reached about ten years later[26].

---

[23] ETH: *Eidgenossiche Technische Hochschule* (Federal Polytechnic).

[24] In a December 1950 memorandum that Picone addressed to the Minister, he emphasized the strategic value that ICC computing facilities could have in support of NATO (North Atlantic Treaty Organization) armed forces; in case of a sudden attack by Eastern enemy – he argued – Amsterdam was possibly fated to invasion while Roma remained strongly sheltered behind the front. In Picone's mind, such a really strange and *ad hoc* speculation risked gaining acceptance on the ground of notoriously massive use of computers by USA military staffs as well as of the fact that, just six month before, cold war degenerated into Korean 'hot' war.

[25] Herman Heine Goldstine – a mathematician of IAS (Institute for Advanced Study) at Princeton – was among the American computer pioneers: he participated to the ENIAC project and had been principal collaborator of John von Neumann for the IAS Machine project. Chapter 7 of his book [16] under the title 'The computer and UNESCO' contains details of the early story of ICC, up to its assignment to Roma. Goldstine was well aware that the Dutch Mathematisch Centrum was already familiar with computer theory and practice: as early as 1948-51 they built a relay machine called Arra and were currently working at a new version with vacuum tube and magnetic drum technology. (Incidentally, a similar situation existed at Eth, where Konrad Zuse's Z4 electromechanical machine was running since 1950 and the Hermet project – showing the influences both of Zuse's Z4 and Aiken's Mark IV – was under way). INAC lacked of similar activities but the undisputable superiority of the Institute prevailed in Goldstine's mind as far as mathematical research and expertise in numerical methods were concerned.

[26] On the occasion, the firm Olivetti gifted ICC with one of the first shipped Elea 6001 scientific computer. Later on ICC became IBI (Intergovernmental Bureau for Informatics), but its aim and scope progressively failed to fit the state of art and UNESCO, in 1988, deliberated its dismissal.

The lengthy struggle came to an end when INAC got the opportunity to buy with ERP funds a commercially available computer and Picone, pressed by urgency, unwillingly abandoned the idea of an Italian machine[27]. A powerful Ferranti Mark I* was selected and the contract readily signed[28]. Two INAC mathematicians – Corrado Böhm and Enzo Aparo – went to Ferranti works at Moston (Manchester) in order to learn programming. They also ascertained the machine's capabilities by coding and running a program for the resolution of a system of 62 linear algebraic equations with as many variables.

The Ferranti-INAC computer, soon christened FINAC, arrived at Roma in January 1955 and became ready for acceptance test in June 1955. In contrast with the CRC machine at Milano, Mark I* was a huge one, shipped piece by piece – small subassemblies or even single components – so that its overall assembly resulted a not at all trivial concern and required several months. Another difference with the experience of the Polytechnic was that INAC personnel currently included only mathematicians, peoples not acquainted with oscilloscopes, vacuum tubes and electric welders; the Institute hired then in a hurry some young engineers to be trained by Ferranti technicians in the course of assembly[29]. The opening ceremony (December 1955) was a memorable event, held before the President of the Republic; Sir Vincent de Ferranti attended too and didn't omit to emphasize the Italian origin of his ancestors.

As expected, activities at INAC – as well as at the closely linked Faculty of mathematics – were deeply affected by the availability of a powerful computer. Things went about the same as happened at the Polytechnic of Milano and we will not to return either to the already discussed tasks and duties that occurred when buying a 'nude' machine nor on the steps that marked the rise of a prolific school for computer scientists, researchers and educators.

---

[27] The extreme effort had been the proposal to hire Samuel Alexander as project leader for a 'made in Italy' computer – obviously a SEAC-like machine – to be built with still granted ERP funds.

[28] At an early stage of negotiation, Ferranti Ltd. was extremely doubtful – not to say worst – about the reliability of Italian counterpart. Let's quote from a letter by B.V. Bowden (Ferranti marketing manager) to Lord Halsbury (the chief of NRDC: National Research Development Corporation). The letter is dated 21st April, 1953.

> *"I am not at all sure in my own mind if their ideas as to how it [a computer] is to be used are any clearer than were those of that lamentable Committee …"*
>
> *"A friend of us …, who spent some time in Rome, gave me to understand that negotiations of this kind are conducted in Italy on a rather unusual way which involves a fair amount of what one had perhaps best describe as 'rather delicate personal negotiations'."*
>
> *"… we are beginning to wander whether they are merely collecting a few quotations in order the better to give the final contract to a firm owned by some relation of one of the Ministers who is more interested in keeping the 1/2million pounds in Italy than in producing a computing machine."*

In 1990, Jeoffry Tweedale – who was then the curator of the National Archive for the History of Computing at the University of Manchester – came across this letter among the NRDC papers and kindly passed to the author together with other documents of same concern. The letter has been entirely reproduced in [17].

[29] The first engineers hired by INAC were Giorgio Sacerdoti – who had just completed a doctoral thesis on computer electronics, perhaps the earliest in Italy – Paolo Ercoli and Roberto Vacca. Ercoli became the most active mover of INAC engineering activities while Sacerdoti soon joined Olivetti's Elea project.

According to Paolo Ercoli, "*Maintenance and modifications of the FINAC were carried on by the Institute, as was customary in those years, so that machines that were identical when bought could be changed progressively and have, after a few years, rather different capabilities*."[30] Among several hardware improvements, the most effective had been new circuitry for overflow automatic detection in arithmetic operations and a feature to handle double precision arithmetic[31].

On the side of software, Corrado Böhm, Dietrich Prinz and others developed the Intint symbolic language – together with its own interpreter – which almost eliminated the need of machine-language programming.

During the second half of the fifties several members of INAC taught the first courses on computer programming and computer science at the Faculties of mathematics, physics and engineering of the University of Roma.

In 1960 Picone retired and Aldo Ghizzetti succeeded as INAC director; in May 1975, on the occasion of his 90th birthday, the Institute was dedicated to its founder with the new – and nowadays persisting – denomination IAC 'Mauro Picone'[32].

FINAC revealed an exceptionally long-lived machine: switch off occurred on June 1967. However, its unavoidable obsolescence became clear much earlier. When the University of Roma bought a general purpose Univac 1108 machine, powerful enough to fulfill administrative and scientific needs of all faculties and institutes, it seemed that FINAC would be superfluous.

INAC disagreed and resumed the idea of setting up an original computer; its director Aldo Ghizzetti, in 1961, addressed then a proposal to Olivetti, which promptly assured a quite robust collaboration.

A joint project was then agreed, according to which hardware implementation and set up was Olivetti's concern: the outcome of the project had to be a full working prototype owned by INAC – the CINAC (Computer INAC) – while Olivetti had to assume the prototype as the ground for an innovative industrial product – the devised Elea 9004 series. The outstanding feature of CINAC / Elea 9004 consisted in its being a 'stack' machine with zero-address instructions, the same architecture that inspired the almost contemporary Burroughs B5500 computer. The joint initiative was an unusually 'software driven' project, meaning that it arose from the convergence of Olivetti and INAC ideas about a machine particularly suited for efficient compilation of programs written in some Algol-like language[33].

Construction of the machine started at Olivetti works in early 1963 and culminated in February 1966 with CINAC delivery, but the seemingly successful outcome of the project is misleading. On Olivetti's side of the story – on which we shall return hereafter – the DEO (*Divisione Elettronica Olivetti*) was sold to General Electric and the

---

[30] Ref. [18].

[31] FINAC's native capacity to operate with 40 bit binary numbers increased to 80. This was in accordance with the quite peculiar nature of INAC's computations that privileged precise (i.e. with as many as possible digits) rather than floating-point arithmetic.

[32] The 'N' (for *Nazionale*) seemed no longer appropriate due to the spread of scientific computation centers that occurred in the meantime.

[33] Inside Olivetti, it was Mauro Pacelli – the responsible of LRE software department – to issue an early proposal. At the same time, the INAC logicians – among them Giuseppe Jacopini and very likely the same Corrado Böhm – cultivated similar ideas. Paolo Ercoli, on the side of INAC, became the most concerned with the development of the project.

devised Elea 9004 series was abandoned because it didn't match the product policy of the Americans; GE did not discontinue collaboration on CINAC but reduced resources to the minimum, whence a notable delay in hardware shipment. INAC, on its own, faced the problem of transferring to the new machine the huge wealth of FINAC software accumulated over the years. The hypothesis of coding it anew was unfeasible, so that they went to a tradeoff: a sort of hardware emulator of FINAC on the CINAC was set up in a hurry. At the same time, due to the lack of resources, the development of interpreters or compilers specifically designed for CINAC never went to satisfactory results. As a result, the innovative and powerful capabilities of CINAC were left hidden and almost unused. Soon after (1969) the final act occurred: the mathematician Guido Stampacchia succeeded Ghizzetti as INAC director and decreed the end of its hardware and software activities[34].

4 CSCE, CEP and the Pisa District

During the Forties of last century, elementary particle accelerators – huge and costly machines of ever growing dimensions that inaugurated in USA the age of 'Big Science' – became the most powerful tool for the advancement of experimental particle physics and outdated the cosmic rays approach that was still alive at Pisa[35]. The construction of a quite advanced accelerator thus became a major concern of the *Istituto Nazionale per la Fisica Nucleare* (INFN), which issued a call for tenders[36].

Several local governments for whose territories Pisa was the cultural centre of attraction, had created a consortium in order to give financial support to the University for some prestigious scientific undertaking: the particle accelerator seemed an extremely propitious occasion. Unfortunately for Pisa, the University of Roma was able to reinforce its own application with more than twice the financial support; the accelerator thus became the 1 GeV elecrtrosynchrotron at the INFN laboratories of Frascati, in the surroundings of the capital city[37].

At Pisa they worked to guess a new destination for already gathered funds and the answer – that had nothing to do with the University's scientific tradition – arose as a fortuitous consequence of the accelerator affair: during August 1954 Marcello Conversi, Giorgio Salvini and Gilberto Bernardini – all of them concerned with the project of the particle accelerator – met the Nobel laureate Enrico Fermi at a Summer School in Physics and, while relaxing in conversation, asked for his advice. He

[34] An account of what happened appears in chapter 7 (pp. 417-420) of [22].

[35] The University of Pisa had a strong tradition in the domain of elementary particle physics; it dated back to the 'school' of Bruno Rossi, who, during the Thirties, pioneered the use of cosmic rays as an inexpensive – but rather erratic – source of particles to be submitted to experiment. Worthwhile mentioning that his experimental apparatus included innovative 'coincidence' and 'anticoincidence' electronic circuits that, in forthcoming computer jargon, became popular as AND and XOR logic gates. [23] The young physicist Marcello Conversi – we shall meet him hereafter – was a great expert of such techniques that belong to non-linear electronics.

[36] *Istituto Nazionale di Fisica Nucleare*: National Institute of Nuclear Physics. INFN was founded in 1951 and the physicist Gilberto Bernardini became its president.

[37] Funds granted by Pisa and Roma amounted to 150 and 400 million Lira respectively, roughly equivalent to 2.2 and 5.8 million Euro.

promptly replied "*build an electronic computer!*"[38] It was not at all a *boutade* and Fermi agreed to express his mind in a letter to Enrico Avanzi, the rector of the University of Pisa. A suggestion issued by Fermi was immensely authoritative so that Avanzi did not hesitate: the university nominated a three member steering committee – Marcello Conversi (Chair), Alessandro Faedo and Ugo Tiberio[39] – and the CSCE was established as soon as March 1955 on the ground of two preliminary reports that Conversi asked for to Alfonso Caracciolo (a detailed survey of the international state of the art)[40] and Mario Tchou (an overall project plan with the estimate of duration and human and financial resources)[41].

Tchou had been just hired by the Olivetti firm and his welcomed participation to the preliminaries of the CEP project was but a sign of Olivetti's intention to become a partner. After a period of friendly collaboration, in May 1956, the University and Olivetti signed a formal agreement: CSCE offered almost free access to inventions and patents that could arise in the course of the project, while the Olivetti counterpart assured a yearly financial contribution of 10 million Lira, free loan of skilled engineers from its own staff and, whenever appropriate, special discounts on auxiliary equipment. It was not an operation of authentic technological transfer – in the style of the close link between the University of Manchester and Ferranti Industries – but represented in Italy a notable and early case of joint effort between academic research and private industry.

Under the supervision of Conversi's steering committee, CSCE, as customary, was structured into a Logic-Mathematical and an Engineering-Electronic section whose leading personalities formed the Acting Group for the technical management of the project[42].

---

[38] Upon receiving the 1939 Nobel Prize for physics, Fermi emigrated to USA to escape the fascist racial laws (his wife came from a Jewish family). After the war he had occasion to make intensive use of electronic computers, especially of the Los Alamos MANIAC machine. That of 1954 had to be his last trip to Italy: a year later he died in Chicago, aged fifty four.

[39] They represented the most concerned Faculties: Physics (Conversi), Mathematics (Faedo) and Engineering (Tiberio). We already mentioned Conversi's previous experiences with electronics. Faedo undertook a brilliant academic and political career (Rector at Pisa, President of CNR, Senator of the Republic) along which – much like Luigi Dadda – he never ceased to authoritatively promote and support the development of informatics in Italy. Tiberio, during the war, had led an electronic laboratory aimed to set up Radar systems for the Italian Navy: a research effort that aborted due to the blindness of higher military authorities.

[40] The 'Caracciolo report' largely drew (and fairly quoted in bibliography) the already mentioned articles [14] and [15] by de Finetti and others.

[41] Conversi circulated the envisaged initiative among the scientific community. Answers and comments were positive and encouraging enough, with a notable exception: Picone vividly objected against wasting precious financial resources in an initiative that appeared superfluous because Italy already had the FINAC computer. He minded that a single powerful machine would suffice for the entire Country. It was of course a short-sighted – and possibly 'egoistic' – opinion; moreover, Picone was still unaware of what already happened at Milano.

[42] Acting Group members: the director of L-M (Caracciolo) and E-E section (part-time, until April 1956; Giuseppe Cecchini until February 1961, when Giovanni B. Gerace succeeded), Elio Fabri (L-M section) and Sergio Sibani (E-E section). Tchou, Cecchini and Sibani, together with Vladimiro Sabbadini, were the four engineers that Olivetti assigned to the CEP project. Worthwhile mentioning that Corrado Böhm, on leave from INAC, joined the L-M section for one year (October 1958 – October 1959). When the CEP computer entered regular service (1960) a third section – *Gruppo Servizio Calcoli*: Computer Service Group – was formed in order to manage the machine, define work schedule and assist users.

In accordance with 1955 original schedule, the Reduced Machine (MR: *Macchina Ridotta*) was completed in December 1957 and became the first computer entirely designed and built by an Italian team.

Despite its limited capacity, MR was powerful enough to serve as an effective computing tool: several anything but trivial computations were performed in the matter of number theory, Monte Carlo methods, atomic and particle physics, crystal structure and other. Setting up of MR mainly functioned as a training laboratory where CSCE personnel acquired mastery on every facet of a computer project.

The large CEP computer had been initially conceived as a mere enlargement of the MR, to be accomplished with the addition of a magnetic drum and a line printer as well as by extending word length from 18 to 36 bit. The rapidly evolving state of the art and the will of an advanced and much more powerful scientific computer lead to a new and completely different design: as a matter of fact, the almost unique tangible heritage of MR experience was the technology of magnetic core high-speed memory. The radical change of the project benefited of further financial support by INFN and CNR but implied a one year delay over the original schedule.

On the ground of the new design[43], setting up of the CEP computer started on March 1959 and was completed on December 1960. MR was completely disassembled in order to reuse its costly components, vacuum tubes above all.

As far as hardware is concerned, CEP appeared as an hybrid computer: traditional vacuum tubes combined in fact with the emerging transistor technology, that firstly concerned the project in 1956 by means of the doctoral dissertation by Franco Denoth – under Gerace's supervision – about 'Transistor arithmetic unit for an electronic digital computer'. The use of transistors in some units of the machine was carefully evaluated in view of their not yet ascertained reliability: the hybrid trade-off was finally agreed due to the ever better performances of transistors and, most of all, in order to reduce space occupancy and power consumption, even at the cost of a double feeding system to fit the quite different voltage and current working-levels of tubes and transistors.

CEP was a microprogrammed computer and the microprogram was registered on a high-speed eprom (0.1 μs access time) based on the original design introduced by Tom Kilburn for the Manchester MUSE-ATLAS machine[44]. The two magnetic drums – one on-line and one spare – were manufactured at the New Canaan laboratories of Olivetti Corporation of America (see next paragraph), but drum control and I/O interface circuitry remained a CSCE concern. A tape control unit connecting six tape drives was added after completion of the project.

Caracciolo's L-M section carried on intensive research on abstract computer science but its most effective contribution to the project was the development of system and support software that decisively enhanced and made easy an effective use of the machine: let's mention at least a linker-loader – aimed to automatically extract from a library and link to the main program the subroutines called for by the

[43] In order to check in advance the soundness of the new logical design, a software simulator of the CEP was developed at FINAC.

[44] The eprom was constructed from a 256x256 mesh of conducting wires mounted over a soft plastic. Contrasting with the usual technique of fixed cores, ferrite rods of 1x10 mm were inserted in (and removable from) the interstices of the mesh wherever a 'ONE' bit were to be stored [24][29].

latter – as well as a Fortran II compiler, augmented with a series of instructions intended to take advantage of peculiar hardware features of the CEP[45].

CEP official inauguration was held on November 1961and Faedo – University's Rector since November 1959 – had the privilege to welcome the President of the Republic Giovanni Gronchi, who was not new to such happenings. More meaningful than that mundane ceremony was the attention that authoritative analysts dedicated to the completion of the project (see Box 1). CEP productive life lasted eight years, with about 3,000 hours per year of working time.

BOX # 1: Some comments in praise of CEP

The University of Pisa's Computer Centre is engaged in logical design, computer programming, numerical analysis, and electronic design and construction. The Computer development work is among the most advanced observed in Europe.

I.L. Auerbach [24]

… it is unfortunate that the CEP was not completed earlier than 1960, before the large-scale importation of foreign computers, when it might have had a wider influence on computers and computer application in Italy …

N.M. Blackman [30]

… though only transistorized in part, CEP is the most advanced and the most powerful university made computer of the West-European Continent.

J.L.F. Kerf [31]

Needless to say the CSCE members tightly collaborated with the University to give regular courses concerning almost every branch of computer science. Educational concern of the University of Pisa culminated in 1969, when a specialization degree in *Scienza dell'Informazione* (Information Science) was formally instituted. According to Italian regulations, such a decision had to be decreed by the Ministry of Education. Faedo submitted then a proposal as early as July 1967, supported by an intended *curriculum* that resulted from an inspection of analogous courses already experienced abroad as well as from discussions with Luigi Dadda, of the Polytechnic of Milano, in order to avoid superposition with the specialization in Electronic Engineering. The decree lasted until March 1969 and Faedo later discovered that the lengthy hesitation of the minister had to be ascribed to quite surprising negative advice by the CNR committee for Mathematics[46]. Faedo was obviously right: about five hundred students promptly optioned the new specialization and, over the years, many other universities followed the example[47].

Upon completion of the project, CSCE incurred an identity crisis: funds that were specifically granted for setting up the CEP computer were exhausted and the same support by Olivetti necessarily came to an end. Fortunately enough – after a quite dramatic period during which short-term survival was allowed by a research project

[45] About the Fortran-CEP, Nelson M. Blackman commented: "*Eight people have been developing a FORTRAN Compiler for the CEP; this is an unusually large effort for Europe.*" [30], p. 264. The team was led by Otello G. Mancino [32].

[46] On 1972 Faedo himself became the president of CNR.

[47] Ref. [28].

on digital solid-state technologies committed by Euratom – CSCE, in 1962, was elected as an Institute of CNR, not exclusively bounded to the CEP mission.

Anyway, CSCE activities had to be redirected: when the same glorious CEP was switched off, it was radically restructured into the IEI Institute (*Istituto di Elaborazione dell'Informazione*) aimed to a broad spectrum of research in – and applications of – information technology[48].

It was not enough and the Pisans looked around for further opportunities. Faedo was particularly concerned: among others, in 1963 he was able to catch for Pisa the donation by IBM of a 7090 powerful computer[49]. An agreement between the University and IBM-Italy led to the establishment of a new scientific centre named CNUCE (*Centro Nazionale Universitario di Calcolo Elettronico*)[50].

Thanks to the presence of prestigious scientific institutions and the on-site availability of skilled manpower fostered by the University, Pisa became a quite attractive location for an ever growing number of undertakings concerned with Information Technology, ranging from research centers of very large industries to small ventures by groups of entrepreneur-scientists: a creative mix of science and business in the perfect style of an high-tech district. It was but the ultimate fall-out of Fermi's early intuition and of the formative years of the CEP project.

5 Olivetti

The firm Olivetti was established in 1908 by Camillo Olivetti with a small typewriter factory at Ivrea, a town near Torino. Along three decades the business grew at a steady rate on the domestic and foreign market until Adriano, the eldest son of the founder, succeeded in 1938 when he was thirty seven. Under his clever and resolute leadership[51], Ivrea's enterprise undertook a substantial upwards jump that became

---

[48] IEI was headed by the mathematician Gianfranco Capriz until 1979, when Franco Denoth succeeded. Building up of entire hardware systems obviously reduced to a minimum; a notable exception was the Tau2-Taumus, an early sound synthesizer that enabled real-time recording, composition, manipulation and reproduction of music. It was an IEI-CNUCE joint project (1973-75); the composer Pietro Grossi acted as music advisor and the system functioned until 1987 at the Conservatory 'L. Tartini' in Firenze.

[49] IBM donated three 7090 to European prestigious universities; the other two went to London and Copenhagen. The choice of Pisa as the third recipient was influenced by the warm familiarity between Faedo and Eugenio Fubini, who was then the IBM vice-president responsible for that business. A familiarity that dated back to the late Thirties, when they were both young assistants at the University of Roma; Fubini left Italy in order to escape the fascist racial laws against the Jewish.

