Frank Harmsen
Knut Grahlmann
Erik Proper (Eds.)

Practice-Driven Research on Enterprise Transformation

LNBIP 89

Third Working Conference, PRET 2011
Luxembourg-Kirchberg, Luxembourg, September 2011
Proceedings

Springer

Lecture Notes
in Business Information Processing 89

Series Editors

Frank Harmsen Knut Grahlmann
Erik Proper (Eds.)

Practice-Driven Research on Enterprise Transformation

Third Working Conference, PRET 2011
Luxembourg-Kirchberg, Luxembourg, September 6, 2011
Proceedings

Springer

Volume Editors

Frank Harmsen
Maastricht University
Minderbroedersberg 6a, 6211 LK Maastricht, The Netherlands
E-mail: f.harmsen@maastrichtuniversity.nl

Knut Grahlmann
Ernst & Young Advisory
Antonio Vivaldistraat 150, 1083 HP Amsterdam, The Netherlands
E-mail: knut.grahlmann@nl.ey.com

Erik Proper
CRP Henri Tudor
29, avenue John F. Kennedy, 1855 Luxembourg-Kirchberg, Luxembourg
E-mail: erik.proper@tudor.lu

ISSN 1865-1348 e-ISSN 1865-1356
ISBN 978-3-642-23387-6 e-ISBN 978-3-642-23388-3
DOI 10.1007/978-3-642-23388-3
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011934744

ACM Computing Classification (1998): J.1, H.3.5, H.4.1, K.6.3

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)

Preface

The PRET working conferences are set up as a one-day event in such a way that they attract an audience from both industry and academia. PRET 2011 was jointly organized with the 13th IEEE Conference on Commerce and Enterprise (CEC). It was a continuation of the PRET 2010 working conference, which was organized as part of the Enterprise Engineering week in Delft, and the PRET 2009 working conference, which was organized as the industrial track at the CAiSE conference in Amsterdam.

PRET has always been intended to be an international conference, bringing together researchers and practitioners in the enterprise transformation field. By coincidence, the first two versions took place in one country, The Netherlands. It was therefore a pleasure that the 2011 edition was in Luxembourg, which is one of Europe's most international cities. The international character of enterprise transformation is without doubt one of the major challenges in the field—as such, transformation is a so-called wicked problem, but add the multi-culture, cross-border, European or even global dimension, and the challenge is even bigger. Luxembourg, with its long-standing heritage of European cooperation, inspired the PRET participants to deal with this challenge.

Enterprise transformation is indeed a wicked problem, i.e., a problem that cannot be solved with conventional measures, but for which the solutions have to be designed and constructed. This is not only due to its often international dimension, but to its complexity in general. Enterprise transformation affects an entire organization, and has a multitude of dimensions, of which culture, processes, technology, people and money are the most obvious ones. The intention of PRET is to consider these dimensions, either on their own or from a more holistic point of view.

Essentially, solutions for this multi-dimensional, wicked problem can be plotted on a scale with two ends. On the one hand, there is the *instruction-based* solution. In this type of solution, approaches and methods are invented (often, but not always, as a result of practice). These approaches and methods *instruct* the 'enterprise engineer' to design enterprises and to design enterprise transformations. On the other end of the scale, one can find the *principle-based* solution. This emphasizes the need for *design principles*, which guide the enterprise engineer toward models and designs. How the actual design will look like, or how the actual transformation should take place, is not specified – that is up to the enterprise engineer and the stakeholders in an enterprise transformation program. For enterprise transformation, this results in a spectrum of choices, which are of course highly situation-dependent – there is no single best way to conduct transformations, and researchers should be thankful for that.

The spectrum or scale between the instruction-based and the principle-based thinking corresponds to similar paradigms in other fields. Design thinking, which

has become popular in the last decade to explain the success of enterprises like Apple, can be considered as one of the more serious attempts to draw an analogy between enterprise transformation and 'design' in general. For instance, the aesthetics of a design (with attributes such as contrast, depth and rhythm), can be considered design principles. An aesthetic attribute such as contrast can be compared to the differentiators of a company. On the other hand, design studies have come up with rigorous engineering approaches, which can be applied in enterprise transformation as well. Design thinking positions itself as a paradigm to take on wicked problems, which should make it very suitable for our enterprise transformation challenges. I hope and expect that the PRET community, which originates from the information systems field, and the design thinking community, which is more geared toward organizational change management, will achieve the synergy that is required to advance the enterprise transformation field.