[50] CNUCE as well, in 1974, was elected to the rank of CNR Institute. Among its manifold activities, let's mention the Laboratory of Computational Linguistic, where Antonio Zampolli and his assistants resumed, refined and broadened the domain of the pioneering work of the Jesuit father Roberto Busa.

[51] His unparalleled style of management combined the restless care for business with the strong believe in the social responsibility of an industry; an attitude that materialized into many-sided benefits towards workers, their families and the surrounding human community. Trade-unions often accused Adriano of subtly hidden paternalism aimed to better submit the workers; on the opposite side, the quite conservative Manufacturers' Association (*Confindustria*) treated him as a dangerous revolutionary utopist and someone arrived to boycott Olivetti's products. Being an exquisitely learned personality, he fostered over the years a sort of cultural circle – comprising architects, sociologists, psychologists and artists – strictly tied to various practical activities of the firm [33][34][35].

particularly intensive during the post-war years, making Olivetti a multinational industrial holding and a leader in the worldwide market of office equipment[52].

Letting aside the already mentioned negotiations with INAC, Olivetti became directly concerned with electronic computing by means of two separate initiatives.

The first one was Olivetti-Bull, a fifty-fifty company established in 1949 in order to market in Italy the punch-card equipment produced by the French *Compagnie des Machines Bull*, the principal opponent of IBM in Europe. Later on, Bull products evolved into early small sized electronic computers: the Gamma3 and the Gamma3-ET (*Extension Tambour*, i.e. with a magnetic drum memory). Olivetti-Bull did not contribute to Olivetti's technological know-how but allowed its personnel to become acquainted with computer programming and, most of all, to fight against the commercial aggressive policy of IBM[53].

The second one was the establishment in 1952 of an Electronic Laboratory at New Canaan (Connecticut). The laboratory acted in practice as a private territory of Dino Olivetti – Adriano's youngest brother – and Ivrea's headquarter ever considered it as as a mere observatory over the USA state-of-the-art. Things went partially differently, mainly due to the skilled technical directorship by Michele (later changed into Mike) Cànepa[54]. The laboratory, worked in fact on several electronic products and an Olivetti-GBM (General Bookkeeping Machine) was mentioned in a 1955 survey of computers available in USA[55]; it was a small machine with 112 vacuum tubes, 450 crystal diodes and a magnetic drum storage. The laboratory specialized on magnetic drums, that apparently became the sole marketable product but with almost negligible result, even if two drums arrived to fit out the CEP computer. As sought from the Italian motherhouse, New Canaan laboratory appeared only a source of expenditure and it was then shut in 1961. Dino Olivetti went back to Italy while Cànepa, after a brief and rather disappointing stay at the Olivetti LRE, definitely migrated to the United States[56].

In Adriano's mind, electronics never represented a departure from the traditional and successful business of office equipment: his long-sighting instinct conceived instead the electronic computer – better to say, the electronic data-processor – as the core of the 'office of the future'[57]. In other words, he clearly felt that the 'computer by

[52] New plants were built in Italy and in abroad strategic locations; an efficient worldwide commercial network was established; mechanical calculators and accounting machines flanked traditional typewriters as flag-product but other products arose like teletypes, office steel furniture, numeric-control machine tools.

[53] The first Gamma3 Italian installation occurred in 1953. The engineer and former commander of the Navy Ottorino Beltrami was the director of Olivetti-Bull, while the very young Elserino Piol emerged as the technical and sales-supporting driving person, thus beginning a brilliant career as a high level manager. The essay [38] contains a detailed history of Olivetti-Bull.

[54] We already met him as an Olivetti advisor to Picone and a temporary member of Aiken's crew for the Mark IV project.

[55] Ref. [37].

[56] A further witness of such unfriendly relationship can be found in the fact that, according to a 1959 announcement of the Elea 9003 [39] the machine had to be equipped with an auxiliary mass storage of up to three magnetic drums – very likely supplied by New Canaan – but this feature was in fact never implemented.

[57] It seems likely that Adriano's vision could be influenced and comforted by the early (about 1950) commercially available and business-oriented computers built by Eckert and Mauchly in the USA (the Univac, soon acquired by Remington Rand) and by Lyons Industries in England (the LEO, meaningfully christened Lyons Electronic Office). No doubt that Adriano Olivetti was well aware of both initiatives.

the scientists for the scientists' was to be superseded by the 'data-processor by the industry for the market'[58].

He was nevertheless perfectly conscious that a strong scientific and technological groundwork was the prerequisite to afford the computer market with an original and competitive product. The first move was to find out a skilled and reliable person to lead the R&D activities of the firm in the domain of electronic computers. Adriano ever had the ability to immediately perceive people's personality; during a trip to the United States, he met Mario Tchou, a thirty years old professor at Columbia University, and was touched by his human and professional traits[59]. Within a few months, in 1954, Tchou was hired with the twofold heavy responsibility to contribute at CSCE's early activities (see latter paragraph) and to set up the already mentioned Olivetti LRE research laboratory. According to Adriano's intuition, Tchou emerged as an excellent manager as well as a resolute but friendly leader of the LRE team, a small group of engineers and physicists – all of them in their twenties – carefully selected by himself.

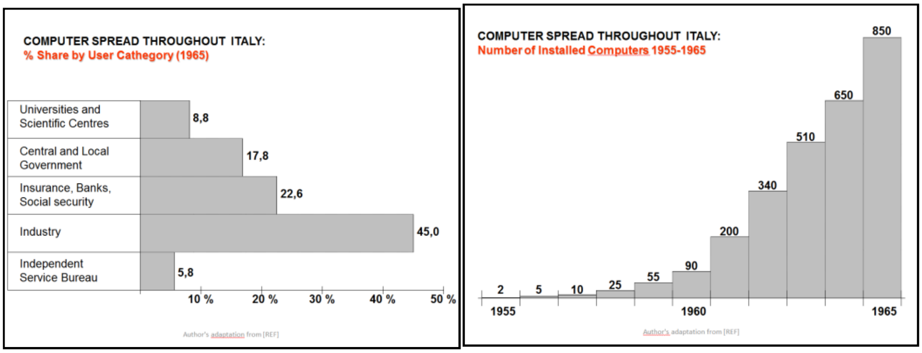


Fig. 1. and 2. Computer spread throughout Italy

Even if CSCE's and Olivetti's initiatives shared the courageous 'make' approach, we owe it to point out some intrinsic difference between them. First of all, before the advent of the general purpose computers, the architecture of a scientific machine markedly departed from that of a business-oriented one. As a further difference, whilst the CEP was intended as a one-of-a-kind machine to be used in a self-contained laboratory-like environment, the Olivetti computer was conceived as an industrial product; it had therefore to fulfill specific needs of mass production as well as easy of maintenance, ergonomics and aesthetic appeal. Last but not least, CEP project was carried on in the style of an 'open' contribution to the advancement of science, thus originating a

[58] Adriano's vision anticipated in fact the observed trend. As far as Italy is concerned, the histograms in figures 1 and 2 witness both the rapidly spread of computers – all of them but the single CEP were manufactured by industry – and the rapidly % decrease of scientific users. Both histograms are the author's elaboration from the Dadda's 1967 report [36].

[59] Mario Tchou was born in Roma on 1924 and ever lived there; his father was in fact a diplomatic officer at the Chinese embassy. When the war ceased he went to the USA, becoming bachelor in electrical engineering (1947), Master of Science in physics (1949) and associate professor of electrical engineering at Columbia University (1952).

flush of papers detailing scientific and technical achievements; Olivetti's policy – according to common practice in industrial research – privileged instead confidentiality so that articles and reports – anyway of more or less promotional scope – appeared only in the latest phase of the project.

In early 1956, after a period of hospitality at the University of Pisa, LRE started operations. It was housed in a pleasant ancient villa at Barbaricìna, a suburb of Pisa, and the LRE crew became currently known as the 'Barbaricìna guys'. With the exception of Giorgio Sacerdoti and Martin Friedman, LRE members almost lacked of previous specific knowledge[60]. When Tchou selected them, he mostly appreciated in fact their intellectual potentialities, so that they only needed brief and intense training to plenty master the fundamentals of the new science. While Sacerdoti knowledgeably assisted him in defining the overall design of the machine, Tchou committed to each member of the staff the care of a specific functional subsystem and regular meetings were held to check the interfaces.

It seemed almost incredible that at the end of 1957, well in advance over the initial schedule, Tchou and his guys – whose number grew up to thirty – completed the Zero Machine (*Macchina Zero*), the vacuum tube prototype of Elea computers, with a transistorized tape control unit designed by Lucio Borriello[61]. A second prototype immediately followed and it was only at this point that Adriano officially confirmed electronics as a new concern of his industry and allowed further investments.

It was not enough because Tchou insisted that Olivetti had to afford the market with a first class product and full transistorization was absolutely needed. Saving the logical and functional design, circuitry had to be completely redefined; a quite demanding task that LRE quickly accomplished so that, in autumn 1959, the production plant was able to deliver the Elea 9003, one of the worldwide earliest – possibly the first – solid-state commercially available computers.

Other notable features were: magnetic core memory expansible from 20,000 up to 160,000 characters in modules of 20,000;[62] variable length memory fields; interrupt signals that allowed the simultaneous execution of three programs; up to twenty magnetic tape units; last but not least, an ergonomic and pleasant appearance invented by the architect Ettore Sottsass.

---

[60] Sacerdoti came from the engineering group of FINAC. Friedman, a Canadian engineer, previously worked with Ferranti Industries at Manchester.

[61] An assessment of synonymous names that often appear in literature:

- Macchina Zero: the vacuum tube prototype (tape control unit made with germanium transistors) built up at Barbaricina on rough laboratory racks.
- 1V / Elea 9001: the same Macchina Zero (#1 Vacuum tube prototype) moved to Ivrea, dressed in a suitable office style and used as a complement to the punch-card data processing centre.
- 2V / Elea 9002: #2 Vacuum tube prototype; tape control unit made with silicon transistors; installed at Milano as a demo-centre and software laboratory.
- 1T / Elea 9003: #1 wholly Transistorized prototype and production model at the Borgolombardo factory.

Elea is a contraction of *Elaboratore Elettronico Automatico*.

[62] A character was coded into a 'Byte' of six bit plus one for parity check. Filippazzi invented a method that halved the bit-selection circuitry; the invention was patented by Olivetti and licensed to Plessey in England.

The computer factory was established at Borgolombardo, near Milano, where the research group of LRE also moved. Production activities together with the setting up of a commercial organization – including hardware and software customer assistance – became the most demanding and financially-draining concern of Olivetti. It soon appeared that the computer business could benefit from the commercial experience already matured by Olivetti-Bull, also because punch-card and electronic data processing shared the same category of customers. Olivetti then acquired 99% of Olivetti-Bull and merged it into the DEO (*Divisione Elettronica Olivetti*), headed by Ottorino Beltrami and with Elserino Piol as marketing manager.

The exciting times of Barbaricìna were over, but intensive research on new products continued under Tchou's directorship, willingly flanked by Roberto Olivetti to which his father Adriano, since the beginning, committed special care to the electronic branch. The most valuable achievement was the mid-range Elea 6001 computer that had a micro-programmed architecture: it was delivered in 1961 and became nearly standard equipment for university computing centers. To improve sales of this successful product, Piol insisted to slightly modify the machine into the Elea 6001-C model ('C' for *Commerciale*), better suited for business applications.

When considering the activities that LRE carried on in the domain of software, some weaknesses appear in retrospect, especially in comparison with the experiences we have discussed in former paragraphs.

During the early phases of the Elea project, attention was almost exclusively focused on hardware while the critical role of software begun to be appreciated only upon completion of the Zero Machine. The care of software was committed to a small – and perhaps under-sized – group led by the mathematician Mauro Pacelli, who privileged the theoretical study of high-level programming languages – the so-called 'logic programming' – leaving almost unattended the domain of application software; an approach that compelled the commercial division to pursue almost at random the software requirements issued by customers.

To give but an example, the lack of a software supervisor left to the application programmer the exceedingly delicate handling of concurrent programs, so that multiprogramming – a distinguished feature of Elea 9003 – rested almost unused.

Later on, the LRE logic programming team defined an original Algol-like language, named Palgo, and built a compiler for the Elea 6001. But it happened that the 6001 users – almost all of them were scientific centres – urgently asked for a Fortran compiler and it was the commercial division, not the LRE, to fulfill in a hurry such a request[63].

The massive import of solid-state components concerned not only Olivetti but also other industries, namely for consumer electronics, giving the opportunity to set up an autonomous production in Italy. Tchou and Roberto convinced Olivetti to catch the opportunity: the SGS company (*Società Generale Semiconduttori*) was then established in association with Telettra and started production under Fairchild

[63] It should be remembered that high-level computer languages were then in their magmatic infancy, far from benefiting of much later standard definition by *super partes* authorities. It seems likely that Pacelli refused Fortran in order not to appear as an IBM-follower. Moreover, Algol represented the leading edge of the state-of- the-art and was ennobled by its academic and mostly European origin.

Semiconductor license[64]. SGS revealed a far-seeing initiative that grew over the years independently from Olivetti's up and down: it is now-days well alive with the new name ST-Microelectronics and with the partnership of the Thompson French industry.

During the early sixties, forty Elea 9003 and seventy 6001 were delivered, that represented about a 20-30% share of the domestic market; a result that could appear rather satisfactory, also because the Italian government never supported Olivetti with benefits – like preference in procurements, protectionism, assignment of strategic projects – that were common practice abroad in backing of the 'national champions' of the computer industry[65]. Moreover, DEO cash-flow heavily suffered – under the pressure of IBM's commercial policy – the practice of renting instead of selling computers. Looking at the future, the most serious drawback appeared when considering that the domestic market was far from reaching the critical width that the computer business necessitated in order to get a profitable return on investment; after all, it was the time of IBM 1401, the first computer to sell over 10,000 worldwide[66]. On the other hand, the powerful multinational structure of Olivetti was strictly bound to the traditional mechanical products and was then unable – as well as reluctant! – to engage the extraneous and completely different business of computers. Roberto Olivetti and Tchou sought to escape national confinement but their effort to gather abroad some industrial and/or commercial partner revealed as fruitless[67].

We owe also mention that Adriano, in 1959, audaciously embarked in taking control of Underwood, an American industry whose production of office equipment fairly matched Olivetti's activities. It was the key to enter the huge and wealthy market of the States and the acquisition of a prestigious American firm by an Italian one caused sensation at the stock exchange; on the occasion, the Olivetti Corporation of America was established and the brand modified into Olivetti-Underwood.

The successful strategic move really strengthened Olivetti's leadership in the international market but the business was soon revealed as less advantageous than expected: not only the stock acquisition price was overestimated but Underwood's works at Hartford (Connecticut) appeared obsolete to the point of requiring heavy investments for restructuring. Due to Olivetti's poor financial capacity, investments implied loans from banks and the growth of indebtedness became a critical and risky concern.

Such was the overall picture when Adriano Olivetti, on February 1960, suddenly died of heart attack; the firm was completely unprepared to face such a disaster and the members of Olivetti family – the majority shareholders – decided to commit management to some extraneous personality. The electronics at Olivetti had lost its

---

[64] Worthwhile mentioning that Federico Faggin began his career at SGS. He was then hired by Fairchild and later on joined Intel, where in 1971, together with Marcian (Ted) Hoff, he built the first microprocessor, the celebrated Intel 4004.

[65] The notion of 'national champion' is borrowed from Chandler's essay [40].

[66] Because of the confinement within domestic borders, Elea computers remained almost unknown abroad. This also accounts for the recurrent lack of consideration by foreign historians.

[67] Similar attempts by Beltrami and Dino Olivetti had the same negative outcome. As far as the computer industry is concerned, European countries repeatedly exhibited over the years their inability to cooperate and the venture of Unidata is paradigmatic: it was established in 1972 as a transnational joint venture between Cii (France), Siemens (West Germany) and Philips (Nederland). Unidata mission consisted in a strong reaction against the supremacy of the Americans but reciprocal jealousy and mistrust of the partners caused its vanishing within a couple of years.

authoritative tutelary deity and the supporters of the mechanical tradition, who prevailed at any level of the hierarchy, felt free to rumour against LRE and Deo that they measured only on the base of poor financial performance.

It was not enough: on November 1961, even Mario Tchou – aged only thirty seven – perished in a car crash, thus leaving Roberto Olivetti the sole defender of electronics[68].

The final scene matured between 1963 and 1964 when the economy entered a worldwide recessive cycle that inevitably affected also the market of office equipment. Olivetti's finance definitely exhausted and the president Bruno Visentini – a distinguished economist – in order to avoid catastrophic bankruptcy, urged for an 'intervention group' of private banks and industries; the group acquired control of the stock – thus reducing to almost nothing the share owned by the Olivetti family – and guaranteed against default but firmly stated dismissal of the electronic division as a preliminary issue[69]. It was argued that – letting aside the excellence in technology – investments required to successfully run the computer market overwhelmed the capabilities of every Italian industry; electronics represented after all only a tiny fraction of Olivetti's overall business that included about thirty affiliate companies abroad, twenty factories and fifty five thousand employees.

Just in that moment, General Electric was seeking for opportunities in order to widen its computer business in Europe; Olivetti's DEO in Italy and Bull in France – that too was facing a financial crisis – appeared the most viable chances and GE took advantage from both.

Between 1964 and 1965 DEO was then sold to the Americans[70]. This marked not only the defeat of an admirable adventure, but also the definitive technological and commercial subduedness of Italy in the domain of computers. It is interesting enough that the government completely failed to appreciate the damage of losing such a unique and strategic asset, while in France the delivery of Bull to foreign hands was felt as a wound to national pride, at the point that President Charles De Gaulle reacted launching the celebrated *Plan Calcul*.[71]

In the meantime a gratifying surprise matured thanks to Pier Giorgio Perotto, an engineer who was with Tchou's team since 1957. Perotto's personality resulted incompatible with the GE representatives so that – together with his assistants Giovanni De Sandre and Gastone Garziera – he didn't quit Olivetti. In fair agreement with Roberto Olivetti and working with extremely poor resources, he was able to work out a small desktop electronic computer; not a toy but instead a professional tool at any rate, programmable by means of an essential but effective language. The Olivetti Programma 101 personal computer was born and when it was exhibited in New York

---

[68] He was elected among the top managers but almost deprived of autonomous initiative. Beltrami – the DEO managing director – and Sacerdoti – who succeeded as responsible of the research laboratory – loyally assisted him but they were both in an even worst position.

[69] Ref. [51].

[70] To be precise, the OGE (Olivetti-General Electric) company was established (75% GE and 25% Olivetti) that became GEISI (General Electric Information Systems Italy) when GE very soon acquired 100%. Later on (1970) Honeywell replaced General Electric and GEISI became HISI (Honeywell Information Systems Italy).

[71] Ref. [47], [48].

at the 1965 Bema Show the success was astonishing, almost obscuring the traditional mechanical products. Over the years, 44,000 machines were shipped worldwide, many of them in the United States with the Olivetti-Underwood brand (the NASA space agency was among the first customers and bought sixty)[72].

The commercial success of the Programma 101 (P 101 for short) shocked the supporters of the mechanical technology: little by little they accepted the new course of Olivetti towards the so called 'light' electronics[73]. Worthwhile mentioning that the P 101 incorporated an I/O original feature that used removable and reusable magnetic floppy-cards as the medium for storing programs and data. Hewlett-Packard later cloned that feature in its HP 9100 model and paid to Olivetti a royalty of about one million dollars.

Olivetti's withdrawal from 'big' electronics keeps attracting the attention of Italian historians who unanimously complain about that unfortunate happening[74].

Assuming a complementary point of view, it has been remarked that the LRE-DEO experience was not scratched out by General Electric but rather merged into a multinational enterprise with new criteria to handle the business[75]. As a matter of fact – and even if some discontinuity occurred as in the case of the already mentioned Elea 9004 project – the Italian managers, human resources, factory and research laboratory were kept almost untouched, and even increased, by General Electric[76].

The story of Elea 4001 is meaningful enough. This medium-sized computer was launched by Olivetti in 1964 and represented the most valuable and ready-to-market asset that DEO brought as a dowry; with a minimum reshaping in order to fit the GE series of computers, Elea 4001 was marketed as GE 115 and sold over 4,000, being particularly successful in the United States. It was the first time that a computer entirely designed and produced in Italy massively reached foreign markets; a result that – according to previous discussion – Olivetti could have hardly achieved.

Over the years, many other 'Italian' computers went through the word under the General Electric – and later Honeywell – brand that left hidden their origin; among them, in the early Seventieth, the Level 2 model of the Honeywell Series 60, was even licensed to the Japanese Hitachi.

---

[72] Ref. [50]. A key factor of the P 101 success was that, quite differently from huge computer systems, it perfectly matched the practices of office equipment sale forces: the P 101 could be sold piece by piece (at the price of $3,200) and was extremely easy to use and to maintain.

[73] Later on, electronics arrived to affect the same traditional office equipment: the Auditronic 770 accounting machine (1969), the Logos 250 calculator (1970) and the ET 101 typewriter (1978) represented the first electronic products of their category.

[74] Ref. [52], [53].

[75] Ref. [54]. To give but an example, 'product planning' methodology was introduced by the Americans and represented a novelty for the Italian management culture.

[76] The computer factory remained at Caluso, near Ivrea, and the research centre was kept at Pregnana (Milano) under the guidance of Franco Filippazzi.

APPENDIX: Where to Meet Computer Relics throughout Italy

MILANO: CRC 102-A Central Processing Unit (*Politecnico di Milano*).
PISA: CEP, CINAC (CPU only), ELEA 6001, Bull Gamma 3, Tau2-Taumus and many more (*Museo degli strumenti di calcolo*); Documents at the *Archivio Generale di Ateneo*.
ROMA: ELEA 9003 (*Museo delle Poste e Telecomunicazioni*); Documents and INAC's mechanical calculators at the *Archivio Storico IAC*.
BIBBIENA (Arezzo): ELEA 9003, functioning! (*Istituto Tecnico 'E. Fermi'*).
IVREA (Torino): Olivetti P101 and many other Olivetti products (*Museo-Laboratorio Tecnologic@mente*); Documents at the *Archivio Storico Olivetti*.

References

[1] De Marco, G., Mainetto, G., Pisani, S., Savino, P.: The Early Computers of Italy. IEEE Annals of the History of Computing 21(4), 28–36 (1999)

[2] Bonfanti, C.: Mezzo secolo di futuro. – L'informatica italiana compie cinquant'anni. Mondo Digitale (3), 48–68 (2004)

[3] Cull, P.: Caianiello and Neural Nets. In: Termini, S. (ed.) Imagination and Rigor – Essays on Eduardo R. Caianiello's Scientific Heritage, pp. 47–61. Springer (2006)

[4] Fadini, B.: Calcolo analogico ed ibrido – L'esperienza dell'Università di Napoli. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, pp. 127–144. AICA, Milano (1991)

[5] Contin, G., Patergnani, G., Piva, F.: Calcolatore elettronico per uso didattico e scientifico. In: Abstract from the Proceedings of the VIII Rassgna Internazionale Elettronica e Nucleare, Roma (1961)

[6] Böhm, C.: Calculatrices Digitales – Du déchiffrage de formules logico-mathématiques par la machine même dans la conception du programme. Annali di Matematica Pura ed Applicata (1954)

[7] Böhm, C., Jacopini, G.: Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules. Comm. of the ACM 9(3), 366–371 (1966)

[8] Croce, B.: Logica come scienza del concetto puro, 7th edn. Laterza (1958); 1st edn. (1905)

[9] Carassa, F.: La nascita dell'elettronica e delle telecomunicazioni: Vecchiacchi e i rapporti con l'industria. In: Il Politecnico di Milano Nella Storia Italiana, vol. II, pp. 499–506. Laterza (1964)

[10] Dadda, L.: Il centro di calcoli numerici e l'introduzione delle discipline informatiche al Politecnico di Milano. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, pp. 7–44. AICA, Milano (1991)

[11] Dadda, L.: Ricordi di un informatico. In: Cuzzer, A. (ed.) La cultura Informatica in Italia – Riflessioni e Testimonianze Sulle Origini, 1950-1970, pp. 67–106. Bollati Boringhieri, Torino (1993)

[12] Nastasi, P.: Picone, il calcolo automatico e *Finac*: una storia lunga 30 anni. Pristem/Storia (12-13), 121–171 (2006)

[13] Guerraggio, A., Mattaliano, M., Nastasi, P.: La "lunga marcia" di Mauro Picone (1885-1977). Università Bocconi, Milano (2010)

[14] de Finetti, B.: Macchine "che pensano" (e che fanno pensare). Tecnica ed Organizzazione (3-4), 182–215 (1953)

[15] de Finetti, B., Kitz, N., Rodinò, G.: Symposium on automatic digital computation – Nat. Phys. Laboratory – Teddington (Londra). La Ricerca Scientifica (7), 1248–1259 (1953) (the article is in Italian)
[16] Goldstine, H.H.: The Computer from Pascal to von Neumann. Princeton University Press (1972)
[17] Bonfanti, C.: L'affare Finac tra Manchester e Roma (1953-1955) ed alcuni documenti inediti ad esso relativi. In: Atti del Congresso Annuale AICA – Palermo, pp. 35–64. AICA, Milano (1994)
[18] Ercoli, P.: Otto anni di attività del calcolatore dell'Istituto Nazionale per le Applicazioni del Calcolo. In: Atti del I° Congresso Annuale AICA, Bologna (1963)
[19] Ercoli, P.: Systems and Programming Aspects of the Computer INAC. Calcolo (4), 441–470 (1966)
[20] Vittorelli, V.: The Olivetti-INAC Computer. Calcolo (4), 481–491 (1966)
[21] Ercoli, P.: From CINAC to FINAC. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e dell'Informatica, Siena, Settembre 10-12, pp. 57–68. AICA, Milano (1991)
[22] Numerico, T., Freguglia, R.: Le ricerche in informatica. In: Simili, R., Paoloni, G. (eds.) Per una Storia del Consiglio Nazionale Delle Ricerche, Laterza, vol. II, pp. 408–440 (2001)
[23] Rossi, B.: Cosmic Rays. McGraw-Hill (1964)
[24] Auerbach, I.L.: European Electronic Data Processing – A Report on the Industry and the State-of-the-Art. In: Proc. of the IRE, pp. 330–348 (January 1961)
[25] Denoth, F.: CEP (Calcolatrice Elettronica Pisana. Dal CSCE all'IEI. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, Siena, Settembre 10-12, pp. 105–115. AICA, Milano (1991)
[26] Andronico, A.: Alfonso Caracciolo di Forino: attività scientifica a Pisa. In: Vanneschi, M. (ed.) La CEP: Storia, Scienza e Umanità Dell'avventura Informatica Pisana, pp. 66–93. Felici Editore, Pisa (2009)
[27] Vanneschi, M.: L'opera scientifica di Giovan Battista Gerace. In: Vanneschi, M. (ed.) La CEP: Storia, Scienza e Umanità Dell'avventura Informatica Pisana, pp. 94–105. Felici Editore, Pisa (2009)
[28] Faedo, A.: L'ambiente pisano e l'istituzione del primo corso di laurea in Scienza dell'Informazione. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, Siena, Settembre 10-12, pp. 119–125. AICA, Milano (1991)
[29] Maestrini, P.: Le tecnologie digitali al tempo della CEP. In: Vanneschi, M. (ed.) La CEP: Storia, Scienza e Umanità Dell'avventura Informatica Pisana, pp. 66–93. Felici Editore, Pisa (2009)
[30] Blackman, N.M.: The state of Digital Computer Technology in Europe. Comm. of the ACM 4, 256–265 (1961)
[31] Kerf, J.L.F.: A survey of the New West-European Digital Computers. Computers and Automation (12), 27–38 (1963)
[32] Mancino, O.G., Sprugnoli, R.: CEP: la Calcolatrice Elettronica Pisana – Scenario, storia, realizzazione, eredità. Pisa University Press (2011)
[33] Caizzi, B.: Camillo e Adriano Olivetti. U.T.E.T., Torino (1962)
[34] Occhetto, V.: Adriano Olivetti industriale e utopista. Cossavella Editore, Ivrea (2000)
[35] Renzi, E.: Comunità concreta: le opere e il pensiero di Adriano Olivetti. Guida Editore, Napoli (2008)
[36] Dadda, L.: Il settore dei calcolatori. In: La Ricerca Industriale per L'Italia di Domain, pp. 549–571. FAST, Milano (1967)

[37] Weik M.A.: A Survey of Domestic Electronic Digital Computer Systems. U.S. Department of Commerce – Office of Technical Services (December 1955)
[38] Bordòli, P.: Olivetti-Bull (1949-1964) – Una tappa importante per l'informatica italiana. Doctoral thesis at the University of Modena and Reggio Emilia (2012)
[39] Sacerdoti, G.: Il calcolatore Olivetti Elea 9003. In: Contributi All'automazione Aziendale, pp. 3–13. Olivetti S.p.A (1959)
[40] Chandler, A.D.: Inventing the Electronic Century. The Free Press (2001)
[41] Filippazzi, F., Sacerdoti, G.: Progetto ELEA: il primo computer made in Italy. In: Convegno Internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, Siena, Settembre 10-12, pp. 187–203. AICA, Milano (1991)
[42] De Witt, G.: Le fabbriche e il mondo. L'Olivetti industriale nella competizione globale (1950-1990). Franco Angeli, Milano (2005)
[43] Piol, E.: Il sogno di un'impresa. Dall'Olivetti al venture capital: una vita nell'Information Technology. Edizioni de Il Sole 24 Ore, Milano (2004)
[44] Beltrami, O.: Sul ponte di comando: dalla Marina Militare all'Olivetti. Mursia, Milano (2004)
[45] Bellisario, M.: Donna e top manager: la mia storia. Rizzoli, Milano (1987)
[46] Silmo, G.: L'Olivetti e l'elettronica: una storia esemplare. Tipografia Giannotti Editore (2010)
[47] Brulé, J.-P.: L'Informatique malade de l'État – Du Plan calcul à Bull nationalisée. Les Belles Lettres (1993)
[48] Mounier-Kuhn, P.-E.: National Policies Towards Informatics in France (1945-1975). In: Convegno internazionale Sulla Storia e Preistoria del Calcolo Automatico e Dell'informatica, Siena, Settembre 10-12, pp. 353–369. AICA, Milano (1991)
[49] Rao, G.: La sfida al futuro di Adriano e Roberto Olivetti. MéLanges de l'École Française de Rome, Tome 115, 643–678 (2003)
[50] Perotto, P.G.: Programma 101 – L'invenzione del Personal Computer: una storia appassionante mai raccontata. Sperling & Kupfer, Milano (1995)
[51] Bricco, P.: Olivetti, prima e dopo Adriano: industria, cultura, estetica. L'ancora, Napoli (2005)
[52] Soria, L.: Informatica: un'occasione perduta. Einaudi, Torino (1979)
[53] Gallino, L.: La scomparsa dell'Italia industriale. Einaudi, Torino (2003)
[54] Bonfanti, C.: L'industria del computer in Italia – Appunti per una storia. In: Luccio. F. (ed.) L'informatica: lo Sviluppo Economico, Tecnologico e Scientifico in Italia, Italia, pp. 75–118. Edifir – Edizioni Firenze (2007)

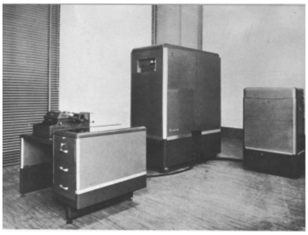


Fig. 2.1. CRC 102-A

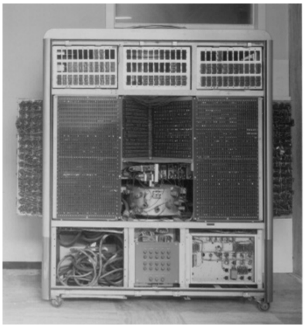


Fig. 2.2. CRC 102-A (CPU) still preserved at the Polytechnic of Milano

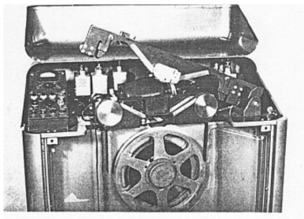


Fig. 2.3. CRC 126 magnetic tape unit

Fig. 2.4. Luigi Dadda (1991)

Fig. 3.1. Mauro Picone (left) with Bruno de Finetti (1952)

Fig. 3.2. FINAC (Ferranti Mark I*)

Fig. 3.3. Official inauguration of the FINAC (1955): President Giovanni Gronchi with Picone (centre), Enzo Apàro (on the left), Paolo Ercoli (on the right) and Corrado Böhm (second from right

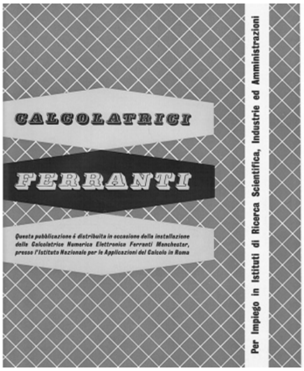


Fig. 3.4. Title-page of a Ferranti's advertising brochure (1955, in Italian!)

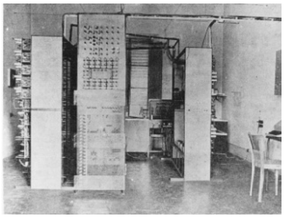


Fig. 4.1. CSCE 'Reduced Machine'

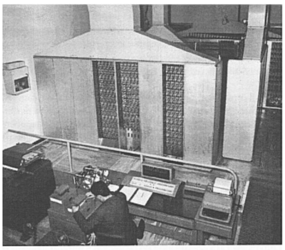


Fig. 4.2. Partial view of the CEP

Fig. 4.3. Alessandro Faedo (about 1964)

Fig. 4.4. Giovanni Battista Gerace (1960)

Fig. 4.5. Marcello Conversi (centre left) and Giuseppe Cecchini (centre right) (1960)

Fig. 4.6. Official inauguration of the CEP (1960): Alessando Faedo on the extreme left side, President Gronchi at centre and Alfonso Caracciolo at his left (1960)

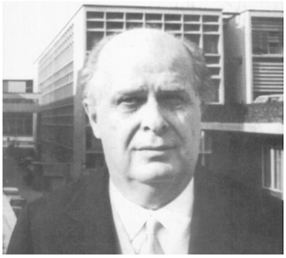


Fig. 5.1. Adriano Olivetti. (About 1958; in the background some Olivetti's buildings in Ivrea)

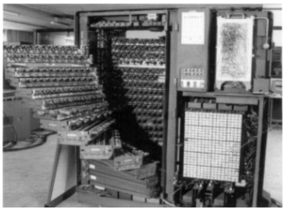


Fig. 5.2. Bull Gamma 3

Fig. 5.3. The 'Barbaricìna guys' (1956) Standing from left: G. Calogero, F. Filippazzi, M. Tchou, R. Galletti, P. Grossi, S. Sibani, G. Sacerdoti. Kneeling from left: L. Borriello, S. Fubini, O. Guarracino, G. Raffo. (On the occasion, M. Friedman was away.)

Fig. 5.4. Roberto Olivetti with Mario Tchou (about 1958)

Fig. 5.5. ELEA 9003 (Industrial design by Ettore Sottsass Jr)

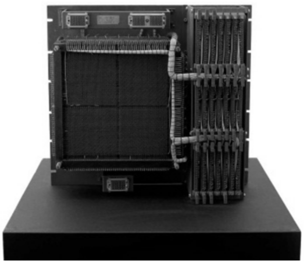


Fig. 5.6. ELEA 9003: a 10 Kchar core memory module

Fig. 5.7. ELEA 6001

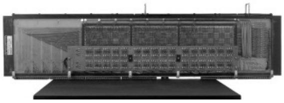


Fig. 5.8. ELEA 6001: microprogrammed ROM

Fig. 5.9. ELEA 4001 – GE 115

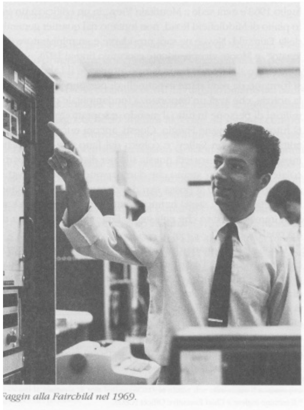


Fig. 5.10. Federico Faggin at Intel (1969)

Fig. 5.11. Pier Giorgio Perotto (1991)

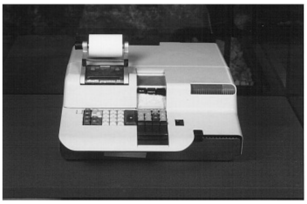


Fig. 5.12. Olivetti P101 desk-top computer (industrial design by Mario Bellini)

Fig. 5.12. Italian stamps: Olivetti M 24 Personal Computer (issued 1986) and Olivetti's Centenary (issued 2008)

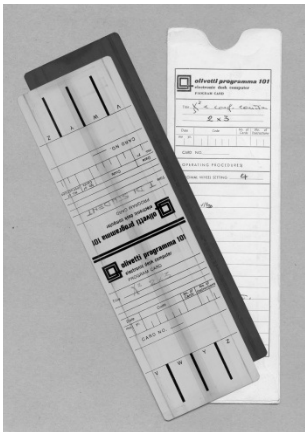


Fig. 5.11. P 101 floppy card (the black reverse is the magnetic-sensitive side)

Institutional Nostalgia – Museum Victoria's Cabinet of Computing Curiosities

David Demant[1] and Arthur Tatnall[2]

[1] Museum Victoria, Melbourne, Australia
[2] Victoria University, Melbourne, Australia
ddemant@museum.vic.gov.au, Arthur.Tatnall@vu.edu.au

Abstract. Museum Victoria has a significant collection of objects that could be described as 'computing curiosities'. Undoubtedly, its most important exhibit is CSIRAC (formerly CSIR Mk1), which was the world's fourth electronic digital computer and the only remaining intact first generation computer in the world. The collection of computers and related items, including calculators, range in date from before CSIRAC (1949) to the iPad. This article examines some of these items, what they did and how they were used at the time.

Keywords: History, electronic digital computers, analogue computers, museum collections.

1 If You Go Down ...

... to the lower ground floor of Melbourne Museum[1], you're sure of a room sized surprise: there displayed is the only intact first generation electronic stored program computer left on the planet. Named CSIRAC[2], it was the fourth ever made and the first computer in Australia[3].

While you watch the video presentation beside the display, people pass by, some even stop. A parent might point out to their child: "*That's the first computer in Australia – it's as big as a room*". The child might ask "*Is that the screen?*" or "*What was it used for?*" The parent will mumble something or quickly read a label before chasing after the kids on the way to the dinosaurs or the cafeteria.

Later, a lone male spends a lot of time at the display. Then his partner appears and the mono-nostalgia-logue begins: "*You remember how I stuck with DOS right up until Windows 3.0 came out in '90. I was sold and then 3.1 came out in, I think, 92, wow! It was miles better. You know ... looking at this thing ... those punch cards[4] … St Kilda*

[1] Melbourne Museum (http://museumvictoria.com.au/melbournemuseum/) is located in Carlton Gardens, Melbourne, Victoria.
[2] CSIRAC stands for Commonwealth Scientific (and) Industrial Research (organisation) Automatic Computer.
[3] CSIRAC operated from 1949 to 1964.
[4] CSIRAC used paper tape for most of its life; punch cards were only used in the beginning. The reference to cards reflects the fact that many more people remember punch cards.

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 348–361, 2012.

Road ... the building's gone now I think ... near the Barracks. . . and dropping them ... you had to go back to the end of the queue …"

Museum Victoria has a historical collection of computers and related items dating from before CSIRAC to the iPad. This article will present a very few of those objects. Hopefully, you will experience an emotion somewhere in between the range bookended by the two situations described above.

2 A Treasure Chest for Posterity

The Museum's database contains over 4,400 computer related records, covering items such as hardware, a variety of media holding software and data, manuals, program listing, reference material, diagrams – block, logical, schematic, timing and mechanical, sales brochures, promotional material, correspondence, office records such as tenders and acquisitions, oral and written histories, and correspondence.

Other historical technological collections include telegraphy, telephony, radio, television, electronics, calculating technology, playback and recording, and printing including typewriting, writing, copying and duplication.

All these collections have been built up over many years by many dedicated people, mainly curators pursuing specific interests realated to exhibitions and displays. Over the most recent period, collection policies have been developed to direct collection management, acquisition of new items and deaccessioning items that are duplicated, damaged irreparably, or have no significance or provenance. A significant aspect of these policies is the insistence on both acquiring items and recording the context and stories that give life to the items.

As a result, Museum staff have been able to be more proactive in developing and managing the collections, rather than responding to offers of items or acquiring items without a clearly articulated context. An essential element in this change in the mode of operation is the use of collection databases, which allow data and relationships to be recorded to great depth; something that it was almost impossible to do with the old card index.

The policies encourage the active participation of the community; close partnerships have been established with experts and organisations outside the Museum, as well as volunteers. This is especially true for the computer collection; the Museum has worked closely with the University of Melbourne's CSIRAC History Team, which includes former CSIRAC staff and users. Again, working with the Internet Macintosh User Group, the Museum has collected over 200 items that embody the story of Apple Computer Inc.

In both these and other cases, the Museum is able to tell stories that reveal the personal and social as well as the professional and technical aspects of the items collected. In one sense the computer collection has come full circle since it is the use of computer databases that have placed the collection within its full context.

2.1 Collections On-Line

At the moment of writing, very little of the collection is on display, but there is an on-going project to put much online (http://museumvictoria.com.au /collections/).

3 Do You Remember the War?

In the early days of electronic computing, there was rivalry between digital and analogue computing. There was no certainty as to which type of electronic computer would be dominant or that one type would dominate the other.

Analogue computers were easier to program and were faster in solving certain classes of problems. This type of computer uses physical quantities as an analogue of mathematical quantities. The Phillips Analogue Computer, for example, used water to model financial liquidity (literally!).

Advances in electronics were not biased towards one form of computing or the other. Electronics made digital electronic computers practicable, while enabling analogue computers to use electrical voltage to represent and manipulate mathematical values.

Analogue computers provided a particularly easy way of performing integration and thus they were readily used for solving differential equations arising from dynamic systems in areas such as circuit design, chemical reactions kinetics, chemical plant simulation and structural analysis, automotive, aircraft or spacecraft design, and electrical power grid modelling. (Museum Victoria 2011c). Analogue computers also provided a more direct hands-on approach to investigating system variables and more direct continuous graphical output of results.

However, with later advances in the ability to store and manipulate information in digital form, digital computers won the contest.

3.1 The Last of the First: CSIRAC, 1949-1964[5]

The CSIR Mk1 (later CSIRAC) was built by Trevor Pearcey and Maston Beard and became operational in 1949. It was Australia's first stored-program computer, and the world's fourth, being used at the CSIRO (Commonwealth Scientific and Industrial Research Organisation) Division of Radiophysics in the University of Sydney and, from 1955 at the University of Melbourne until 1964 (Pearcey 1988). In 1951, CSIRAC was the first computer ever to play music[6].

At Melbourne, CSIRAC provided a computing service for science and industry, operating for approximately 30,000 hours and tackling around 700 projects. These included calculations for weather forecasting, forestry, loan repayments, building design, psychological research and electricity supply.