The third edition of PRET also marked a further collaboration between academia and practice. This year we invited some top practitioners to submit case studies. We did this with the specific purpose of cross-fertilization and learning: academia learning from practice about 'real-world' issues, and practitioners actually challenging the researchers to come up with good ideas for their transformation programs. This is why we accepted fewer academic papers, to give room for the case studies. As usual, the academic papers were all based on empirical research, to emphasize the "P" in PRET. The papers were submitted as book chapters, with an average size of about 25 pages. This size, enabling a slightly more in-depth coverage of the research topic, stimulated a meaningful discussion at the working conference, with the goal of developing the field of enterprise transformation, creating synergy and jointly identifying topics for further research.

June 2011 Frank Harmsen

Organization

Steering Committee

Erik Proper	Public Research Centre Henri Tudor, Luxembourg, and Radboud University Nijmegen, The Netherlands
Frank Harmsen	Ernst & Young IT Advisory and Maastricht University, The Netherlands

Organizing Committee and Program Board

Frank Harmsen (Chair)	Ernst & Young IT Advisory and Maastricht University, The Netherlands
Erik Proper	Public Research Centre Henri Tudor, Luxembourg and Radboud University Nijmegen, The Netherlands
Knut Grahlmann	Ernst & Young IT Advisory, The Netherlands
Khaled Gaaloul	Public Research Centre Henri Tudor, Luxembourg
Wolfgang Molnar	Public Research Centre Henri Tudor, Luxembourg

Program Committee

Antonia Albani	University of St. Gallen, Switzerland
Bas van Gils	BiZZdesign, The Netherlands
Bas van der Raadt	Schiphol Airport, The Netherlands
Eric Dubois	Public Research Centre Henri Tudor, Luxembourg
Erik Proper	Public Research Centre Henri Tudor, Luxembourg, and Radboud University Nijmegen, The Netherlands
Frank Harmsen	Ernst & Young IT Advisory and Maastricht University, The Netherlands
Gregor Engels	Capgemini SDM and University of Paderborn, Germany
Hajo Reijers	Eindhoven University of Technology, The Netherlands
Inge van de Weerd	University of Utrecht, The Netherlands

Jan Mendling	Humboldt University Berlin, Germany
Khaled Gaaloul	Public Research Centre Henri Tudor, Luxembourg
Marc Lankhorst	Novay, The Netherlands
Matthias Ekstedt	Royal Institute of Technology, Sweden
Patricia Lago	Free University Amsterdam, The Netherlands
Robert Lagerström	Royal Institute of Technology, Sweden
Robert Winter	University of St. Gallen, Switzerland
Sietse Overbeek	Delft University of Technology, The Netherlands
Stefan Strecker	Duisburg-Essen University, Germany
Stephan Aier	University of St. Gallen, Switzerland
Stijn Hoppenbrouwers	Radboud University Nijmegen, The Netherlands
Wolfgang Molnar	Public Research Centre Henri Tudor, Luxembourg

Supporting Organizations

Public Research Centre Henri Tudor, Luxembourg
Ernst & Young, The Netherlands

Table of Contents

Regular Papers

Case Studies

IT Performance Management in Five Organizations

(A Comparative Study)

Suzanne Haanappel[1], Roel Drost[1], Frank Harmsen[1,2], and Sjaak Brinkkemper[3]

[1] Ernst & Young Advisory, Antonio Vivaldistraat 150, 1083 HP Amsterdam, The Netherlands
{suzanne.haanappel,roel.drost}@nl.ey.com
[2] Maastricht University, Minderbroedersberg 6a, 6211 LK Maastricht, The Netherlands
f.harmsen@maastrichtuniversity.nl
[3] Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508TB Utrecht, The Netherlands
s.brinkkemper@cs.uu.nl

Abstract. This research provides insight in how organizations can achieve and optimize the benefits of IT performance management. Added value from investments is achieved through a focus on IT performance management that enables the removal of non value-adding activities and processes (and thereby organizational efficiency) and the enhancement of organizational effectiveness. In our work we have used a framework that contains IT performance management characteristics that are mapped to organizations, to determine their IT performance management maturity. Although the organizations subjected to our study operate in the same sector, our results show a high diversity in applied IT performance management approaches and the different maturity levels between organizations.