CSIRAC used mercury delay line storage with a total capacity of 1024 words and an access time of 10 milliseconds. Programs were stored and loaded on paper tape and the operator, who was often the programmer, commanded a bank of switches, which had to be manipulated during program execution.

After being decommissioned, CSIRAC was immediately donated to Museum Victoria. It was already recognised as an icon as it had been in service since 1949. (McCann and Thorne 2000).

---

[5] 'The last of the first' is the title of a very comprehensive book about CSIRAC, compiled by Peter Thorne and Doug McCann, published by the University of Melbourne.

[6] The project to resurrect he music of CSIRAC is fully detailed in the book by Paul Doornbusch entitled 'The Music of CSIRAC' (Doornbusch 2005).

Fig. 1. CSIRAC and Trevor Pearcey
Photo courtesy of Museum Victoria

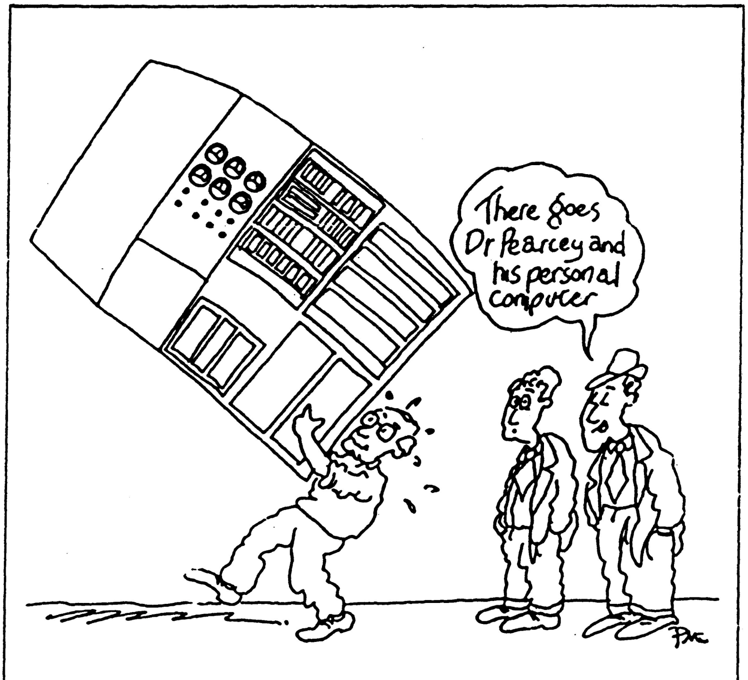


Fig. 2. Cartoon of CSIRAC and Trevor Pearcey, 1992 (ACS Victorian Bulletin)
Picture courtesy of Museum Victoria

3.2 Getting into Gear; Automatic Totalisators and Bombing Computers

The first automatic totalisator system was invented in 1913 by George Julius (1873-1946). It was set up at Ellerslie Racecourse, New Zealand. Julius established Automatic Totalisators Ltd. (A.T.L) to manufacture and successfully market the machine around the world. The last one, in London, was switched off in 1987.

Museum Victoria has parts from two different totalisators, which employed electromechanical switching. One is an electromechanical calculating machine, manufactured by Automatic Totalisators Ltd of Sydney, Australia, circa 1926. Known as the 'Julius Horse Adder', the totalisator was installed in the Mentone Race Club, Victoria, where it remained in use until its closure. From 1947 it was used at the Ipswich Amateur Turf Club, Queensland, until the machine was phased out in 1978.

Museum Victoria also has two electromechanical bombsight computers (sic)[7], designed by H&B Precision Engineers, Manchester for use with bombsights in Canberra jet bombers; one for use in the T4 Canberra bomber (1952) and the other for the Mark 8 279 bombsight (1959) (Museum Victoria 2011a). Both were manufactured for the Royal Australian Air force.

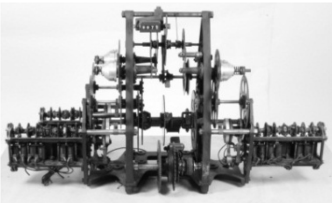


Fig. 3. Section of gearing from a totalisator installation. Believed to be part of the first Melbourne totalisator, 1931.

Fig. 4. Electro- mechanical Totalisator: 'Julius Horse Adder', c 1926.

Fig. 5. Bombsight computer

Photos courtesy of Museum Victoria

3.3 Establishing the Grid: The Westinghouse Network Analyser, 1950

The State Electricity Commission of Victoria (SECV) used an analogue computer, the Network Analyser, for the development of the Victorian Power System from 1950 into the 1960s.

In the 1950s, population and industry expanded in Australia and the standard of living also rose. The SECV responded by developing a grid system, as previously each population and industrial centre had been supplied independently.

The Network Analyser was used to model the connection between the Latrobe Valley and Melbourne and from Melbourne to NSW as the development of complex power systems was too difficult and time consuming to do using hand calculators. (Museum Victoria 2001).

[7] Electromechanical devices are sometimes called computers or computors. The term 'computer' is currently used to describe a stored program electronic machine; the first one being the 'Baby', which ran a program for the first time in 1948.

The Analyser had some advantages over digital devices; it used a step-by-step process in which designers could develop an overall understanding of the system being modelled. With an electronic computer, the whole network was produced in one stage and the designers then had to unravel it. The Analyser played a vital role in the power grid, which was the actual basis of everyday life in Victoria and Australia since. Water and sewerage grids were also established using non-digital technology.

An announcement in the Melbourne Argue newspaper at the time (The Argus 1950) spoke of the Network Analyser as an 'electric brain' costing £40,000:

> *"An 'electric brain', the only one of its kind in the British Empire, will arrive in Australia in four months for the State Electricity Commission. It can solve in a few days electrical problems associated with complicated power systems which would take two skilled electrical engineers a year to work out. Technicians simply feed in the data and the machine supplies the answers."*

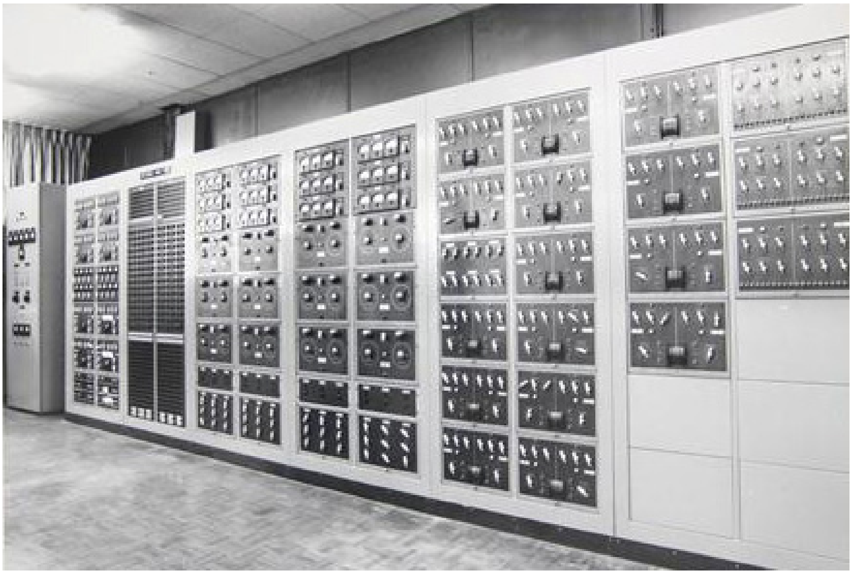


Fig. 6. Network Analyser – Westinghouse Electric Corporation
Photo courtesy of Museum Victoria

3.4 Comparing Potentials – The Melbourne University Dual Package Analogue Computer (MUDPAC), 1961 – 1986

This analogue computer was principally used for the solution of engineering problems, for designing control systems and modelling large scale dynamic systems, by the Electrical Engineering Department at Melbourne University; it cost $70,000 (University of Melbourne 2004).

It consisted of two consoles with patch-boards and used vacuum tubes. It was in fact two computers, which could be used separately or together. When the two consoles were wired together, they provided more processing power. Output was mainly on strip-charts and sometimes on oscilloscopes. Users of the analogue

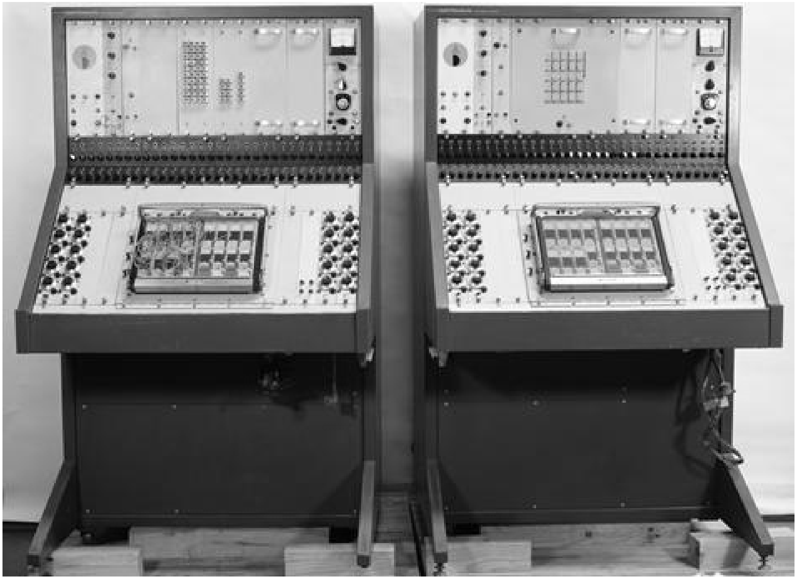


Fig. 7. MUDPAC Analogue Computer
Photo courtesy of Museum Victoria

computer programmed the machine by making connections between the different functional components, such as amplifiers, integrators and potentiometers, using a patchboard. (Museum Victoria 2011c).

4 Squeezing More and More into Less and Less

The first generation of computers were huge and operated using electronic valves. The term 'mainframe' originally referred to the large cabinets housing the central processing unit and main memory of early computers. They were installed in special rooms of their own**.**

The second generation of computers (mid-1950s to mid-1960s) processed data both in real-time and in batches. With the coming of miniaturisation; some computers left their dedicated spaces and entered the office.

Third generation computers (mid-1960s to present) excelled at multiprocessing and multiprogramming. This is the time when large machines were joined by the personal computer. The true power was soon hidden away in servers; mainframes also became smaller and more compact, even if some, the supercomputers, still occupied whole floors.

4.1 The Baby Grows up: The Ferranti Sirius, 1961

The Ferranti Mark 1, probably the world's first commercially available general-purpose computer, was produced in 1961 in partnership with Manchester University, based on the university's development of the very first electronic stored program computer – the Manchester 'Baby', in 1948. (Computer History Museum 2012).

The Ferranti Sirius is a small business computer, designed to be used in a normal office environment.

The Sirius had sufficient versatility, speed and reliability to carry out the requirements of a wide range of organisations that, did not require the larger installations needed by mainframes. Sirius' reduced physical size and increased capacity was due to the development of miniaturisation as well as the increasing sophistication of the computer industry.

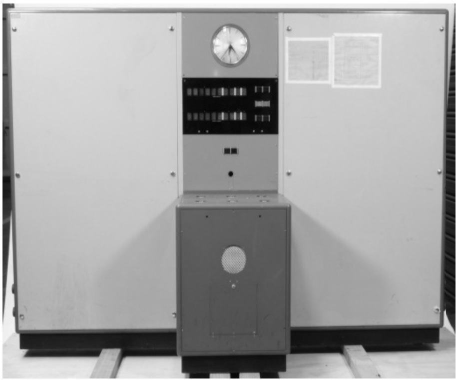


Fig. 8. Ferranti Sirius (section)

Photo courtesy of Museum Victoria, the computer is in the Museum's Science Works Store

4.2 Networking: The Control Data Model 3200, 1964

The Museum's CDC 3200 computer represents the early period of the adoption of digital technology by Australia by institutions such as Universities. It also symbolises the early stages of the development of computer networks in Australia.

Fig. 9. Disk drive; part of a Control Data model 3200 computer system

Photo courtesy of Museum Victoria

Control Data Corporation and CDC 3200 computers played a vital part in the setting up of two major computer networks at a time where there was no such term as computer network; (the first internet-type networks were developed in the late 1960s and early 1970s). The network covered Adelaide, Sydney and Melbourne.

CDC3200 computers were used by the Bureau of Census and Statistics to carry out a business data processing operation. The CSIRO employed the CDC3200 to provide a scientific computing service to its forty or so divisions as well as engaging in computing research.

The machine in the Museum's collection was used by academic and administrative staff, and students at Monash University in the 1960s. It was operational from 1964 to 1979; it had replaced a Sirius computer (Museum Victoria 2012a).

4.3 Making the Transition: Pacific Data Systems PDS 1020 Mini Desktop, 1964

A large desktop console, this machine is an example of the transition from floor mounted computers with work stations (minicomputers, for example the IBM System

3) to microcomputers with the standard configuration of box, screen, keyboard and later, mouse.

It used paper tape for data and program input, took about half an hour to warm up, and was built completely with solid-state circuitry.

The PDS 1020 was described by the manufacturer as an engineering digital computer that combined slide-rule convenience and calculator simplicity with the versatility and capacity of the modern general purpose digital computer. It could be used to solve a broad spectrum of engineering problems (Museum Victoria 2012c).

The PDS 1020 employed a delay line memory. Delay line memories were used in early computers such as CSIRAC, and were an important technological development from World War II.

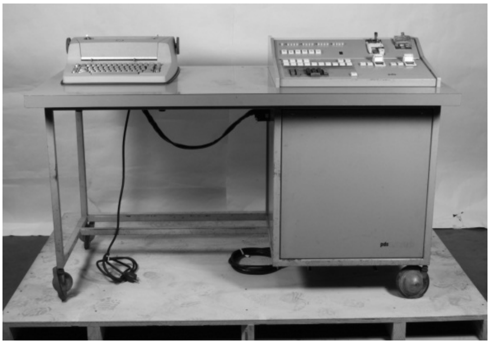


Fig. 10. Pacific Data Systems PDS 1020 mini desktop computer. Desktop with typewriter and paper tape input/output.

Photo courtesy of Museum Victoria

4.4 In the Pink: The IBM System 3, 1975

The IBM System 3 computer is intermediate in size and capacity between mainframe and minicomputers. The computing system was designed to look attractive in an office while having near to mainframe potential. This computer was used for accounts and scheduling by City of Camberwell.

Fashion was one reason why this computer system was acquired by the Museum; the machines were given a distinctive pink colour (Museum Victoria 2012b).

This computer system is typical of the period around the 1970s when firms started to target the computing needs of small businesses, machines being designed to fit into an office environment rather than requiring a room of their own as did mainframes. Up to that time, people expected computers in neutral colours

Fig. 11. IBM System 3

Photo courtesy of Museum Victoria

for example, IBM machines were given neutral colours such as beige. However, some people found the pink to be disconcerting.

5 Getting More PC – The Return of Hands-On

The first generation computers were definitely hands-on; they needed a lot of human input, even when they were running programs. The arrival of the PC returned the control and power back to the digits.

5.1 Heralding Changes: The IBM Model 1130, 1968

When released in 1965, the IBM 1130 computer system was IBM's lowest cost system and was intended for the technical and engineering market. A large desktop console, this machine is an example of the transition from floor mounted computers with workstations (minicomputers, e.g. the IBM System 3) to microcomputers with the standard configuration – box, screen, keyboard and, later, mouse.

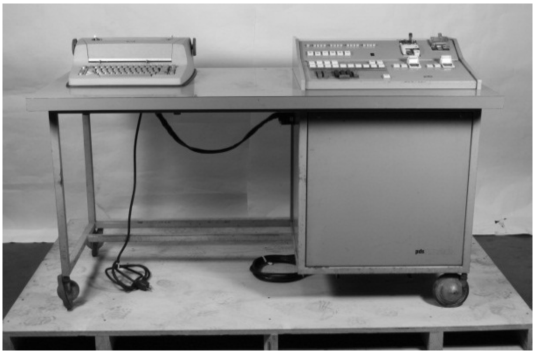


Fig. 12. Magnetic disc memory system, keyboard/printer/ control panel on desk, part of IBM 1130 computer system

Photo courtesy of Museum Victoria

This machine played a role in introducing computers to the Australian newspaper industry. The machine was used by the Herald and Weekly Times.

The period was marked by cumbersome efforts made to 'automate' existing practices. The words were typed in from journalists' copy, a magnetic memory generated, and a paper type ribbon developed which was sent to the hot metal type setter. If any mistakes were made, then the whole paper tape had to be redone.

5.2 A Machine for all Seasons: The DEC PDP-8 Data Processor Minicomputer, Circa 1968

The PDP-8 was probably the first successful general purpose commercial minicomputer, and was introduced by the United States firm Digital Equipment Corporation (DEC) in 1965.

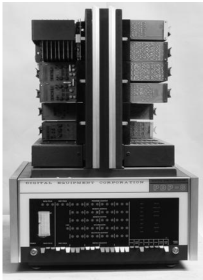


Fig. 13. PDP-8 minicomputer

Photo courtesy of Museum Victoria

The machine was used by the CSIRO Division of Trobophysics (later Material Science) for examining, for example, crystal structure and phenomena associated with friction. The computer was also used by Monash University (Museum Victoria 2011a).

The machine used discrete transistor circuitry and was designed by Gordon Bell (Slater 1989). More than 50,000 systems were sold, a record at that time.

5.3 Holistic Medicine: The Searle Medical Computer, 1971-1987

The Searle Medical Computer is an early example of the use of digital technology in medicine; it was donated to the Museum by the Shepherd Foundation.

The Shepherd Foundation opened its Automated Multiphasic Health Testing (AMHT) Centre in South Melbourne in May 1971. AMHT was used for patient data gathering, analysis, storage, and print-out. George Fredrick Shepherd, founder of the Sheppard Foundation, believed that everyone was entitled to an annual medical check-up at a reasonable cost.

The PDP8/1 computer (see previous paragraph), was the heart and brain of the Centre. Searle Medidata of Waltham, Massachusetts developed the software.

The first stage of the testing was referral of a patient by their doctor. Then, during the patient's visit to the Centre, a computerised questionnaire was completed, covering many aspects of the patient's medical and personal history. The AMHT profile included information on electrocardiograph, pulse, blood pressure, anthropometry, Achilles tendon reflex relaxation time, chest x-ray, audiometry, hearing, vision, blood biochemistry, full blood examination, urine chemical and dipslide culture. A report, including advice, was sent to the referring doctor. (Sheppard Foundation 2011).

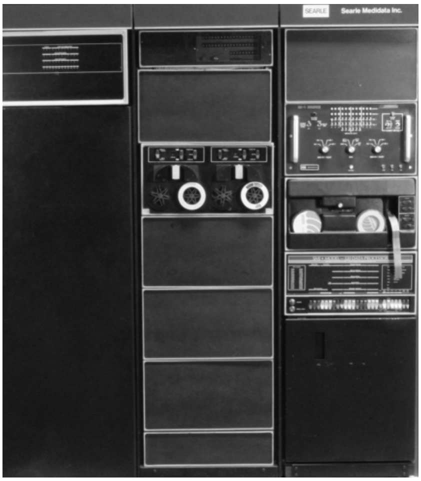


Fig. 14. Searle Medidata Medical Computer based on PDP-8/1

Photo courtesy of Museum Victoria

The initial charge was AU$50, most of which was covered by a Government health scheme (Museum Victoria 2011b). For 16 years, the Centre flourished; among its many achievements, the two that stand out relate to female patients – the introduction of mammography for women over 50 years or age and of pap smears (Museum Victoria 2012e; Museum Victoria 2012d).

The history of AMHT was marked by controversy. Some members of the medical profession saw AMHT as a form of population health screening with possible adverse effects such as incorrect or over-diagnosis. Others viewed it as a valuable tool dealing with the whole health care process.

6 Reviving Old Memories – Yesterday and Tomorrow

6.1 Yesterday . . .

In the late 1990s, the CSIRAC History Team developed modern equipment to read CSIRAC's obsolescent paper tapes[8]. A project to re-construct the CSIRAC music was the stimulus for the construction of this system to read the original CSIRAC paper tapes (Demant 2010). Equipment was set up to read the paper tapes and save the data in electronic form. The equipment was assembled in 1997, long after CSIRAC had ceased all operations in 1964. CSIRAC used to read paper tapes using paper tape readers of a a very different design and vintage.

The project gives us hope that younger but still obsolescent software can be resurrected . . . well, at least the worthwhile stuff.

6.2 . . . and Tomorrow

A first generation iPad, 2010, was acquired by the Museum in 2012. This particular iPad was one of over 500 iPads, distributed in June 2010 to eight Victorian schools as part of an initiative by the Victorian Government to trial the impact of iPads on learning and teaching.

Fig. 15. The launch of the iPad School Project by the Department of Education, Victoria, 2010 Photo courtesy of Museum Victoria

[8] The project was developed in order to recreate the music that CSIRAC had been programmed to generate. CSIRAC was the first computer ever to do this, in 1950.

The participating schools included the Victorian College of the Arts Secondary School, which donated this iPad to Museum Victoria in 2012. The programme, utilising the iPad for learning in schools, was still in operation at the time this item was donated to the Museum.