Keywords: Performance Management, Performance Measurement, Cost Allocation, IT Performance Maturity, Maturity Assessment.

1 Introduction

Organizations need to receive value from their investments and IT performance management can assist in this by making the performance of IT transparent. IT performance management is defined as the area of setting goals, governing and improving the performance of IT. IT performance management is an aspect of organizational controls and the achievement of organizational and strategic goals [1]. The management and control of IT is critical as the incomes of organizations are under pressure and return on investment is more important than before [2, 3]. At the same time, IT investments frequently result in unexpected, uncertain and undesired results [4, 5, 6]. Therefore appropriate measures are required that identify and create metrics for measuring the contribution of IT to the organization's value chain [7].

Measuring the results or return on IT is complicated. Benefits can be intangible [8] or it is impossible to measure these benefits independently, as a multiple of organizational activities has contributed to the improvement of a single business

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 1–27, 2011.

benefit making a clear allocation impossible. The difficulty with measuring IT performance has led to an increase in the evaluation and assessment of IT investments [9]. To deliver against business requirements, management needs to install an internal control system or framework [10]. Implementing IT performance management allows organizations to become proactive and prevents the chasing of non-measurable and/or opportunistic IT goals [2, 11]. Additionally evaluating IT investments enables a natural learning process for the organization. Examples of techniques to assess the performance of IT are: value chain assessment, relative competitive performance, strategic match analysis and evaluation [9].

Organizations adopt IT performance management in order to achieve specific objectives such as reducing expenses and creating a competitive advantage [7]. The need for organizations to achieve value on investments and become proactive, requires research in the area of IT performance management. Such research can support organizations to reap more benefits from IT performance management.

In practice, there seems to be a relation between organizational (situational) characteristics and the success of IT Performance Management endeavors. It makes no sense to apply sophisticated approaches and techniques in organizations that do not have the required maturity to benefit from them. Therefore, a one size fits all approach to IT Performance management does not exist. In order to investigate the above in more detail, we have developed a framework consisting of the characteristics an organization should have to achieve a high IT performance management maturity. These characteristics were mapped to five organizations to determine their IT performance management maturity level. Based on these maturity levels the organizations have been compared with each other.

This paper is structured as follows. After we have described our research approach, we present in section 3 the comparison framework, based on well-known existing frameworks. In section 4, we compare the organizations in our study based on our framework. We draw conclusions from this inter-organizational comparison and we confront these conclusions with a generic IT performance management approach. This generic approach is a synergy of the IT performance management approaches of the various organizations. After this confrontation, we present a short discussion. Finally, conclusions are drawn and suggestions for further research are given.

2 Research Approach

We have chosen Case Study Research (CSR) as the approach to conduct our study. Because there is little research available in the field of IT performance management, it is important to first identify what IT performance management is about, before anything can be said about this concept. Afterwards, the development of theories and testing them enables the construction of valid statements, which can be used for future research. CSR allows for the development of theories based on the research results. Furthermore it can combine data collection techniques which are both qualitative and quantitative and seem to be suitable for this research [12,13].

Some advantages of CSR are that variations in approaches (e.g. regarding instruments), allow for both quantitative and qualitative analysis of the data. In addition, detailed qualitative accounts can help explain complexities of everyday

situations which may not be captured through experimental or survey research. However CSR has also been criticized with the assumption that scientific valid research is not possible. Some of the disadvantages named are:

- Designing, scoping and collecting the results can be difficult and result in large amounts of data [12];
- Difficulties in generalizing research results and subjectivity of the data collection and analysis processes [14];
- A frequent criticism of case study methodology is that its dependence on a single case renders it incapable of providing a generalized conclusion [14].

To mitigate these disadvantages, we focused on one sector: the utilities sector. A maximum of five organizations was involved to prevent data overload. There were several interviews with different people from each participating organization, to overcome subjectivity problems. Next to these interviews, documentation from the organization was used, in order to compare the interviews with other sources.

Organizations have been made anonymous, but a short characteristic, retrieved from the 2009 annual reports of each organization, is provided below. We have used rounded revenue numbers (to the nearest 50 million) and FTE numbers (to nearest 500) to assure that the participating organizations remain anonymous.