One of the children, who participated in that project, will one day see that iPad, and their grandchild will exclaim: " . . *A what? . . . And you . . . actually touched it ... with your finger!?*"

References

Computer History Museum. Selling the Computer Revolution - Ferranti Limited (2012), http://www.computerhistory.org/brochures/companies.php?alpha=d-f&company=com-42bc1acec3452 (retrieved June 2012)

Demant, D.: Why the Real Thing Is Essential for Telling Our Stories. In: Tatnall, A. (ed.) HC 2010. IFIP AICT, vol. 325, pp. 13–15. Springer, Heidelberg (2010)

Doornbusch, P.: The Music of CSIRAC - Australia's First Computer Music. Common Ground Publishing, Melbourne (2005)

McCann, D., Thorne, P.: The Last of the First - CSIRAC: Australia's First Computer. The University of Melbourne, Melbourne (2000)

Museum Victoria. Network Analyser - Westinghouse Electric Corporation, Pittsburgh, USA 1950 (2001), http://museumvictoria.com.au/collections/items/1763754/network-analyser-westinghouse-electric-corporation-pittsburgh-usa-1950 (retrieved May 2012)

Museum Victoria. Computer - DEC, PDP-8, circa 1968 (2011a), http://museumvictoria.com.au/collections/items/374302/computer-dec-pdp-8-circa-1968 (retrieved May 2012)

Museum Victoria. Medical Computer - Searle, PDP8/1, 1971-1987 (2011b), http://museumvictoria.com.au/collections/items/406935/medical-computer-searle-pdp8-1-1971-1987 (retrieved May 2012)

Museum Victoria. The Melbourne University Dual Package Analogue Computer: MUDPAC (2011c), http://museumvictoria.com.au/collections/themes/5592/the-melbourne-university-dual-package-analogue-computer-mudpac (retrieved May 2012)

Museum Victoria. Computer - Control Data Model 3200, 1964 (2012a), http://museumvictoria.com.au/collections/items/396846/computer-control-data-model-3200-1964 (retrieved May 2012)

Museum Victoria. Computer - IBM, System 3, circa 1975 (2012b), http://museumvictoria.com.au/collections/items/400457/computer-ibm-system-3-circa-1975 (retrieved May 2012)

Museum Victoria. Computer System - Pacific Data Systems, PDS 1020 Mini Desktop, circa 1964 (2012c), http://museumvictoria.com.au/collections/items/397741/computer-system-pacific-data-systems-pds-1020-mini-desktop-circa-1964 (retrieved May 2012)

Museum Victoria, Information Folder - Medidata Automated Multiphasic Health Testing Systems, Searle Medical Computer, PDP8/1 (2012d), http://museumvictoria.com.au/collections/items/1827698/information-folder-medidata-automated-multiphasic-health-testing-systems-searle-medical-computer-pdp8-1-1970 (retrieved May 2012)

Museum Victoria. Sales Brochure - Medidata Automated Multiphasic Health Testing Systems, Searle Medical Computer, PDP8/1 (1971) (2012e), http://museumvictoria.com.au/collections/items/1827672/sales-brochure-medidata-automated-multiphasic-health-testing-systems-searle-medical-computer-pdp8-1-1971 (retrieved May 2012)

Pearcey, T.: A History of Australian Computing. Chisholm Institute of Technology, Melbourne (1988)

Slater, R.: Portraits in Silicon. MIT Press, Boston (1989)

The Argus, Robot "Brain" for S.E.C. cost £40,000. The Argus, p. 6. National Library of Australia, Melbourne (1950)

University of Melbourne. Melbourne School of Engineering - 1960s (2004), http://www.eng.unimelb.edu.au/MSE150/1960s.html (retrieved May 2012)

The Changing Face of the History of Computing: The Role of Emulation in Protecting Our Digital Heritage

David Anderson, Janet Delve, and Vaughan Powell

Future Proof Computing Group, School of Creative Technologies,
University of Portsmouth, UK
cdpa@btinternet.com,
{Janet.Delve,vaughan.powell}@port.ac.uk

Abstract. It is becoming increasingly common for some source material to arrive on our desks after having been transferred to digital format, but little of the material on which we work was actually born digital. Anyone whose work is being done today is likely to leave behind very little that is not digital. Being digital changes everything. This article discusses the issues involved in the protection of digital objects.

Keywords: digital preservation.

Erasmus summarised well the outlook of many historians of computing when he remarked "*When I get a little money, I buy books; and if any is left I buy food and clothes*"! Most of us love second-hand bookshops, libraries and archives, their smell, their restful atmosphere, the ever-present promise of discovery, and the deep intoxication produced by having the accumulated knowledge of the world literally at our fingertips. Our research obliges us to spend many happy hours uncovering and studying old letters, notebooks, and other paper-based records. It is becoming increasingly common for some source material to arrive on our desks after having been transferred to digital format, but little of the material on which we work was actually born digital. Future historians of computing will have a very different experience. Doubtless they, like us, will continue to privilege primary sources over secondary, and perhaps written sources will still be preferred to other forms of historical record, but for the first time since the emergence of writing systems some 4,000 years ago, scholars will be increasingly unable to access directly historical material. During the late 20th and early 21st century, letter writing has given way to email, SMS messages and tweets, diaries have been superseded by blogs (private and public) and where paper once prevailed, digital forms are making inroads, and the trend is set to continue. Personal archiving is increasingly outsourced – typically taking the form of placing material on some web-based location in the erroneous belief that merely being online assures preservation. Someone whose work is being done today, is likely to leave behind very little that is not digital, and being digital changes everything.

Digital 'objects' are, in many ways, a curious product of the 20th Century and have attributes that make them quite unlike any of the objects that preceded them. Their immaterial nature gives rise to diverse concerns for those tasked with preservation

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 362–384, 2012.

activity, requiring substantial changes to be made to preservation practice as well as demanding a significant alteration in the way we think about the nature of objecthood in the digital context.

> *"We depend on documents to carry messages through space and time. In many cases, this reliability is achieved through fixity: letterforms inked on paper can survive for long periods of time. But with newer media, such as video, this reliability is achieved not by fixity but by repeatability. The moving images on a video screen are by their very nature transient. I will never be able to see those very images again. But I can play the tape repeatedly, each time seeing a performance that, for all practical purposes, is 'the same as' the one I saw the first time." (Levy 2000)*

Digital objects, unlike their physical counterparts, are not capable directly of human creation or subsequent access but require one or more intermediate layers of facilitating technology. In part, this technology comprises further digital objects; software such as a BIOS, an operating system or a word processing package, and in part it is mechanical; a computer. Even a relatively simple digital object such as a text file (ASCII format) has a series of complex relationships with other digital and physical objects from which it is difficult to isolate it completely. This complexity and necessity for technological mediation exists not only at the time when a digital object is created but is present on each occasion when the digital object is edited, viewed, preserved or interacted with in any way. Furthermore the situation is far from static as each interaction with a digital object may bring it into contact with new digital objects (a different editor for example) or new physical technology.

The preservation of, and subsequent access to, digital material involves a great deal more than the safe storage of bits. The need for accompanying metadata, without which the bits make no sense, is understood well in principle and the tools we have developed are reasonably reliable in the short term, at least for simple digital objects, but have not kept pace with the increasingly complex nature of interactive and distributed artefacts. The full impact of the lacunae will not be completely apparent until the hardware platforms on which digital material was originally produced and rendered, become obsolete, leaving no direct way back to the content.

Within the digital preservation community the main approaches usually espoused are migration, and emulation. The very great extent to which opinions appear to be polarised between these two approaches (Bearman 1999; Rothenberg 1999) – neither of which can claim to be a complete solution – is perhaps indicative of the extent to which the fundamental issues of long term digital preservation have not yet been addressed (Stawowczyk Long 2009). The arguments and 'evidence' offered on both sides of the debate are frequently far from convincing.

The focus of migration is the digital object itself, and the process of migration involves changing the format of old files so that they can be accessed on new hardware (or software) platforms. Thus, armed with a suitable file-conversion program it is relatively trivial (or so the argument goes) to read a WordPerfect document originally produced on a Data General mini-computer some thirty years ago, on a brand new iPAD3. The story is, however, a little more complicated in practice. There is something in excess of 6,000 known computer file formats, with more being produced all the time, so the introduction of each new hardware platform creates a potential need to develop afresh thousands of individual file-format convertors in order to get access to old digital material. Many of these will not be produced for lack of interest among those with the technical knowledge to develop them, and not all of the tools which are created will work perfectly. It is fiendishly difficult to render with complete fidelity every aspect of a digital object on a new hardware platform. Common errors include variations in colour mapping, fonts, and precise pagination. Over a relatively short time, in a digital version of 'whisper down the lane' errors accumulate and erode significantly our ability to access old digital material or to form reliable historical judgements based on the material we can access. The cost of storing multiple versions of files (at least in a corporate environment) means that we cannot always rely on being able to retrieve a copy of the original bits.

The challenges represented by converting a WordPerfect document are as nothing compared to those of format-shifting a digital object as complex as a modern computer game, or the special effects files produced for a Hollywood blockbuster. This fundamental task is well beyond the technical capability or financial wherewithal of any library or archive. While it is by no means apparent from much of the literature in the field, it is nevertheless true that in an ever-increasing number of cases, migration is no longer a viable preservation approach.

Emulation substantially disregards the digital object, and concentrates its attention on the environment. The idea here is to produce a program which when run on one environment, mimics another. There are distinct advantages to this approach: it avoids altogether the problems of file format inflation, and complexity. Thus, if we have at our disposal, for example, a perfectly functioning IBM 360 emulator, all of the files which ran on the original hardware should run without modification on the emulator. Emulate the PS3, and all of the complex games which run on it should be available without modification – the bits need only be preserved intact, and that is something which we know perfectly well how to accomplish. Unfortunately, producing perfect, or nearly perfect emulators, even for relatively unsophisticated hardware platforms is not trivial. Doing so, involves not only implementing the documented characteristics of a platform but also its undocumented features. This requires a level of knowledge well beyond the average and, ideally, ongoing access at least one instance of a working original against which performance can be measured. Over and above all of this, it is critically important to document for each digital object being preserved for future access, the complete set of hardware and software dependencies it has and which must be present (or emulated) in order for it for it to run[1]. Even if all of this can be accomplished, the fact remains that emulators are themselves software objects written to run on particular hardware platforms, and when those platforms are no longer available they must either be migrated or written anew. The EC funded KEEP project[2] has recently investigated the possibility of squaring that particular circle by developing a highly portable virtual machine onto which emulators can be placed and which aims to permit rapid emulator migration when required. It is too soon to say how effective this approach will prove, but KEEP is a project which runs against the general trend of funded research in preservation in concentrating on emulation as a preservation approach and complex digital objects as its domain.

---

[1] See TOTEM `http://www.keep-totem.co.uk/`

[2] `http://www.keep-project.eu`

Even in a best-case-scenario, future historians, whether of computing or anything else, working on the period in which we now live will require a set of technical skills and tools quite unlike anything they have hitherto possessed. The vast majority of source material available to them will no longer be in a technologically independent form but will be digital. Even if they are fortunate enough to have a substantial number of apparently well-preserved files, it is entirely possible that the material will have suffered significant damage to its intellectual coherence and meaning as the result of having been migrated from one hardware platform to another. Worse still, digital objects might very be left completely inaccessible in virtue of either not having a suitable available hardware platform on which to render them, or rich enough accompanying metadata to make it possible negotiate the complex hardware and software dependencies required.

It is commonplace to observe ruefully on the quantity of digital information currently being produced. Unless we begin seriously to address the issue of future accessibility of stored digital objects, and take the appropriate steps to safeguard meaningfully our digital heritage, future generations may have a much more significant cause for complaint. In addressing this challenge, it appears not unreasonable to suppose that Computer Museums might have a productive role to play:

> *"To avoid the dual problems of corruption via translation and abandonment at paradigm shifts, some have suggested that computer museums be established, where old machines would run original software to access obsolete documents (Swade 1998). While this approach exudes a certain technological bravado, it is flawed in a number of fundamental ways." (Rothenberg 1999)*
>
> *"… preserving obsolete hardware and software (a solution which supposes that complete museums of obsolete equipment could be maintained so that any configuration of once used hardware and software could be replicated. Rothenberg reiterates pragmatic arguments against this, many of which I published in a 1987 report. They were not novel then. Fortunately, as far as I know, no serious investment in this 'solution' has ever been attempted.)" (Bearman 1999)*

So it seems that one of the things about which Bearman agrees with Rothenberg, is that reliance on Computer Museums is not the answer to the problem of digital preservation. It is probably worth mentioning here that approaching digital preservation as a problem that has a single 'correct' answer, if only we could discover it, is a caricature of what is, in reality, a complex and multi-faceted series of challenges. Generally, it should be observed, Rothenberg and Bearman write in somewhat polemical terms, and in very large measure this detracts from their cases.

Rothenberg's concerns about computer (hardware) museums essentially come down to a single point: hardware (including media) will deteriorate over time to the stage where old machines will not be able to access the software written for them (Rothenberg 1999). The long-term inevitability of physical deterioration of systems will not come as a surprise to anyone familiar with the 2^{nd} Law of Thermodynamics[3].

---

[3] The 2^{nd} Law of Thermodynamics asserts the universal principle that absolutely everything decays.

On the positive side, Rothenberg concedes two 'limited roles' for computer (hardware) museums: performing *"…heroic efforts to retrieve digital information from old storage media*." and verifying the behaviour of emulators. Rothenberg's characterisation of these roles as 'limited' notwithstanding, these are, in fact, absolutely essential activities; the first is arguably the only way in which future generations can gain access to some important historical material which would otherwise be lost forever, and the second is vital if we are to verify that the behaviour of emulated computer platforms is faithful to the original. It is particularly important to be able to determine if, for example, some unexpected behaviour exhibited by a digital object running under emulation is the result of a lack of fidelity in the emulator or would have been present on the original platform.

Bearman, writing about computer (software) museums, makes a closely related point.

> *"… by documenting standards and widespread operating functions, software archives preserve a record of the fundamental structures of the software environment which will contribute to future understanding of more specialized software." (Bearman 1987)*

Similar considerations apply equally well to verifying migrated digital objects and computer museums are of equal value independently of the digital preservation approach preferred.

All computers have undocumented features, and preserving original hardware in working condition for as long as reasonably possible is an important aspect of digital preservation. Of course it is not possible to preserve computer systems forever but this only lends urgency to the need to gather as much information as possible from machines while they are available to us. The cost of funding computer museums, particularly when viewed from the national or international perspective, is not high. Rothenberg's assertion that:

> *"It is unlikely that old machines could be kept running indefinitely at any reasonable cost, and even if they were, this would limit true access to the original forms of old digital documents to a very few sites in the world, thereby again sacrificing many of these documents' core digital attributes." (Rothenberg 1999)*

somewhat misstates the position. It is certainly true that preserving old machines in working order is subject to the law of diminishing returns. That is one of the reasons why we should endeavour to make the best use of old machines while it is still feasible for us to do so.

Rothenberg appears to be insensitive to the fact that computer museums would have a continuing role into the future, broadening their collections as more and more machines become obsolete. The custodial remit of computer museums would run further than simply keeping in working order the oldest machines in their care, and it is to be expected that for each machine in a museum's collection, a decision would have to be made, at some point in the future, to preserve the device in a non-working condition. This does nothing to diminish the importance of computer museums, which could and should become repositories of collective knowledge in just the same way as any other memory institution. The 2nd Law of Thermodynamics is unavoidable and if Rothenberg's arguments were sound, they would apply with equal force to every other sort of library or museum; all statues will eventually crumble and every book ever written will, given sufficient time, turn to dust. Should we therefore conclude that it is folly to suggest a role for memory organisations in preserving our cultural, technological and scientific heritage?

It is fair to say that Rothenberg does not think highly of migration as an approach to digital preservation:

> *"While it may be better than nothing (better than having no strategy at all or denying that there is a problem), it has little to recommend it. …. however, to the extent that it provides merely the illusion of a solution, it may in some cases actually be worse than nothing. In the long run, migration promises to be expensive, unscalable, error-prone, at most partially successful, and ultimately infeasible." (Rothenberg 1999)*

Unfortunately many of the arguments that Rothenberg deploys in reaching these damning conclusions seem to apply with similar force when directed at emulation. Rothenberg opines:

> *"… migration is labor-intensive, time-consuming, expensive, error-prone, and fraught with the danger of losing or corrupting information. Migration requires a unique new solution for each new format or paradigm and each type of document that is to be converted into that new form. Since every paradigm shift entails a new set of problems, there is not necessarily much to be learned from previous migration efforts, making each migration cycle just as difficult, expensive, and problematic as the last. Automatic conversion is rarely possible, and whether conversion is performed automatically, semiautomatically, or by hand, it is very likely to result in at least some loss or corruption, as documents are forced to fit into new forms." (Rothenberg 1999)*

However emulation also requires considerable expenditure of time and effort in order to arrive at a successful outcome. It is true that a great deal of excellent work has been undertaken in the emulation community[4], which has provided benefits for the digital preservation community without giving rise thereby to any outlay of resources by museums, libraries or archives. But this software windfall should not be allowed to engender complacency. Robust emulation software remains 'labor-intensive, time-consuming, expensive' to develop. The emulators currently available to us do not provide either complete coverage of all the required hardware platforms, nor completely reliable and faithful reproduction of the machines that have been emulated. They are, in Rothenberg's way of speaking, 'error-prone, and fraught with the danger of losing or corrupting information' if we were to rely on them for our digital preservation needs. We can have no complaint however, as the available emulators are often the result of the unpaid efforts of enthusiasts, and reflect the interests and obsessions of their authors rather than being driven by digital preservation requirements.

The digital preservation community has made one attempt to develop an emulator. The Dioscuri project was conceived in 2004 and has been under continuous development since 2006. The main development has been undertaken by Nationaal

[4] For example with the Multiple Arcade Machine Emulator (M.A.M.E.) [see `http://mamedev.org/`] and the Multiple Emulator Super System (M.E.S.S.) [see `http://www.mess.org/`]

Archief of the Netherlands, the Koninklijke Bibliotheek, and Tessella plc, and has benefitted from the efforts of a number of others including Jeff Rothenberg. Financial support for the building of the emulator has been provided by the European Commission and has taken place both within the Planets project and the KEEP project.

Dioscuri is attempting to emulate a well understood and documented hardware platform (PC x86), numerous copies of which are extant. Conditions for producing an emulator really do not come much better than this. So, if emulator development were simple, even relatively so, then after six years of planning and well-funded development, Dioscuri would be complete and the digital preservation community would be reaping the benefits of its use and perhaps taking forward the lessons learned and applying them to the development of other emulators. However, writing at the end of last year, Stawowczyk Long concluded in his report for the National Library of Australia and the International Internet Preservation Consortium:

> *"Dioscuri … has very limited capabilities, It could only be tried with MS DOS 6.2 and MS Windows 3.1 operating systems. Dioscuri is … rather slow. … media files could not be rendered sufficiently well to give a useful performance." (Stawowczyk Long 2009)*

Emulators, as we have observed, are not simple to write. It is a time-consuming and highly complex activity. As things stand, neither emulation nor individual emulators have been developed to the point where emulation can seriously challenge migration as a digital preservation approach. Some important first steps towards developing emulation have been taken (e.g. in the EC funded KEEP project) , but these are the first steps only, and it is as yet to soon to say where precisely they will lead. We can, however, be reasonably certain that they will not lead to the complete replacement of migration nor, for reasons that we will cover in greater detail below, is it desirable that they should do so.

Rothenberg has a concern that the evolution of formats, encodings and software paradigms defies prediction.

> *"As has been proven repeatedly during the short history of computer science, formats, encodings, and software paradigms change often and in surprising ways. Of the many dynamic aspects of information science, document paradigms, computing paradigms, and software paradigms are among the most volatile, and their evolution routinely eludes prediction." (Rothenberg 1999)*

It is difficult to see anything in these observations that applies, in any important sense, to migration that does not also apply with equal force to emulation per se. The substance of the point appears to be that the future has often proved unpredictable and is likely to do so again. Each new hardware paradigm is apt to cause problems for all the emulators written to run on the old paradigm and there is not the least justification for believing that, for example, a Commodore 64 emulator written to run on a Mac Powerbook of the early 21st century, will be able to run, without significant conversion, on an as yet unconceived of hardware platform. The KEEP project is directly focussed on just this aspect of emulator portability and it is hoped that progress will be made towards keeping emulator environments portable. However, there is nothing about emulation in and of itself, which makes it immune to the disruption caused by the introduction of new approaches to computing and the inevitable obsolescence of the old.

Even if we were to take seriously Rothenberg's description of the problem, it would remain an open question as to whether an emulation approach to digital preservation were the best, response; it is certainly not the only route open to us. Gladney, for example, has very plausibly suggested a shift towards the routine development of durable digital objects coupled with moving the responsibility for digital object durability away from archival employees to information producers (Gladney 2008).

Another concern to which Rothenberg gives voice is that migration (unlike emulation) involves urgency.

> *"… there is a degree of urgency involved in migration. If a given document is not converted when a new paradigm first appears, even if the document is saved in its original form (and refreshed by being copied onto new media), the software required to access its now-obsolete form may be lost or become unusable due to the obsolescence of the required hardware, making future conversion difficult or impossible."*

To the degree that Rothenberg has a point here it is difficult to see why it does not apply with at least equal force to emulation. Let us leave aside the fact that he has not provided any evidence of previous hardware paradigm shifts causing the sort of data loss or inaccessibility to which he alludes – after all disaster might be waiting to strike at any moment. Let us also leave aside the fact that recent experience within the digital preservation community indicates that the amount of migration intervention that has actually been required was less than expected – the future may be much worse than the past in this respect. Rothenberg does not make it clear why migration involves a peculiar degree of urgency. The introduction of a new hardware paradigm would mean that every emulator written to run on the previous paradigm would no longer function on the new device. This would leave us with two options:

- Write new emulators
- Migrate the old emulators to run on the new platform

The first option is at least as complicated as having to develop new migration tools, while the second is itself a form of migration and must, on that account, be susceptible to any problems which migration faces. The KEEP approach represents a version of option two. KEEP's proposal is to develop a virtual machine as the platform on which emulators will be written (or ported) to run, but which is designed in such as way as to be migratable without difficulty to any conceivable hardware platform of the future. Thus, the plan is to ensure that emulated environments once written can be kept portable. While KEEP's focus is on the portability of emulators, it should not be forgotten that just as emulators can be written to run on the KEEP Virtual Machine, so could other applications such as word processors, spreadsheets, databases etc. Therefore one possible way of ensuring relatively easy migration of files and applications from one hardware platform or paradigm onto another would be to target them for an easily portable virtual machine. Rothenberg is not alone in speaking about emulation and migration as if they were two diametrically opposed approaches. We regard this as a false dichotomy. Migration and emulation are better viewed as complementary approaches. Some digital objects are more amendable to migration than others; ascii files migrate much more easily than complex interactive digital objects such as games. Even if some digital objects prove, in practice, to be intractable to migration, the emulators on which they run will either have to be themselves migrated or completely replaced at some time in the future. A future in which emulation completely displaces migration is not one that we take seriously. On the whole it would be much better if the migration and emulation 'camps' sought common ground.