The organizations presented above, were compared to the framework presented in section three. The framework is composed of IT performance management

Table 1. Information about participating organizations

Company	Full-Time Equivalent	Net revenue (in millions Euro)	Description
A	1500	450	Water company, responsible for water supply and nature conservation in several Dutch provinces.
B	500	150	Water company that is responsible for the water supply in one of the Dutch provinces.
C	7000	5100	International energy supplier that has the production and delivery of electricity, gas and water as its core activities.
D	3500	1400	Utility company that manages the energy network in several areas of the Netherlands.
E	5500	1450	Utility company that manages regional energy networks and provides network related services in the area of complex energy structures, in several Dutch provinces

characteristics retrieved from scientific literature. Characteristics were selected if and only if their construction process was clearly explained and substantiated. The result of the analysis of scientific literature was a list with multiple IT performance management characteristics.

After that the characteristics were categorized into a normalized process model, because they could be divided into preparing and executing activities that closely resembled the activities from this cycle. The characteristics were also divided into different IT performance management categories. These categories are based both on the context of the characteristics, but also on experience from the authors. The mapping and grouping of the characteristics has been discussed several times, with people experienced in the field of IT for validation purposes.

Table 2. Case Study Protocol

Activity	**Sub activity**
1. Getting started	1.1 Identify previous research 1.2 Identify main research question (MRQ) 1.3 Identify sub research questions (SRQs) 1.4 Provide overview of research project 1.5 Describe confidentiality and data storage 1.6 Form of publication
2. Case selection	2.1 Determine the use of a single-case or multiple-case approach and embedded or holistic design and show link with MRQ and SRQs 2.2 Define criteria for case selection (selection/number/establishing contact) 2.3 Scheduling of visits 2.4 Length of sessions 2.5 Equipment and stationery
3. Collecting Data	3.1 Craft instruments and protocols (qualitative and quantitative) 3.2 Data collection description
4. Analyzing data	4.1 Identify criteria for interpreting findings 4.2 Identify data elements to answer MRQ and SRQs and how to combine them into an answer 4.3 How triangulation of perspectives from multiple participants will be achieved 4.4 Description of “within case” analysis 4.5 Description of “cross sectoral” analysis 4.6 Consider range of outcomes and identify alternative outcomes
5. Plan Validity	5.1 Check construct validity 5.2 Check internal validity 5.3 Check external validity
6. Wrap up	6.1 Enfolding literature (compare with conflicting/similar literature) Limitations

The research consisted of four main steps: (1) research planning, (2) literature research, (3) case studies, (4) synthesis of performance management approaches. The planning resulted in a Case Study Protocol (CSP) roadmap for this research, which is depicted in table 2.

To compare the findings from the research, a questionnaire was developed. The data gathered through the questionnaire were used to answer the research questions. The questionnaire contained the following six main questions:

1. What is the context of IT within the organization?
2. Which goal does the organization want to achieve with IT performance management?
3. Which Key Performance Indicators are used, why and what are the results?
4. How is IT performance management located in the business?
5. How is IT performance management used to govern the organization?
6. How do you make sure that IT is both effective and efficient?

Some important aspects are:

- Triangulation of perspectives from multiple participants: Several persons from the same organization have been interviewed.
- "Within case" analysis: During the within case analysis the results of the cases have been compared to the findings from the literature study to find patterns.
- Cross sector analysis: During the cross sector analysis the different cases have been compared, using the framework described in section 3.

3 Comparison Framework

We have developed a performance management framework that allows for the comparison of performance management characteristics retrieved from literature. Our framework consists of two dimensions:

- Characteristics of *processes*.
- Maturity levels of *organizations*.

3.1 Performance Management Process Characteristics

The characteristics are categorized into a normalized process model. We have chosen to use the well-known Plan, Do, Check, Act (PDCA) processes as our process model. The characteristics identified can be divided into preparing and executing activities closely resembling this cycle. The phases of the cycle are developed by [13] and allow for optimizing and improving a single process model[16]. A short description of each of the processes is presented below:

- *Plan* to improve operations by identifying what is going wrong and planning solutions for these problems.

- *Do* implements changes to solve problems. First this is done on a small scale first to minimize disruption of routine while testing if they really work.
- *Check* if the small changes are achieving desired results. Also check key activities to ensure that the output allows for the identification of new problems when they arise.
- *Act* to implement the changes when they show to be successful. This is done by making them part of the organizational routine. Also involve other parties who are affected by the changes and whose cooperation is necessary to implement the changes on a larger scale and share experiences or lessons learned.

In addition to the PDCA process the characteristics were also divided into the different categories of IT performance management we identified. We chose and named these categories as we identified them as the main elements of IT performance management during the analysis of the wide range of scientific literature selected. As mentioned before literature was only selected when the IT performance management characteristics construction process was clearly explained and substantiated. The categories are:

A) *Performance management* is the area of setting goals, responsibility accounting and monitoring, analyzing, governing and improving the performance of IT
B) *Performance measurement* is the development and adoption of a strategic set of performance metrics and using them to plan, implement, operate and monitor the strategies, functions and processes of the organization with
C) *Cost allocation* is the allocation of IT costs to the business.

A - Performance management	
Process	**Characteristic**
Plan	1. Performance management requires good insight into the organizational processes [17] 2. Performance management should contain both the operation and business side of IT [18] 3. Before making changes to the performance management approach the current situation should be clear and understood [19]
Do	1. The vision and strategy should be translated into concrete goals which should be organized based on four points of view: financial, customer, internal organization and learning and growth abilities of the organization [2] 2. The IT performance of the organization should be discussed during regular meetings[8]
Check	1. Business objectives should be met [18]
Act	1. Reward employees based on the KPIs they are responsible for [20] 2. Use results for the development of new IT priorities [8] 3. Use performance management results to increase performance [21]. 4. Control or measure the performance based on a benchmark [22].

B - Performance measurement	
Process	**Characteristic**
Plan	1. Replace intuition by facts [21]. 2. Performance measurement frameworks need to be balanced, multidimensional (BSC), comprehensive and integrated [23,24,25] 3. Use a strategic and balanced set of KPIs to plan, implement, operate and monitor the strategies, functions and processes of the organization [26] 4. Measurement frameworks should be complemented by performance measurement techniques and improvement initiatives based on the requirements and goals of the organization [23] 5. Performance measurement should be based on the current strategy and can include metrics that anticipate on future goals for the organization [18]
Do	1. Performance measurement requires a mixed approach to measure both financial and non-financial aspects [24,25] 2. Management teams need to be explicit about their performance priorities and corresponding relationships [20] 3. Measure performance by using performance measures that support critical business processes [27] 4. Performance criteria should be well-defined [28] 5. Task and responsibilities with regard to the flow of information should be explicit [20] 6. Performance measures should reflect the requirements and goals of the organization [23]
Check	1. The measurement system needs to be aligned with the company's goals to reward people in proportion to their performance on the measures that are important [29]
Act	1. Performance measurement results require actions [18] 2. Methods for taking new performance measures should evolve as the company's experience increases [29]

C - Cost allocation	
Process	**Characteristic**
Plan	1. The shared service center should be based upon the business planning [30] 2. A clear hierarchy and explicit responsibilities should be in place [30] 3. There should be a buyer and seller in the form of an IS and a user department [30]
Do	1. Internal allocation is needed to gain acceptance and success of the shared service center [32] 2. The costs created by the shared service center are allocated to departments which use the services [30] 3. The elements that make the costs are placed in a charge model [30] 4. The IT environment architecture should be developed and activities identified [32] 5. Employees should get support from the business [32] 6. All parties involved should participate in the business case to create commitment to the changes that will come [32] 7. Communication channels and committees should be created to guarantee a good governance structure [32]
Check	1. Insight into the cost structure is needed to manage the relationship with customers [33] 2. There should be management by exception where both favorable and unfavorable variances get attention [34] 3. Favorable and unfavorable variances should be analyzed [34]
Act	1. Corrective actions based upon the variances found in the Check-phase should be launched [34]

3.2 Maturity Framework

In addition to the framework an IT performance maturity model was developed. This model allows for the inter-organizational comparison of the organizations that participated in the research. The maturity model is based on the Capability Maturity Model. The CMM is intended to provide software organizations with guidance on how they could gain control of their processes for developing and maintaining software[35]. However the maturity levels can also be used to define performance management maturity levels, because performance management is aimed at gaining control of the process. CMM and IT Performance Management bear a lot of similarities, especially regarding process improvement.