Migration (unlike emulation) is, on Rothenberg's account, at a disadvantage because it is an ongoing activity.

> *"Worse yet, this problem does not occur just once for a given document (when its original form becomes obsolete) but recurs throughout the future, as each form into which the document has migrated becomes obsolete in turn. Furthermore, because the cycles of migration that must be performed are determined by the emergence of new formats or paradigms, which cannot be controlled or predicted, it is essentially impossible to estimate when migration will have to be performed for a given type of document – the only reliable prediction being that any given type of document is very likely to require conversion into some unforeseeable new form within some random (but probably small) number of years." (Rothenberg 1999)*

Rothenberg is quite correct to draw our attention to the fact that our current digital documents are such that it is not possible merely to save their bitstreams and periodically refresh them in order to secure their access for future generations. As indicated above, a shift in hardware platforms or the change of a paradigm requires preservation action to be taken. But this is true whether the preservation strategy is based on migration or emulation. Indeed whatever the preservation strategy employed (other than choosing not to preserve) it would be possible to think of circumstances in which regular preservation action might be required. Rothenberg's observation is not therefore pertinent to migration alone but is more by way of a statement of how the world works, and as such need not be considered further in the present context. Similar remarks apply to Rothenberg's concerns about the 'unpredictability' of the occurrence of circumstances that will necessitate a preservation intervention. It is to be expected that the mean time between preservation interventions is shorter with some preservation strategies than others. It is difficult to see how this could be proven in advance or determined with any great accuracy but if this information were available it would be valuable (but not decisive) in helping to determine the strategy adopted by individual institutions. Ex hypothesi, nothing except good fortune, can protect us from the unexpected or the unpredictable; so it is reasonable to conclude that however technically brilliant an emulation strategy might be developed, it too will be vulnerable to unforeseen circumstances.

Rothenberg is concerned that migration (unlike emulation) is a piecemeal activity.

> *"Since different format changes and paradigm shifts affect different (and unpredictable) types of documents, it is likely that some of the documents within a given corpus will require migration before others. … This implies that any given corpus is likely to require migration on an arbitrarily (and uncontrollably) short cycle, determined by whichever of the component types of any of its documents is the next to be affected by a new format or paradigm shift." (Rothenberg 1999)*

Rothenberg places a great deal of the weight of his argument on the notion that the future is unknowable. He decorates this relatively banal observation with unsupported claims that computing paradigms are particularly susceptible to unforeseen change. Here his variation on the theme of unpredictability concentrates on the notion that some unforeseen changes will impact one type of document more than another. He is quite right of course; indeed he is rather more right than he cares to admit. After all, there are changes likely to occur in the future but about which we presently know nothing that will affect some emulators more than others, or some aspects of some emulators more than others. So what? Those responsible for preserving digital materials for access by future generations will have to respond to whatever circumstances arise. Choosing an emulation approach does not provide a safeguard against the unpredictable and Rothenberg has not offered any grounds for thinking that a migration approach will leave the digital preservation community particularly exposed to an uncertain future.

According to Rothenberg, migration (unlike emulation) does not scale well.

> *"Finally, migration does not scale well. Because it is labor-intensive and highly dependent on the particular characteristics of individual document formats and paradigms, migration will derive little benefit from increased computing power. It is unlikely that general purpose automated or semiautomated migration techniques will emerge, and if they do, they should be regarded with great suspicion because of their potential for silently corrupting entire corpora of digital documents by performing inadvertently destructive conversions on them. As the volume of our digital holdings increases over time, each migration cycle will face a greater challenge than the last, making the essentially manual methods available for performing migration increasingly inadequate to the task." (Rothenberg 1999)*

The effort expended in producing an emulator sufficiently robust and reliable to plays a significant role in the context of managed digital preservation is, as we have seen, substantial. The further away in time we move from having access to the original hardware which is being emulated, the less confidence we have in the accuracy of each new emulator. Once the original hardware is no longer available for inspection and comparison we are (at best) left with comparing the behaviour of the newly developed emulators with the performance of the most trusted previous emulator, and this is liable, over time, to result in silent degradation of emulator performance. The question of whether migration involves a greater or lesser amount of human resource than emulation is entirely empirical. Rothenberg is quite mistaken in thinking that the matter can be settled in advance as a matter of principle. Furthermore, it is likely that at some points in time migration will involve less human effort than emulation while at others the situation will be reversed.

We can be reasonably confident that the performance of emulators will be better on faster machines having greater computing resources available to them, but similar expectations must surely be reasonable concerning migration as well. Is it not likely that decision support systems and artificial intelligence driven programs running on future machines will increasingly automate the process of migration? Rothenberg's pessimism is not well supported by argument and seems more axiomatic than reasoned.

Rothenberg is correct to point out that each migration cycle presents new challenges but we simply cannot know in advance if these will always be more taxing than the cycles that have gone before. Each emulation cycle also presents new challenges but in the absence of further information it is idle to speculate concerning the extent to which we might be facing an upward spiral of complexity or some other pattern.

Rothenberg's judgement is that migration is essentially an approach based on wishful thinking but this harsh assessment is not supported by persuasive argument. Such worthwhile points as he has made apply with more or less equal force to his preferred approach of using emulation for digital preservation. He seems either unaware of, or unconcerned about, the fact that emulators are themselves digital objects which, if they are to continue to be of use, must either be migrated on to future hardware platforms or run under another layer of emulation. The effect of this simple fact is to undermine a great deal of what Rothenberg says. Curiously, Rothenberg fails to voice a great deal of what might be said in support of emulation. For example, he does not place enough weight on the notion that migration is better suited to simple digital objects than to the more complex challenges increasingly faced by those responsible for preservation within an institutional context. Too much subtlety is sacrificed in order to draw clear 'battle lines' with the advocates of migration. The KEEP project recognises that value of emulation in dealing with complex digital objects but is also aware of both the practical necessity of migration and its continuing value to end users. Even if it were possible to arrive at a situation in which an emulation could be deployed for all digital objects it would still remain the case that migration would have a role to play. There are numerous situations in which end users would prefer access to files via migration rather than emulation. For example, in making use of old research material in the context of a new report it is usually more desirable to migrate the old work for incorporation in the new than to ask for the material to be made available under emulation. Context is everything and there will continue to be demand for both migration and emulation for as far into the future as we can see.

The depiction of digital preservation as being faced with a choice between migration and emulation is not only a false dichotomy; it is also highly counter-productive. It encourages practitioners to take sides when they should be working together to devise common and pluralistic approaches to a complex set of preservation problems.

Migration purports to concentrate on the information content of the digital object itself and attempts to preserve for future generations the ability to access that content in the face of constantly changing technology. Key to this approach is the very Socratic notion of 'significant properties' that was developed in the CEDARS project and has recently received useful coverage within the InSPECT project (Knight and Pennock 2009) and from the British Library (Dappert and Farquhar 2009). Significant properties are held to refer to the very essence of a digital object; its intellectual content. The argument goes that so long as the significant properties of a digital object are retained, the object's intellectual content will have been preserved in spite of any 'superficial' changes in form or appearance. A property of a digital object that is not held to be significant can, so the reasoning goes, simply be ignored in preservation actions. This does not however represent the ideal state of affairs. As Hedstrom & Lee put it: "In an ideal world, free from technical and economic constraints, libraries and archives would preserve their physical and digital collections in their original form with all significant properties intact." (Hedstrom and Lee 2002). The Planets project also engaged with the idea of significant properties (Montague et al., 2010b).

Migration, as a preservation approach, presupposes that the information content of digital objects is both fixed and discernable, and can survive intact any changes to form and structure which are necessitated by a succession of migration processes. However there is little reason to suppose that any of this is true in general. It is not difficult to provide examples which show that the meaning and significance of a digital (or any other) object changes over time and place. Objects which had little perceived significance at the time of their creation often come to have much greater importance for future generations. For example, the Rosetta Stone which at the time of its production in 196 BCE served to record a decree issued at Memphis on behalf of Ptolemy V, came in the 19th Century to have a different (and much greater) significance in unlocking our understanding of Egyptian hieroglyphics. Not only does the Rosetta Stone serve to show that the significance of objects changes over time, but it also shows that the information content of objects is a derived property rather than something which is intrinsic. The creators of the Rosetta Stone would presumably have had no intention of providing a means by which to understand hieroglyphics. They would, one must suppose, have been much more intent on ensuring that Ptolemy's decree was widely understood. The stone only came to have the importance we attribute to it because knowledge of hieroglyphics was lost from the world and may well have remained lost had not the stone existed. Suppose the creators of the Rosetta Stone had been asked to identify its 'significant properties' and been charged with developing a migration-based preservation strategy for preserving its information content into the future. It is entirely possible that they would have been satisfied to preserve a single version of the text in (say) modern Egyptian so long as it captured reasonably well the sense of Ptolemy's original decree. If the original Stone had not also been preserved, the information content it was originally understood to have contained would, today, be of only passing interest and might perhaps not be worth retaining at all.

Writing in a slightly different context, Dappert & Farquhar come to much the same conclusion; "*... it is not possible to determine out of context which properties reflect content and which reflect circumstance.*" (Dappert and Farquhar 2009).

For digital counterparts of the Rosetta Stone, the situation is made even more complex because preserving an original bitstream in addition to any modified versions will not, of itself, assure future generations access to the original digital object, unless further steps are taken to protect some route back to the original technology platform.

In an interesting and generally stimulating paper (Hedstrom and Lampe 2001) the question of the extent to which users discriminate between or much care about whether the digital objects are preserved using a migration strategy or an emulation approach is explored. Hedstrom and Lampe's methodology was to train test subjects for one hour on a game called Chuckie Egg, originally devised for the BBC Micro. The subjects first played the game on the original hardware platform using an unmodified copy of the game. Next, the participants were divided into two groups the first of which of which played on a migrated version of the game while the other played on an emulator. Participants' responses were gathered under four headings:

satisfaction, perceived ease of use, performance, perceived differences between the original game and the game in the test condition.

Many of the differences that users reported concerned the hardware environment and Hedstrom & Lampe were struck by the sensitivity of users to small changes in the digital object. They concluded that: "Although some of these attributes may be unique to interactive games, this user test suggests that archivists and librarians need a much more refined definition of the characteristics of digital objects that may warrant preservation, regardless of whether emulation or migration is the preferred technical strategy. Users in our study identified attributes such as motion, speed, and sound quality, which are present in many contemporary interactive digital objects but have received scant attention in discussions of digital preservation.'(Hedstrom and Lampe 2001).

In some ways, this experiment provides the most favourable conditions under which to assess migration. In order to draw conclusions about how migration might compare to emulation as a preservation approach over the long term it would have been better to provide the subjects with a migrated digital object that had undergone a succession of migrations, preferably each carried out by a different team. During each migration, one might expect that a digital object will exhibit minor behavioural deviations from the previous migrated version. Over a reasonable number of iterations we might expect this to result in significant and quite noticeable changes. Had the participants in Hedstrom & Lampe's experiment been exposed to a digital object that was repeatedly migrated rather than a 'first generation' migration object, the difference in the user experience offered between accessing preserved digital objects via emulation and via migration might have been more apparent. To illustrate the point, a sentence (in English) was entered into the online translation service Babel Fish[5] the output (in Italian) was fed back into Babel Fish and the process was repeated a number of times with the following results:

1. (English) This might be the result of successive migration interventions.
2. (Italian) Ciò ha potuto essere il risultato degli interventi successivi di espansione.
3. (French) Cela a pu être le résultat des interventions suivantes d'expansion.
4. (Portugeese) Aquilo pôde ser o resultado das intervenções seguintes d' expansão.
5. (English) That could be the result of the following interventions d' expansion.
6. (Korean) 그것은뒤에오는내정간섭 d'의결과일수있었다; 확장.
7. (English) It the domestic intervention d' which comes after; Justice resultant one possibility was; Expansion.

For properties of a digital object that are not treated as significant there is no reason to assume that their 'fidelity' would be any better preserved across successive migrations than was our test sentence. The effects of this little test appear very stark. However, even under ideal conditions, translation from one natural language to another is a complex and 'lossy' activity. There are no strong grounds for believing that all the 'significant' properties of a digital object are capable of being retained across machine language migrations any better than with those of text survive natural

[5] `http://babelfish.yahoo.com/translate_txt`

language migrations. It should also be borne in mind that each migration presents its own challenges and the amount of 'noise' picked up during the move from one hardware platform to another is not a constant. The degree to which a digital object will be damaged during migration will be dependent on the characteristics of the digital object itself and the peculiarities of the platforms concerned.

The potential loss of digital objects and their underpinning hardware presents challenges that traditional archaeology would be hard pressed to resolve as form and function have become separated by layers of abstraction and interrelated complexity. Yet technological obsolescence, collective cultural neglect, lack of familiarity in the cultural conscience coupled with a lack of intuitive affordance of the carrier or hardware exterior and mechanics, means that hardware and software often face the same contextless consignment to obscurity and interpretation that can beset some ancient artefacts and yet the process is occurring in timescales measured in decades rather than centuries.

The awareness of the need for interpretation and intermediated imitation is not new it has merely become more daunting in its scope so that sustaining backward compatible systems on each new generation of hardware is simply neither tenable nor, perhaps more importantly, profitable for developers. Indeed supporting emulation of hardware platforms by commercial companies potentially incurs some losses when marketing new hardware and software and obsolescence can be viewed by some companies as a welcome benefit of systems and software with limited shelf lives as this promotes perpetual renewal of perceived needs in the consumer market.

This in and of itself need not be detrimental in the context of digital preservation after all it is this consumer driven development that has produced competition and subsequent advances in technology which in turn have created the significant impact on all aspects of society that demands preservation but also the ability to envisage plan and technologically implement preserving data on a scale that has never been previously undertaken with other information media. However whilst the capability may exist in theory, the practical implementation and adoption of any given strategy, is beset with problems of phenomenal complexity.

Identifying strategies, technological concerns, stakeholders, investment, obligations legalities and risk are the purpose of this paper but where resources are identified and evaluated they need to be viewed in their wider context and should not in any way be considered stand alone solutions.

Portability of emulation across current and future platforms is obviously a cornerstone in KEEP but in order for this to have current and long term application in digital preservation, a wide array of effective emulators need to be identified classified archived and preserved to ensure that incorporation into an emulation framework is broad enough and robust enough to meet user group needs. Furthermore such a framework and its accessible inventory must ensure adequate, sustainable and adaptable growth as new historical systems are identified, emulated, and added, and new future developments integrated.

Possibly only a virtual system could support such an approach with current levels of understanding but the virtual technology and sourcing emulators are not the only issues. Currently emulation could be viewed as independent yet cooperative, altruistic in motive, yet opportunistic in priorities, and user driven by the desires of enthusiast groups that are disparate in their intentions and often operate in isolated niches.

For long term preservation, utilising an emulation strategy, to operate with any significant breadth and depth and any hope of longevity these groups and resources of similar talents and expertise need to be de-marginalised and inform or contribute directly to a centralised systematic approach to emulating.

Furthermore whilst emulation is sometimes referred to as a strategy for data that gets left behind, interim strategies are variable in effectiveness and in implementation and adherence. To facilitate the recovery of the data left behind standards need to be identified for preservation of carrier mediums and the hardware to read and interpret them. It is likely that as things are progressing data that gets left behind will become more commonplace unless awareness and education of the issues are disseminated widely and furnished with a low burden solution for developers and commercial users.

Currently our cultural approach to the needs and scale of the issues involved in digital preservation have been slow in realisation and much of society remains unaware of the problems and risks that are being faced. Some elements are responding quicker than others but without a clearly laid out unified approach with appropriate sharing of the burden, costs and responsibilities the results will inevitably leave significant digital gaps as a legacy not just for future generations but also for the future of our current generation.

With regard to digital information and objects, memory institutions have established a number of clear guidelines with regard to the selection criteria for digital information and digital objects themselves and digital preservation standards are clear and well established e.g. OAIS etc.

Curation and Archiving is established practice and within such institutions clear policy guidelines and standards are available with a wide range of supporting resources including systematic approaches to migration and many are implemented with good systematic approach to adherence and review at significant institutions. However implementation and adherence is not universal to other appropriate organisations and companies and tackling this potential chasm requires a technological safety net and a creative approach to raising awareness and supporting those companies in engaging in preservation without creating a load that cannot be maintained. In the interim period such information may at best be consigned to abandoned media or permanently and irretrievably lost. Nevertheless standards in this discipline are well reasoned, clearer, better structured and more widely achievable than most but are dependant on other disciplines keeping pace in order for their data to remain accessible.

Hardware preservation on the other hand, with some exceptions, lacks clear guidelines or coherent aims that can be employed as a universal approach. Researching to find policies and standards for the preservation strategies for computing hardware and peripherals should provide an extensive and specific understanding of the different technologies and all the risks prevalent to the collective elements of any given hardware. However enquiries and research yielded no apparent identifiable policy guidelines or standards on preservation (other than for shipment or short term working use). Furthermore individual sources of advice are often generic and sometimes conflicting on critical points even to the extent as to whether machines should be retained. Machine preservation and maintenance approaches often are tackled ad hoc, and although regulated standards might not provide a definitive answer, guidelines and best practice, informed by research and experience, would surely be beneficial.

Working within the context of the KEEP project, the University of Portsmouth Future Proof Computing Group have started identifying emulators that are currently available or in development (a small selection of which are presented in appendix A to this paper). Naturally such a list reflects the diverse intentions of the authors and outcomes and information can be lacking, inconsistent and at face value defy simple categorisation.

Although other projects have attempted to identify emulator sources as part of their approach to digital preservation (Camileon, Creative Archiving at Michigan and Leeds) the reliance on disparate individuals whose personal endeavours are prone to the vagaries of individual circumstances and the fortunes of the host sites through which they are accessed, leave such projects vulnerable to unpredictable loss. As such the emulator landscape can shift significantly between projects with many emulator sites, including those indexed with a range of emulators, being lost, often with little or no warning, in the last few years (emulation9, emula zone, emulation.cc, system16 etc.) and ironically even the findings and reports of such projects as Camileon are also often no longer available online on their sites.

In order to make a sustainable strategy for expanding the range of emulators available to institutions and archivists for a coherent digital preservation plan there needs to be a more systematic approach to establishing an inventory resource of emulators themselves. In addition there needs to be a coordinated approach by institutions and archives to identify emulation needs i.e. systems that have a high risk and high impact that have yet to be emulated, and standards required of an emulator, that will allow data to be effectively presented. This will then allow institutions to establish targeted working groups with users, developers and emulator enthusiasts to resolve those emulation needs.

Whilst some projects have attempted to tap into the resources of enthusiasts and emulator groups there is as yet no established, mainstream, institutionally backed, project, to our knowledge, that has attempted a systematic production of key emulator types within a strategy for long term preservation, with identified outcomes and standards. While emulation has a significant role to play in digital preservation, the lack of standardisation criteria leaves our digital legacy in the hands of niche interest groups that are not adequately supported or endorsed by the institutions that need them, and lack legitimacy and recognition in legal terms.

There is a real need to foster closer association with emulation authors, industry and institutions to provide legitimacy and mutual trust within an established protocol which would enable digital preservation approaches such as the KEEP emulation framework to meet the varied needs of diverse research, curatorial and interest groups by incorporating additional emulation developments as they arise into a system that will continue to port the emulation inventory to new hardware systems.

Locating emulator sources however is not the whole solution as indicated by an extract from the NISO document: Initiatives Principle 4: A good digital initiative has an evaluation component.

Of course the veracity of any such system requires the incorporated emulators to meet a defined standard and the only viable means of establishing and checking this is to compare it to known outcomes from the original machines ideally with confirmation from a user or developer that the performance is a faithful representation of the common experience. Given the nature of rapid obsolescence in the computer industry, such quality checking, needs to be established with some imperative, whilst the limited number of extant, original, machines can be sourced before extinction.

To this end some resources remain available to us, with a number of established museums and some private collections that provide a means to inform emulator development as well as an opportunity to test the behaviour of emulators against the performance of the machines they purport to emulate. In addition they provide an invaluable safeguard by maintaining such machines so that vital data may be extracted and accessed. However such museums are not ideally equipped to work in an appropriately large scale without additional support.

Often such museums are staffed by volunteers and often comprise retired experts and developers along with enthusiasts and key figures in computing history and it is often they, as much as their exhibits, that bring history alive with their unique insights, profound knowledge and personal accounts. Furthermore their knowledge is often extensive and detailed with practical experience of the working machine, its idiosyncrasies and common practice that are not always readily elicited from the literature. These individuals are experts in a very real sense whose knowledge and accounts are also rare and sadly in a pragmatic sense potentially vulnerable to irreplaceable loss. There is a very real need here to make a record of that applied expertise while the opportunity of the dialogue between the individual expert and the operational machines is still available to us. Our reliance on these individual enthusiasts and private groups to maintain our heritage somehow seems to miss the point that they themselves are part of our heritage and their legacy is not yet secure. As part of the effort to see that this is done, the UK National Museum of Computing (TNMOC) at Bletchley Park, working with the University of Portsmouth have carried out a series of interviews with TNMOC staff, volunteers, enthusiasts, experts, historians, engineers restorers, developers preservationists and retired industry users amongst others. The outputs from these events will be made available in due course through the University of Portsmouth website.