The framework defines five maturity levels that form the foundation for continuous process improvement. In this research the description of maturity levels by [35] is adapted to fit the scope of the research. Maturity levels can be defined as evolutionary plateaus toward achieving a mature performance management process. Each of the levels comprises a set of process goals that, when satisfied, stabilize an important component of the performance management process. Below a short description of the five maturity levels for IT performance maturity is provided.

1. *Initial:* The IT performance management process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort. During this phase the employees are struggling against the process, or inventing it as they go along [36].
2. *Repeatable:* Basic IT performance management processes are established to measure and manage IT performance and to identify related IT costs created throughout the organization. The necessary process discipline is in place to repeat earlier successes with the measuring and management of IT and identifying the IT costs.
3. *Defined:* The IT performance management process for measurement, management and cost allocating is documented, standardized, and integrated into a standard IT performance management process for the organization. This standard IT performance management process is used every time an activity related to measurement, management and cost allocating is performed.
4. *Managed:* Following the Defined level, the organization-wide process used at level three has been instrumented so that it is quantitatively understood and controlled [36]. Detailed measures of the IT performance measurement, IT management process and cost allocation are collected. All elements are understood and controlled.
5. *Optimizing:* At this level, the organizations IT performance management processes operate as a matter of routine and stimulate people to focus on continuous improvement [36]. Continuous IT performance management improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

To be able to compare the organizations based on their maturity, we have determined for each characteristic of the IT performance management aspect a value: (3) satisfied, (2) partially satisfied or (1) not satisfied. When a characteristics and all of its elements were fully applied by an organization, value 3 was granted. When a characteristic was partially applied, i.e. not all elements were implemented value 2 was applied and when the characteristic was not implemented at all value 1 was granted.

The current set of organizations used in this research was not large enough to conduct a statistical comparison therefore, we have compared the organizations on a case by case basis. After the characteristics has been assigned their value, the average per phase and average per organization were calculated. We have mapped these averages to the capability maturity levels to compare the results and to develop an hypothesis about the potential relations between results.

4 Comparing the Five Organizations

This section presents insight into the level organizations have implemented the IT performance management characteristics. These results can be used to visualize possible similarities between implementation levels of the characteristics. Additionally the results can be used to develop hypothesis and show where organizations can make improvements.

4.1 Maturity Levels for Performance Management

Based on the values explained in the previous section, the following levels per category of IT performance management were calculated for each of the organizations. Table 3 depicts the distribution of implementation levels over the PDCA cycle and the organizations for performance management.

Table 3. Maturity levels for each of the organizations

Performance Management	Company A	Company B	Company C	Company D	Company E
Plan	1,6	1,6	2,8	2,6	1,8
Do	1,5	1,0	2,0	2,5	2,0
Check	1,0	1,0	3,0	3,0	1,0
Act	1,0	1,3	1,3	2,0	1,8

Table 3 shows that while some organizations have high maturity levels in one process, they do not score similar in the other processes. Additionally it shows that where some organizations score rather high on the implementation level of the performance management characteristics, others score low and do not seem to have implemented any of the characteristics in their organization. In figures 1 and 2 the averages per phase and per organization are presented.

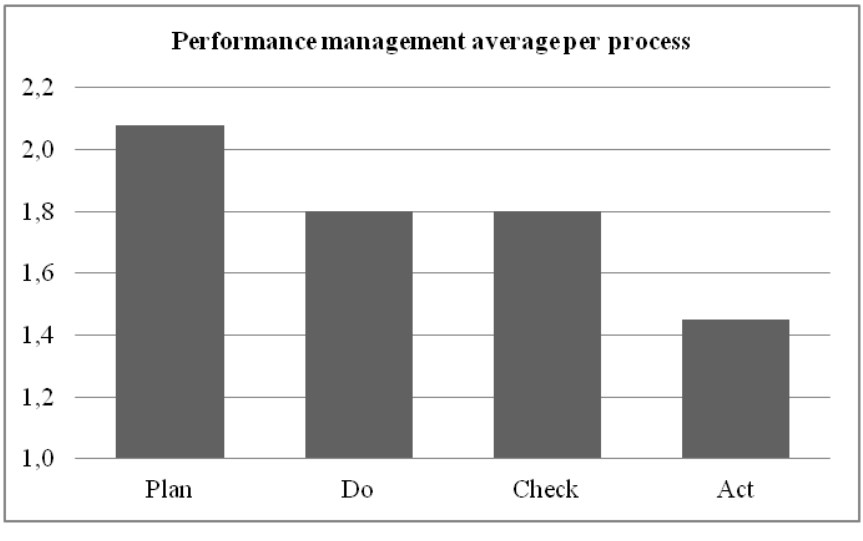


Fig. 1. Performance management average per process

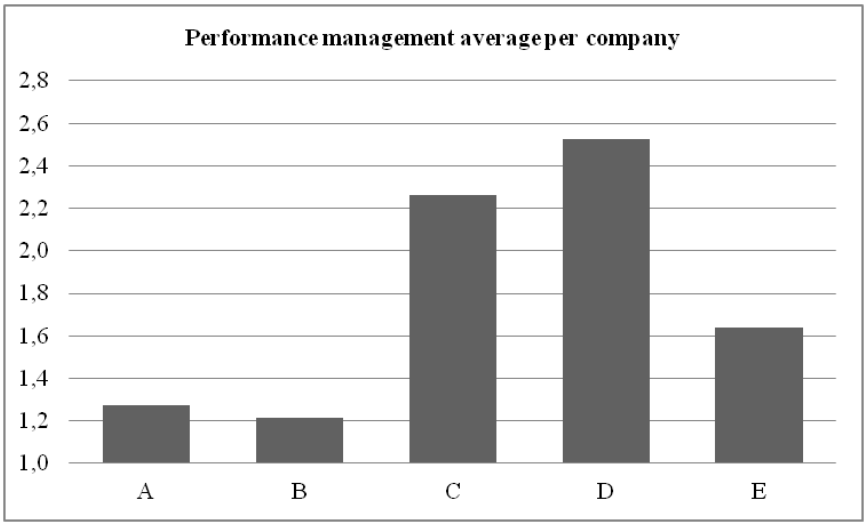


Fig. 2. Performance management average per company

Figure 1 shows that the implementation levels of performance management are unequally spread over the processes. It can be concluded that, with exception of the Plan process, the average of the organizations has not even partially implemented performance management characteristics belonging to the other processes. The value two of Plan in figure 1, means that the majority of organizations have at least partially implemented the performance management characteristics. The lower the value of a process, shows that the average implementation level of the organizations is low. We talk about average, because there can be organizations that have partially/fully implemented the characteristics, but this value is influenced by other organizations that have not implemented any of the performance management characteristics, resulting in a lower average.

When comparing the results per phase with the maturity levels per company, it shows that these results are diverse. Company A, B and E score below value two of partially implementing performance management characteristics. But company C and D have already partially implemented characteristics for the majority of the processes. The results from company A and B show that for the half of the processes the organizations have not implemented any of the performance management characteristics.

4.2 Maturity Levels for Performance Measurement

Table 4 presents the distribution of performance management maturity levels. The table shows that, similar to the performance management results, organizations have divergent implementation levels. Below the average per process and organization are presented.

Table 4. Maturity levels for each of the organizations

Performance Measurement	Company A	Company B	Company C	Company D	Company E
Plan	1,2	2,0	2,0	2,2	2,2
Do	1,0	1,0	1,5	2,5	2,3
Check	1,0	1,0	2,0	3,0	2,0
Act	1,0	1,5	2,0	2,0	1,0