References

Bearman, D.: Collecting Software: A New Challenge for Archives & Museums. Archival Informatics Technical Reports 1, 80 (1987)

Bearman, D.: Reality and Chimeras in the Preservation of Electronic Records. D-Lib Magazine 5(4) (1999)

Dappert, A., Farquhar, A.: Significance Is in the Eye of the Stakeholder. In: Agosti, M., Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonas, G. (eds.) ECDL 2009. LNCS, vol. 5714, pp. 297–308. Springer, Heidelberg (2009)

Gladney, H.M.: Durable Digital Objects Rather Than Digital Preservation. ErpaePrints (2008)

Hedstrom, M., Lampe, C.: Emulation vs. Migration: Do Users Care? RLG Diginews 5 (2001)

Hedstrom, M., Lee, C.A.: Significant properties of digital objects: definitions, applications, implications. DLM-Forum, Barcelona (2002)

Knight, G., Pennock, M.: Data Without Meaning: Establishing the Significant Properties of Digital Research. International Journal of Digital Curation 4(1), 159–174 (2009)

Levy, D.M.: Where's Waldo? Reflections on Copies and Authenticity in a Digital Environment. Authenticity in a Digital Environment, pp. 24–31. Council on Library and Information Resources, Washington D.C (2000)

Rothenberg, J.: Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation. Council on Library and Information Resources (1999)

Stawowczyk Long, A.: Long-term preservation of web archives – experimenting with emulation and migration methodologies. Internation Internet Preservation Consortium 54 (2009)

Swade, D.: Preserving Software in an Object-Centred Culture. In: Higgs, E. (ed.) History and Electronic Artefacts, pp. 195–206. Clarendon Press, Oxford (1998)

Appendix A: Computer Emulators/ Multiple Emulators/ Atypical Emulation

Computer Emulators list [1] (updated March 2012). This list is not comprehensive. Please note that this list does not cover specific emulated games nor games consoles except where covered under multiple emulation systems or under the auspices of arcade emulations.

Neko Project II	http://www.emulator-zone.com/doc.php/computer/kekoproject2.html	Windows	Freeware
CPCE	http://www.emulator-zone.com/doc.php/computer/cpce.html	Windows/Dos	Freeware
CCS64	http://www.emulator-zone.com/doc.php/computer/ccs64.html	Windows	Freeware
Hoxs64	http://www.emulator-zone.com/doc.php/computer/hoxs64.html	Windows	Freeware
Mini vMac	http://www.emulator-zone.com/doc.php/computer/minivmac.html	Many	Free
PearPC	http://www.emulator-zone.com/doc.php/computer/pearpc.html	Windows	Freeware
YAPE	http://www.emulator-zone.com/doc.php/computer/yape.html	Windows	Freeware

Intelivision Console Emulator

Nostalgia	http://www.emulator-zone.com/doc.php/misc/nostalgia.html	Windows	Freeware
WinArcadia	http://www.emulator-zone.com/doc.php/misc/winarcadia.htm	Windows	Freeware
RetroCopy	http://www.emulator-zone.com/doc.php/misc/retrocopy.html	Windows/Linux	Freeware
Turbo Engine	http://www.emulator-zone.com/doc.php/misc/engine.html	Windows/Linux	Freeware

Arcade Emulators: (http://www.emulator-zone.com/doc.php/arcade/)

MAME	http://www.emulator-zone.com/doc.php/arcade/mame.html	All(?)	Freeware
Kawaks	http://www.emulator-zone.com/doc.php/arcade/kawaks.html	Windows	Freeware
FinalBurn Alpha	http://www.emulator-zone.com/doc.php/arcade/finalburnalpha.html	Windows	Freeware
Zinc	http://www.emulator-zone.com/doc.php/arcade/zinc.html	Windows	Freeware

Nebula	http://www.emulator-zone.com/doc.php/arcade/nebula.html	Windows	Freeware
Calice	http://www.emulator-zone.com/doc.php/arcade/calice.html	Windows	Freeware
VivaNonno	http://www.emulator-zone.com/doc.php/arcade/vivanonno.html	Windows	Freeware
Daphne	http://www.emulator-zone.com/doc.php/arcade/daphne.html	Many	Free
Raine	http://www.emulator-zone.com/doc.php/arcade/raine.html	Windows	Freeware
RetroCopy	http://www.emulator-zone.com/doc.php/misc/retrocopy.html	Windows/Linux	Free

MAME (Multiple Arcade Machine Emulator)

Official MAME Emulators

File	Link	Platform	License
MAME (regular) 0.145 Win32 command line version	http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b.exe	Windows	Freeware
MAME (i686) 0.145 Win32 command line version (I686 optimized)	http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b_i686.exe	Windows	Freeware
MAME (64 bit) 0.145 64-bit Windows command-line binaries	http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/winmame/mame0145b_64bit.exe	Windows	Freeware
MAME 0.100 DOS command line version	http://www.emulator-zone.com/download.php/emulators/arcade/mame/official_version/dosmame/mame0100b_dos.zip	Windows	Freeware

MAME Ports

File		Platform	License
MAMEUI (32bit) 0.145 32 bit version	http://www.emulator-zone.com/download.php/emulators/arcade/mame/mameui/MameUI32_0.145.7z	Windows	Freeware
MAMEUI (64bit) 0.145 64 bit version	http://www.emulator-zone.com/download.php/emulators/arcade/mame/mameui/MameUI64_0.145.7z" \t	Windows	Freeware
MAMEUI FX 32 0.145 MAMEUI	http://www.emulator-zone.com/download.php/emulators/arcade/mame/MAMEUIFX32/mameuifx32_0145.exe	Windows	Freeware

MAME Frontends

File		Platform	License
Emuloader 5.8.2 Command line MAME frontend	http://www.emulator-zone.com/download.php/emulators/arcade/mame/frontend/emuloader/el582-bin.rar	Windows	Freeware
MAME Classic 5.5.0 Command line MAME frontend	http://www.emulator-zone.com/download.php/emulators/arcade/mame/frontend/mame_classic/FI550.zip	Windows	Freeware

MAME Support Files

File		Platform	License
mameinfo.dat 0.136 Information DAT file	http://www.emulator-zone.com/download.php/emulators/arcade/mame/support_files/mameinfo-dat/Mameinfo0136.zip	N/A	Freeware
Cheat.dat 0.132 Cheat codes DAT file	http://www.emulator-zone.com/download.php/emulators/arcade/mame/support_files/cheat-dat/cheat0132.zip	N/A	Freeware
History.dat 0.136 History DAT file	http://www.emulator-zone.com/download.php/emulators/arcade/mame/support_files/history-dat/mamehistory136.zip	Windows	Freeware
Highscore.dat 0.132 High score DAT file	http://www.emulator-zone.com/download.php/emulators/arcade/mame/support_files/highscore-dat/mame0132.zip	Windows	Freeware

Multiple Systems Emulators (M.E.S.S.)

File		Platform	License
M.E.S.S.	http://www.emulator-zone.com/doc.php/misc/mess.html	Windows	Freeware
M.E.S.S. (32bit) 0.145	http://www.emulator-zone.com/download.php/emulators/misc/mess/mess0145b.zip	Windows	Freeware
M.E.S.S. (64bit) 0.145	http://www.emulator-zone.com/download.php/emulators/misc/mess/mess0145b_x64.zip	Windows	Freeware
MESS 0.101b Multi Emulator Super System	http://www.emulator-zone.com/download.php/emulators/mac/mess/macmess0101b.zip	Macintosh	Freeware

Multiple Emulator Super System Claims to Emulate the Following:

- Adventurevision
- Amstrad 464plus
- Amstrad 6128plus
- Amstrad PC1512 (version 1)
- Amstrad PC1512 (version 2)
- Amstrad PC1640 / PC6400 (US)
- Amstrad PC20
- APF M-1000
- Apple][
- Apple][+
- Apple //c
- Apple //c Plus
- Apple //c (UniDisk 3.5)
- Apple //e
- Apple //e (enhanced)

- Amstrad/Schneider CPC464
- Amstrad/Schneider CPC6128
- Amstrad/Schneider CPC664
- APEXC (as described in 1957)
- APF Imagination Machine
- Atom with Eprom Box
- Bally Pro Arcade/Astrocade
- Bally Pro Arcade/Astrocade (white case)
- BBC Micro Model A
- BBC Micro Model B
- BBC Micro Model B+ 128k
- BBC Micro Model B+ 64K
- BBC Micro Model B with WD1770 disc controller
- C64GS (PAL)
- C65 / C64DX (Prototype, German PAL, 910429)
- C65 / C64DX (Prototype, NTSC, 910111)
- C65 / C64DX (Prototype, NTSC, 910523)
- C65 / C64DX (Prototype, NTSC, 910626)
- C65 / C64DX (Prototype, NTSC, 910828)
- C65 / C64DX (Prototype, NTSC, 911001)
- CBM4064/PET64/Educator64 (NTSC)
- Channel F
- Chess Champion MK II
- Colecovision
- Colecovision (Thick Characters)
- Color Computer
- Color Computer 2
- Color Computer 2B
- Color Computer 3 (NTSC)
- Color Computer 3 (NTSC; HD6309)
- Color Computer 3 (PAL)
- Color Computer (Extended BASIC 1.0)
- Colour Genie EG2000
- Commodore 128 French (PAL)
- Commodore 128 German (PAL)
- Commodore 128 Italian (PAL)
- Commodore 128 NTSC
- Commodore 128 Swedish (PAL)
- Commodore 16/116/232/264 (PAL)
- Commodore 16/116/232/264 (PAL), 1551
- Commodore 16 Novotrade (PAL, Hungarian Character Set)
- Commodore 30xx (Basic 2)
- Commodore 30xx (Basic 2) (business keyboard)
- Commodore 364 (Prototype)
- Commodore 40xx FAT (CRTC) 50Hz
- Commodore 40xx FAT (CRTC) 60Hz
- Commodore 40xx THIN (business keyboard)
- Commodore +4 (NTSC)
- Commodore +4 (NTSC), 1551
- Commodore 64 (NTSC)
- Commodore 64 Swedish (PAL)
- Commodore 64/VC64/VIC64 (PAL)
- Commodore 80xx 50Hz
- Commodore 80xx 60Hz
- Commodore 80xx German (50Hz)
- Commodore 80xx Swedish (50Hz)
- Commodore B128-40/Pet-II/P500 60Hz
- Commodore B128-80HP/710
- Commodore B128-80LP/610 60Hz
- Commodore B256-80HP/720
- Commodore B256-80HP/720 Swedish/Finnish
- Commodore B256-80LP/620 50Hz
- Commodore B256-80LP/620 Hungarian 50Hz
- Apple //e (Platinum)
- Apple I
- Aquarius
- Arcadia 2001
- Atom
- Dragon 64
- Enterprise 128
- Enterprise 128 (EXOS 2.1)
- EURO PC
- Famicom
- Galaksija
- Gamboy - (PAL) Japanese SMS BIOS v2.1
- GameBoy
- GameBoy Color
- GameBoy Pocket
- Game Gear - European/American
- Game Gear - European/American Majesco Game Gear BIOS
- Game Gear - Japanese
- Game Gear - Japanese Majesco Game Gear BIOS
- Geneve 9640
- HB-8000 Hotbit 1.1
- HB-8000 Hotbit 1.2
- IBM PC 08/16/82
- IBM PC 10/27/82
- IBM PC/XT (CGA)
- Intellivision
- Intellivision Keyboard Component (Unreleased)
- Intellivision (Sears)
- Inves Spectrum 48K+
- Jupiter Ace
- Kaypro 2x
- KC 85/3
- KC 85/4
- KC Compact
- KIM-1
- Laser 110
- Laser 200
- Laser 210
- Laser 310
- Laser 350
- Laser 500
- Laser 700
- Lisa2
- Lisa2/10
- LNW-80
- Lynx
- Lynx (alternate rom save!)
- Lynx II
- Macintosh 512ke
- Macintosh Plus
- Macintosh XL
- Mark III - (PAL) Japanese SMS BIOS v2.1
- Master System III Compact (Brazil) - (PAL) European BIOS with Sonic The Hedgehog
- Master System II - (NTSC) US/European BIOS with Alex Kidd in Miracle World
- Master System II - (PAL) European BIOS with Sonic The Hedgehog
- Master System II - (PAL) US/European BIOS with Alex Kidd in Miracle World
- Master System - (NTSC)
- Master System - (NTSC) Hacked US/European BIOS v1.3

- Commodore Max (Ultimax/VC10)
- Commodore SP9000/MMF9000 (50Hz)
- CP400
- CPS Changer (Street Fighter ZERO)
- Dragon 32
- Master System - (PAL) Hacked US/European BIOS v1.3
- Master System - (PAL) Japanese SMS BIOS v2.1
- Master System - (PAL) US/European BIOS v1.3
- Master System - (PAL) US/European BIOS v3.4 with Hang On
- Master System Plus - (NTSC) US/European BIOS v2.4 with Hang On and Safari Hunt
- Master System Plus - (PAL) US/European BIOS v2.4 with Hang On and Safari Hunt
- MC-10
- Megadrive / Genesis
- Microbee 32 IC
- Microbee 32 PC
- Microbee 32 PC85
- Microbee 56
- Microtan 65
- MSX 1
- MSX 1 (Japan)
- MSX 1 (Korea)
- MSX 1 (UK)
- MSX 2
- MSX 2 (BASIC 2.1)
- MSX 2 (Japan)
- MTX 512
- MZ-700
- MZ-700 (Japan)
- Nascom 1 (NasBug T1)
- Nascom 1 (NasBug T2)
- Nascom 1 (NasBug T4)
- Nascom 2 (NasSys 1)
- Nascom 2 (NasSys 3)
- Nintendo Entertainment System (NTSC)
- Nintendo Entertainment System (PAL)
- Odyssey 2
- Oric 1
- Oric Atmos
- Oric Telestrat
- PC200 Professional Series
- PC-8801 MKIISR (Hires display, VSYNC 24KHz)
- PC-8801 MKIISR (Lores display, VSYNC 15KHz)
- PC/AT (CGA, MF2 Keyboard)
- PC/AT (VGA, MF2 Keyboard)
- PC (CGA)
- PC Engine/TurboGrafx 16
- PC (MDA)
- PC/XT (VGA, MF2 Keyboard)
- PDP-1
- PET2001/CBM20xx Series (Basic 1)
- Philips P2000M
- Philips P2000T
- PK-01 Lviv
- PK-01 Lviv (alternate)
- PK-01 Lviv (prototype)
- Pocket Computer 1251
- Pocket Computer 1350
- Pocket Computer 1401
- Pocket Computer 1402
- Pocket Computer 1403
- Master System - (NTSC) US/European BIOS v1.3
- Master System - (NTSC) US/European BIOS v3.4 with Hang On
- Master System - (PAL)
- Sam Coupe
- Sanyo / Dick Smith VZ200
- Sanyo / Dick Smith VZ300
- Sorcerer
- Sord M5
- Spectrum I+
- Superboard II
- Super GameBoy
- Super Nintendo Entertainment System (NTSC)
- Super Nintendo Entertainment System (PAL)
- Super Vision
- SVI-318
- SVI-328
- SVI-328 (BASIC 1.11)
- System-80
- Tandy 1000HX
- Tatung Einstein TC-01
- TC-2048
- Texet TX8000
- TI-81 Ver. 1.8
- TI-85 ver. 10.0
- TI-85 ver. 3.0a
- TI-85 ver. 4.0
- TI-85 ver. 5.0
- TI-85 ver. 6.0
- TI-85 ver. 8.0
- TI-85 ver. 9.0
- TI-86 homebrew rom by Daniel Foesch
- TI-86 ver. 1.2
- TI-86 ver. 1.3
- TI-86 ver. 1.4
- TI-86 ver. 1.6
- TI99/4A Home Computer (Europe)
- TI99/4A Home Computer (US)
- TI99/4A Home Computer with EVPC
- TI99/4 Home Computer (Europe)
- TI99/4 Home Computer (US)
- TI Avigo 100 PDA
- TI Model 990/10 Minicomputer System
- TK-90x Color Computer
- TK-95 Color Computer
- TRS-80 Model I (Level I Basic)
- TRS-80 Model I (Radio Shack Level II Basic)
- TRS-80 Model I (R/S L2 Basic)
- TS-2068
- UK101
- UK-2086 ver. 1.2
- Vectrex
- VIC1001 (NTSC)
- VIC20 (NTSC)
- VIC20 (NTSC), IEEE488 Interface (SYS45065)
- VIC20 PAL, Swedish Expansion Kit
- VIC20/VC20(German) PAL
- XP-800 Expert 1.0
- XP-800 Expert 1.1
- Z88
- ZX Spectrum
- ZX Spectrum 128
- ZX Spectrum 128 (Spain)
- ZX Spectrum +2

- Pocket Computer 1403H
- Pravetz 8D
- Pravetz 8D (Disk ROM)
- Pravetz 8D (Disk ROM, RadoSoft)
- Salora Fellow
- ZX Spectrum +3
- ZX Spectrum +3e
- ZX Spectrum +3e (Spain)
- ZX Spectrum +3 (Spain)
- ZX Spectrum +4
- ZX Spectrum +2a
- ZX Spectrum +2 (France)
- ZX Spectrum +2 (Spain)
- ZX Spectrum (BusySoft Upgrade v1.18)
- ZX Spectrum (Collier's Upgrade)
- ZX Spectrum (De Groot's Upgrade)
- ZX Spectrum (LEC Upgrade)
- ZX Spectrum (Maly's Psycho Upgrade)

Multiple Systems Emulators (Kawaks)

File		Platform	License
Kawaks 1.62	http://www.emulator-zone.com/download.php/emulators/arcade/kawaks/winkawaks162.zip	Windows	Freeware

Multiple Systems Emulators (Nebula)

File		Platform	License
Nebula 2.25b CPS2, Neogeo, PGM and Konami emulator	http://www.emulator-zone.com/download.php/emulators/arcade/nebula/nebula225b.zip	Windows	Freeware

File		Platform	License
ScummVM	http://www.emulator-zone.com/doc.php/computer/scummvm.html	Many	Open Source
ScummVM (32bit) 1.1.1	http://www.emulator-zone.com/download.php/emulators/computer/ScummVM/scummvm-1.1.1-win32.zip	Windows	Freeware
ScummVM (64bit) 1.1.1	http://www.emulator-zone.com/download.php/emulators/computer/ScummVM/scummvm-1.1.1-win64.zip	Windows	Freeware

My Fascination with Computing History

John Impagliazzo

John.Impagliazzo@hofstra.edu

Abstract. This narrative on computing history reflects the experiences of the author and his involvement with computing history over a quarter century. The discussion portrays a transition from loathing history as a student to embracing computing history as a professional. The author shows how storytelling can produce interesting excursions on technical subjects and ways in which teaching computing with history can elevate student interest. He also provides examples showing ways in which historical events could complement computing studies. The article also explains how the author's earlier efforts in using history to teach computing led to a landmark publication and subsequent activities within IFIP leading to conferences and related publications.

Keywords: Computing history, computing education, history and computing, human factors in computing.

1 Background

In the course of one's life, certain events transcend their primal characterizations. As a student, I had always found history uninteresting. For me, the subject was one of memorization of certain facts such as persons, dates, and places. The usual basis for class and course examinations was memorization and little critical thinking was ever involved. That pattern continued to the point where I would only study history when required to do so. With a few exceptions in my circle of student colleagues, almost everyone knew I disliked history.

As years passed, I began to kindle a new liking for the subject. Pioneers in mathematics, engineering, physics, and computing began to fascinate me. People such as Galileo, Newton, Euler, and von Neumann began to influence my thinking with the realization that contributions to their subjects were more than an equation or a process. These were people, humans, who had advanced their subject areas to new heights. Should I, we, not know more about them?

The juncture of this epiphany occurred in the 1980s. I had become a close friend of the famous mathematician, Marshall H. Stone. I had begun to write his mathematical biography and in the many meetings I had with Marshall, he would always bring up "Johnny", referring to eminent mathematician and physicist John von Neumann. Although von Neumann died in 1957, a worldwide celebration of his achievements was lacking. I can still hear Marshall say, "We must do something for Johnny".

As it happened, I eventually organized the conference called the "Legacy of John von Neumann" in May of 1988 at Hofstra University. Famous scientists, engineers, mathematicians, and Nobel laureates who knew of or had an affinity to von Neumann

A. Tatnall (Ed.): Reflections on the History of Computing, IFIP AICT 387, pp. 385–395, 2012.

and his works spoke at the conference. Even von Neumann's family members and some of his students participated in the event. For me, this gathering of scientific dignitaries became the turning point of my new fascination with history and in particular, computing history. The proceedings from this event [2], which I co-edited, still receive international acclaim.

In the late 1980s, I began incorporating elements of history in the computing courses I taught. Almost every class included a morsel or story on some historical nuance related to the topic at hand. If we were discussing the von Neumann bottleneck, I would ask, "Who was John von Neumann"? If we were discussing a finite automaton, I would ask, "Who developed graph theory"? Of course, few if any students knew the answers to these questions. They would then do a lookup somehow (remember, web browsers did not exist then), and the next day I would have a brief discussion surrounding the findings from the students. The most embarrassing question for students was,

> *Who invented the computer?*

After dismissing responses such as Bill Gates or Steve Jobs, the discussions often became quite interesting – even heated or bizarre. I would berate the class by saying something like,

> *What a disgrace! Computer science majors and they don't even know who invented the computer.*

with the hope of garnering their greater attention and perk their curiosity.

In the mid-1990s, ACM designated me to chair the 1996 ACM Special Interest Group on Computer Science Education (SIGCSE) conference in Philadelphia, Pennsylvania. This 1996 ACM gathering included more than the SIGCSE event. It also included the beginning of the fiftieth anniversary celebration of ACM, the fiftieth anniversary of the ENIAC computer with its ceremonial re-launching at the University of Pennsylvania, the last ACM Computer Science Conference, and the chess match between the IBM Deep Blue computer and the world chess master and champion Garry Kasparov. The year before, John A.N. Lee from Virginia Tech published his book on computing pioneers [11]; I invited JAN to be one of the keynote speakers at the SIGCSE event. JAN was the former editor-in-chief of the *IEEE Annals of the History of Computing*; he was also the chair of Working Group 9.7 (WG 9.7) on the History of Computing of the International Federation for Information Processing (IFIP) [3]. JAN and I soon became friends, a friendship that furthered my growing interest in computing history.

2 Storytelling

As already mentioned, I had been using history in computing classes since the late 1980s; storytelling was one of my approaches. The study of computing hardware and software could be a "dry" experience for students; teachers usually present technical material in sterile ways. By using the technique of storytelling, I provided background to the theories of hardware and software topics. In this manner, students would gain a better understanding of the topic under development and learn theoretical concepts in the context of social, political, or economic conditions. I have found that students often gain a better understanding of the subject when they learn how diverse events and ideas could fuse together to stimulate discussion and inventive ideas on the evolution of new and modern devices and machines. Experience shows that when I involve history and storytelling in computing courses, they make the learning experience much more interesting and meaningful.

Introductory courses, for example, often take an "overview" approach. This allows flexibility in the way a teacher could present topics. Students attending introductory courses often come from a variety of specialties; this heterogeneous mix of students creates a perfect environment to use historical morsels to elevate student interest; it also allows instructors to explore and use inventive measures to retain student attention. Computing history is one way to merge technical and non-technical topics to foster a constructive learning experience.

3 Teaching Computing with History

George Santayana (1863-1952) had alerted us on the importance of history. One variation of his saying is, "Those who cannot learn from the past are condemned to repeat it" [12]. As teachers, we should not ignore the past and the lessons learned for a more promising future. We could use history to resurrect ideas and concepts that either progress had overtaken or were ahead of their time. Without computing history, the core of computing topics becomes only inanimate objects of hardware and software without human connection. It is a teacher's duty to integrate the technical with the non-technical and to connect events to form a tapestry of knowledge for students. Hence, the use of history in computing helps students and professionals make sound technical and business decisions based on their knowledge and experience of the past. Therefore, we see that learning computing topics in context can greatly enhance the way we learn technical aspects of the subject. Learning topics in isolation could be meaningless.

One of the downfalls of the computing profession is the poor way it promotes its own heroes. Unlike other areas such as mathematics, engineering, and science, teachers of computing tend to shy away from hailing the achievements of others. As a result, students hardly hear of the heroes from their profession, perhaps with the exception of Bill Gates and Steve Jobs. How sad. Our computing field was the invention of pioneers. We should not forget them or deny them their due credit. They have paved the way for this relatively new and emerging field. Indeed, they are our heroes.

Students entering their first year of college in the fall of 2012 were likely born in 1994, shortly after the invention of web browsers that we take for granted today. Do they know the story of the trailblazers of this use of the internet? Do they know how this computing network has reached the level of achievement we inherited and use today? I doubt it. Hence, exploring the achievements and challenges of our internet heroes would be one way to expand students' knowledge and engage them in a subject having both technical and non-technical ramifications. An exposure to computing history in this regard would reveal many nuances that could broaden students' thinking and they would likely become more inquisitive and perhaps even more inventive.

Students need history because all mature disciplines transmit their origins and stories to new advocates. It is an acculturation process and teachers usually find it difficult to visualize items of the present time. So, how can we understand them? Older computing systems were simpler and easier to understand. Hence, students need to understand that change is the only constant in the computing field. That is, all elements of computing are evolving and improving. Indeed, students deserve to know the truth regarding the evolution of the subject they are learning.

4 Examples of Using History

When people used the word ‘computer’ in the early days, they generally meant a mechanical, electro-mechanical, or an electronic device to do calculations. In fact, until the mid-1940s, a computer was actually a *person* who did calculations. A more formal and more modern definition of a computer is an electronic calculating device that contains a memory component. Examples of today’s computers include desktop and personal computers, mobile phones, smartphones, tablet devices, and a myriad of other electronic devices used to manipulate and store data as well as communicating information.

Let us consider a few cases to show how one could use history to enhance computing topics. Here, we focus on topics that one might teach and show how the inclusion of individuals and events can enrich the topic at hand. We also note that we should not classify any of the machines mentioned in this section as computers in the modern sense. The reason is that the modern definition of a computer involves an active memory component that can store operating instructions and data. These machines are void of that component.

4.1 Early Mechanical Computers

Perhaps the earliest (about 3000 BCE) mechanical device for doing arithmetic calculations is the abacus, a device still in use today in some parts of the world. More sophisticated though less practical devices for doing simple arithmetic are the gear-driven adding machine called the “Pascalene” of Blaise Pascal (1642) and a calculator using a stepped cylinder gear called the “Stepped Reckoner” of Gottfried Leibniz (1674). Although these mechanical calculators do not fall under the category of “computer”, students could explore how they provided some of the ideas and building blocks of the computing machines of today.

In 1822, Charles Babbage began to design and build the Difference Engine, a mechanical machine intended to calculate tables. Babbage never completed the gear-driven machine, but it was a precursor to his follow-up invention of the Analytical Engine in 1832. Although never completed, this mechanical engine became the basis for engineering modern computing machines. For this reason, some historians consider Babbage to be one of the inventors of the computer. Allowing students to explore this acclamation provides a sound entrée to other interesting overtones related to Babbage and computers.

One hundred years had lapsed without any real progress in developing computing machines. Enter Konrad Zuse. A quick search of the person would reveal that he indeed produced original computers in the late 1930s and early 1940s. Some historians consider Zuse as one of the inventors of a computer. His machines were electro-mechanical devices. A topic such as this would be very appropriate in the early studies of computing. Zuse's machines (called Z1, Z2, and Z3 in their early developments) used physical relays (devices that close and open) to do binary switching. Zuse also invented the first high-level programming language, Plankalkül, to play games on the machines he invented. I wonder how many students or teachers know this fact. A good way to engage students is to have them research the works of Zuse and report their findings.

John Atanasoff designed the ABC computer in 1937 and built it with his graduate student Clifford Berry at Iowa State University. The ABC was an all-electronic (no mechanical part) machine. This special-purpose computer was able do only one thing—it solved systems linear equations. The computer became operational in 1942. The machine implemented the use of binary digits, was totally electronic, was able to systemize memory and computation (similar to a primitive operating system), and used regenerative capacitor memory, items found in every modern computer. Many historians regard Atanasoff as the (forgotten) inventor of the computer. Students and teachers who research the work of Atanasoff will find many interesting revelations.

At the Moore School of the University of Pennsylvania in the early 1940s, John Mauchly and his graduate student and engineer Presper Eckert were building a machine to do military and weather calculations. Called the Electronic Numerical Integrator and Computer (ENIAC) and funded by the federal government, historians consider this machine as the first general-purpose computer because one could program it by changing its hardware circuit boards. The ENIAC was a colossal machine; it contained over 18,000 vacuum tubes (valves) and it had a high probability for system failure. Students who research the story of this machine and the people involved with it should find the investigation quite informative.

One should note that the ABC and the ENIAC machines did not exist without controversy. Eckert and Mauchly patented the ENIAC in the late 1940s. However, many believed that the patent was not valid because the patent office did not consider the "prior art" requirement for patents, especially since Mauchly visited Atanasoff in 1941 and learned about the features of the ABC computer, some of which were incorporated in the ENIAC. After a long legal battle, an adjudication revealed that Eckert and Mauchly did not themselves first invent the ENIAC; instead they derived the subject matter from Atanasoff. Students should be fascinated from the ramifications surrounding these two machines.

4.2 Universal Computers

It is worth mentioning that not all computers need to be physical machines. Since 2012-2013 represents the centennial of the birth of Alan Turing, let us discuss briefly the idea of his universal computer. A universal machine (computer) is one that could calculate any computable function. We will skip the details of this, only to mention that Turing proved that every computable function is solvable by such a machine, which we call a Turing Machine. This topic and all its ramifications appear in courses on computability, theory of computing, and other related subjects. So, how does one enrich this topic using history?

Some historians consider Alan Turing as one of the original inventors of the computer. An interesting exercise here is to have students explore the veracity of this statement and have them investigate other inventors as mentioned above. A study of the 1937 universal machine would reveal that this machine existed only in theory; that is, it was not physical entity. The immediate debate that would occur would involve the challenge of whether a non-physical machine would qualify as a "computer", which could form the basis for a philosophical dialogue and debate among students. No one has actually built this theoretical machine; however, many simulations of it exist. In fact, in my classes I would often ask students to build a simulator for such a machine as a group project or as an individual exercise. The opportunity to program an original computer stimulates great interest among students, given that they have at their disposal modern languages and tools to simulate a working universal machine. Indeed, one could hold a class contest to see which student or group makes the best simulator.

The development of universal simulators is of great interest to students. The simulators allow them to examine this machine and modify its parameters. It helps to show that Turing's claim for universality is important to the understanding of the foundations of the computing field. It also provides the occasion to discuss the life story of Alan Turing and his other contributions to the field of computing such as being a pioneer in computational biology, his code breaking activities at Bletchley Park during World War II, and his pioneering work in machine learning. The study of theoretical computing could be a dull topic for many students. However, with the infusion of computing history, this topic could easily burst into new episodes to enrich the mind and elevate the spirit of most students.

4.3 Computer Memory and Pipelining

Computer memory components can take various forms. Electronic single memory components came into existence in 1947 when F.C. (Freddie) Williams and Tom Kilburn developed a cathode-ray tube memory device and in 1948 when Andrew D. Booth developed the magnetic memory drum. These single memory devices soon reverted to electronic circuit storage devices in the 1950s. Teachers of computing could use these facts as a launch point for further discussion on the development of computer memory and the concept of hierarchical memory. For example, what was the significance of the Atlas computer at Manchester University? What did Maurice Wilkes propose in a 1965 short paper on caching? How did IBM commercially implement the first cache machine in the IBM 360/85 machine? All of this has led to great efficiency advances with memory hierarchies and virtual memory schemes considered common in modern day machines. Asking students to explore and research these inventions engages them in their studies and expands their understanding of the topics they are learning.

Another efficiency scheme is pipelining, an architectural scheme that enables a computer to do multiple activities within a given timeframe. The Control Data CDC 6600 from 1964, often considered "the first supercomputer", was the first commercial machine that used pipeline methodologies and dynamic scheduling. This strategy evolved to the use of multiple function units, score boarding, and new algorithms such as the Tomasulo algorithm. In turn, these novel processes evolved to reduced instruction set computers (RISC) in the mid-1980s through the contributions of John Cocke, which led to the series of IBM Power machines. Pipelining and its extensions are commonplace in today's computers. Asking students to explore these aspects of computing should perk and increase their interest regarding the manner in which companies build computers today.

4.4 Reflections

With respect with Sections 4.1 and 4.2 above, we should note that John von Neumann first proposed a blueprint of a modern computer in his 1945 white paper [13], from which we attribute the idea of the "von Neumann architecture" in his honor. Computers in the modern sense first appeared in the late 1940s. Before that time, computers were electronic computational devices such as the ABC special-purpose computer of Atanasoff in 1942 and the ENIAC general-purpose computer of Eckert and Mauchly in 1946. The ability of a machine to store dynamically and process data forms the basis of a modern meaning of a computer. Precursors to the computer in its modern meaning do provide much fodder for student-teacher discussion and interaction. Broadening the non-technical dimensions and understanding of the computing field is a recipe for sound and informed computing practices in one's career.

5 Engaging Students

Students do find the development and episodes of the computers mentioned in Section 4 intriguing. So, how does a teacher incorporate some of this learning in an introductory course? One presumption is that instructors have some familiarity with the work of pioneers or they can quickly learn this aspect of computing. It is good to do some show-and-tell activities of relays and valves (or vacuum tubes) if available or at least show pictures of them. In this modern day of smart phones and tablets, students are beginning to appreciate the challenges of the past; they also learn how past challenges have led to the devices they commonly use today.

We could ask some interesting questions related to computer hardware elements and devices. For example, have students ever held, touched, or even seen the internals of a computer with its components?

Computing topics often center on software and applications. Few students receive exposure to the electrical and electronic elements of a computer and even fewer have touched resistors, capacitors, transistors, vacuum tubes, memory chips, floppy disks, or motherboards. Generations of computer developments offer students the opportunity to examine these items and try to understand the challenges of space and heat problems others confronted with early machines. Hence, a discussion of people such as Zuse, Turing, or Atanasoff opens doors for exciting discussions, particularly the challenges they faced in building original computers. The discussion injects "humanizing" elements in the teaching of computing and technology.

Another challenge involves the teachers themselves. Are computing teachers and instructors sufficiently conversant with the history of the subject they teach? Probably not. It is unlikely that teachers have formally studied computing history; the knowledge they have is self-taught. Hence, teachers of computing and its technologies often shy away from its history in favor of technical approaches, even though they realize that knowledge of history could be useful in avoiding future pitfalls. A brief discussion of computing pioneers among students could greatly enrich computing topics and allow students to explore new avenues of learning.

6 The "IFIP Report"

At the IFIP World Computer Congress (WCC) in Canberra, Australia, JAN Lee and I met and discussed the development of a document that could become a guide for teachers to embed history in the computing curriculum. Of course, this was music to my ears since I was already doing those activities for almost a decade. JAN and I strategized ways to accomplish this end, particularly if it had international appeal. Since we were both at WCC, it became immediately clear that IFIP could be that vehicle. Since JAN already represented the Working Group 9.7, a working group of IFIP's Technical Committee 9 (TC9), it was useful to include a representative from Technical Committee 3 (TC3) on Education. Gordon Davies, then chair of computer science at the Open University in the U.K. and active in TC3, became a logical choice. We also recruited two computing historians, Michael Williams of Calgary and Martin Campbell-Kelly of Warwick, to form a team of five. The group designated me as their leader.

Over a period of two years, the team developed drafts of what would become the content of this document; it would eventually receive the approval of IFIP. One contrasting and diverse opinion was whether history in computing should be taught solely as a course or sprinkled throughout the curriculum. Another point of contention was whether one should teach a history course as a chronology of events or in some other manner. For the former point, the team decided to include both opinions to allow flexibility in approach. For the second point, the document suggested a "whatever works best" approach for a given situation. The team did not want to be prescriptive in its recommendations to IFIP; it simply wanted to increase the awareness of computing teachers allowing them to develop their own pathways on ways to use computing history. It also wanted to make teachers aware of resources available to them to enrich a student's learning experience through the inclusion of historical insights and dimensions.

The team titled the final document "History in the Computing Curriculum" and presented it to the IFIP constituents (TC9 and TC3) for approval in 1998, which it received. The document, also known as the "IFIP Report", was subsequently published in the *IEEE Annals of the History of Computing* in January of 1999 [4]. The contents of the thirteen-page report included sections such as (a) need for history content, (b) curriculum content, (c) implementation of a basic curriculum, and (d) assessment. Of significant interest was the Appendix to the report. It included among other things a chronology of historical events in computing, relevant websites, audio-visual materials, some examples of learning clusters, references to existing computing history courses, sample course syllabi, and listings of journals, textbooks, and other works on computing history. In all, the report received wide acclaim and it is still referenced to this day.

7 Experiences as WG 9.7 Chair

It would be good to explore how different circumstances have led me to acquire a zeal for organizing conferences and publishing books on computing history. The principal force for me was and still is promoting the use of history to motivate student learning. As mentioned earlier, my use of computing history was a normal theme in my courses since the late 1980s. I was and am passionate about computing history because students and professionals begin to think "outside the box" and look beyond machines and languages to see how the history of a topic broadens a person's outlook on the world. Computing history also allows one to investigate and research topics in the context of lifelong learning and make connections between technology and human reality. Indeed, computing history enhances the study of computing and expands student learning of the subject.

The von Neumann conference that I chaired in 1988, coupled with my desire to include computing history in my courses, coupled with the publication of the "IFIP Report" have all contributed to my escalation and interest in computing history. I joined WG 9.7 and soon thereafter started attending TC9 meetings. In 2001, JAN Lee decided to step down as chair of WG 9.7 and the TC9 appointed me as his successor. At that time, the working group was very small with only about a half-dozen members. I decided to change that image of "just a few" and by the end of my six-year tenure, well over sixty people were members of the working group, most of whom were historical dignitaries or pioneers in computing. IFIP allows participation as a working group chair only for two, three-year terms. In 2007, Arthur Tatnall succeeded me as its new chair.

The infusion of computing history through WG 9.7 provided a pathway toward expanding the message of computing history. The organization of events to herald the importance of history was destined to become pivotal benchmarks toward promoting that idea. As I am very fond of Nordic culture, I thought it best to launch an event that would "unite" the Nordic countries (Denmark, Finland, Iceland, Norway, and Sweden) under a common cause of computing history. Within a year, I was soon able to assemble the organizing and program committees composed primarily of Nordic representatives to launch the first WG 9.7 conference called the "History of Nordic Computing" (HiNC) that had taken place in 2003 in Trondheim, Norway [1]. More than seventy-five computing pioneers attended the event. The enthusiasm shown in Trondheim propelled the organization of a second event in Turku, Finland, in 2007 called HiNC2 [8]. The success of this event soon led to the organization of HiNC3 in Stockholm, Sweden, in 2010 [9]. Currently, plans are underway for a HiNC4 conference in Copenhagen, Denmark, in 2014. The original thought of uniting the Nordic countries under an umbrella of history has definitely become a sustained reality.

As the HiNC conferences began to gain prominence, another thread of WG 9.7 parallel events was taking place. At a 2003 TC3/WG 3.6 (Distance Education) conference in Geelong, Australia, some TC3 members met with JAN Lee and me to discuss the possibility of a WG 9.7 history conference on education, supported by TC3. Shortly thereafter in 2004, the first history of computing and education (HCE) event [5] had taken place in Toulouse, France, as a track within the IFIP World Computer Congress (WCC). As this was the first education event for WG 9.7, all were rather delighted to know that over two dozen people attended the presentations. At the IFIP WCC is Santiago, Chile, in 2006, the next HCE event occurred [6], where the focus was on history and education in the South American region. The success HCE2 became fodder for another history track within the WCC in Milano, Italy, in 2008 [7]. This was the last of the HCE conferences. At that time, participants decided it would be best to decouple the combined education and history themes; future WG 9.7 events would focus on computing history in the broader sense, which would even include education.

Perhaps the most extraordinary event I organized as WG 9.7 chair was the 2006 conference on "Perspectives on Soviet and Russian Computing" (SoRuCom) in Petrozavodsk, Russia [10]. This event had taken about five years in the making and necessitated traveling to Russia (at my own expense) several times. Petrozavodsk is approximately 435 kilometers north of St. Petersburg and 980 kilometers northwest of Moscow. The Petrozavodsk State University had taken an interest in hosting this event, which brought together over a hundred-fifty Soviet scientists, academicians, and interested parties to have a franc exchange of ideas and points of view. One must remember that in the time of the Former Soviet Union, people worked on projects under a shroud of secrecy. It was astonishing to see many of the computing pioneers finally meeting; they were able to discuss their projects openly and without restraint. The SoRuCom conference represented a landmark event in the preservation of the history of Soviet computing. The SoRuCom conference led to the SoRuCom-2 conference held on Novgorod, Russia, in September of 2011; I had little involvement with this conference other than being a member of its organizing committee.

8 Summary

We should remember that any course on the fundamentals of computers and computing is by its very nature a course on the history of computing. Unfortunately, teachers usually skip the non-technical nuances and focus only on technical issues. Historical excursions take only a few minutes to execute; the time invested could stimulate student interest and make them more interested in their studies. It is my hope that this journey through time as expounded by this narrative serves as an awakening that will motivate others to add historical dimensions to enrich their courses.

References

1. Bubenko, J., Impagliazzo, J., Sølvberg, A. (eds.): History of Nordic Computing. Springer, New York (2005)
2. Glimm, J., Impagliazzo, J., Singer, I. (eds.): The Legacy of John von Neumann. Proceedings of Symposia in Pure Mathematics, vol. 50. American Mathematical Society (1990)
3. IFIP, Technical Committee 9, ICT and Society, http://ifiptc.org/?tc=tc9
4. Impagliazzo, J., Campbell-Kelly, M., Davies, G., Lee, J.A.N., Williams, M.: History in the Computing Curriculum. IEEE Annals of the History of Computing 21(1), 4–16 (1999)
5. Impagliazzo, J., Lee, J.A.N. (eds.): History of Computing in Education. Kluwer Academic Publishers (2004)
6. Impagliazzo, J. (ed.): History of Computing and Education 2. Springer, Heidelberg (2006)
7. Impagliazzo, J. (ed.): History of Computing and Education 3. Springer, Heidelberg (2008)
8. Impagliazzo, J., Järvi, T., Paju, P. (eds.): History of Nordic Computing 2. Springer, Heidelberg (2009)
9. Impagliazzo, J., Lundin, P., Wangler, B. (eds.): History of Nordic Computing 3. Springer, Heidelberg (2011)
10. Impagliazzo, J., Proydakov, E. (eds.): Perspectives on Soviet and Russian Computing. Springer, Heidelberg (2011)
11. Lee, J.A.N.: International Biographical Dictionary of Computer Pioneers, 816 p. Taylor & Francis (December 1995)
12. McCormick, J.: George Santayana: A Biography. Transaction Publishers, New Brunswick (2003)
13. Von Neumann, J.: First Draft of a Report on the EDVAC. In: Aspray, W., Burks, A. (eds.) Papers of John von Neumann on Computing and Computer Theory. MIT Press, Cambridge (1987)

Author Index