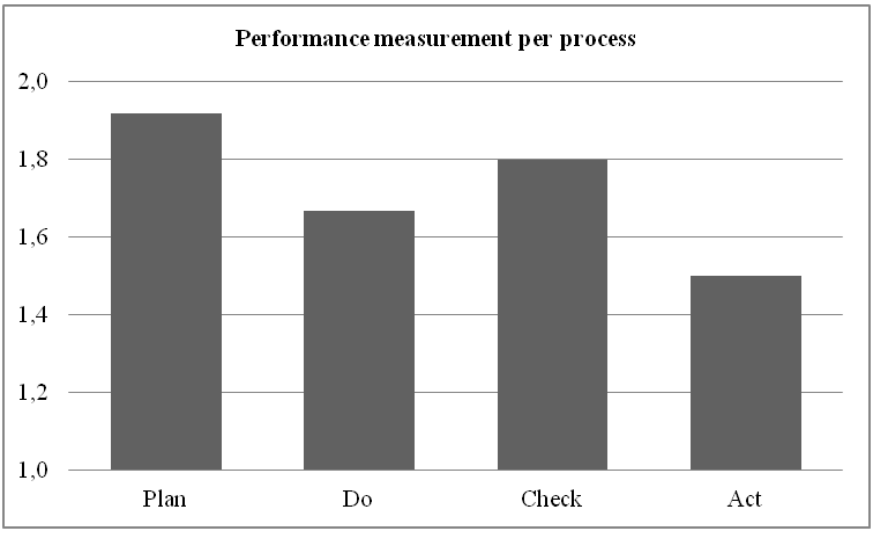


Fig. 3. Performance measurement average per process

All processes have a maturity value higher than one, meaning that organizations have implemented at least some of the characteristics for some of the processes. However from table 4 it shows that only company C and D have implemented characteristics for all four processes. Differences between the processes are currently strongly influenced by the number of characteristics in the framework. If a process contains only one characteristic, this characteristic determines the maturity level. When a process contains a larger number of characteristics, the more extreme values are neutralized by the values of the other characteristics. Over time, when new characteristics are added to the framework, the large influence of single

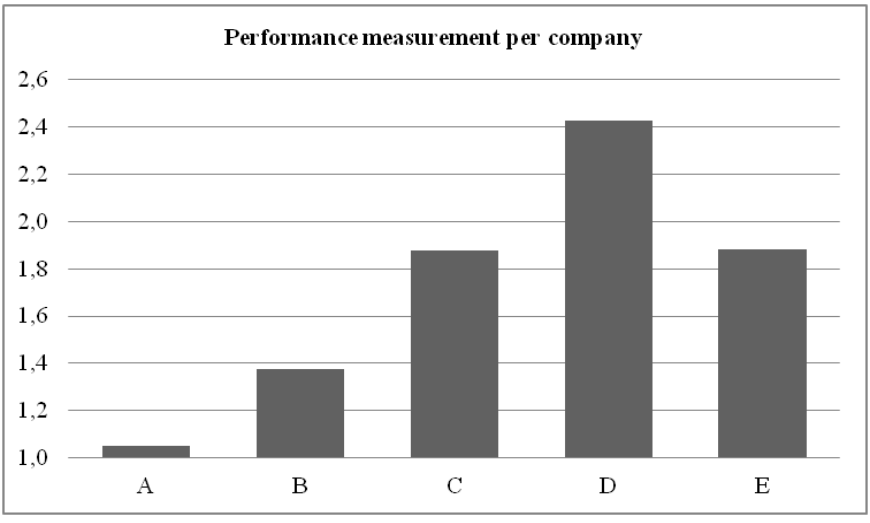


Fig. 4. Performance measurement average per company

characteristics on the maturity level will decrease. At that point dependencies between the phases could be identify, which is currently prevented by the large influence of some of the characteristics.

The organizational average performance measurement maturity levels for the companies are diverse. Where company A has barely achieved value 1, company D has partially implemented performance measurement characteristics for all processes. In this overview company A and B have, as in the performance management overview, the lowest values. Company D has once again the highest value and has partially implemented characteristics for all processes in both the performance and performance measurement area.

4.3 Maturity Levels for Cost Allocation

The cost allocation overview shows the following averages per organization and process. For company A no values are entered as this subject was not covered in enough detail during the interview. This should be taken into consideration when analyzing the averages per phase, as these consist of one less value in comparison with the phases of the other IT performance management elements. Additionally the column with the cost allocation average for company A is empty.

Table 5 shows that the implementation averages per organization vary, just as in the other tables presented for the previous IT performance management elements. It has to be noted that company A is not visualized in figure 6 as the subject of IT cost allocation was not subsequently covered during the interview. Where some organizations have only partially implemented cost allocation characteristics other have partially, or as in the plan process of company C, fully implemented these characteristics. The differences between the processes and companies are presented in figure 5 and 6.

Table 5. Maturity levels for each of the organizations

Cost Allocation	Company A	Company B	Company C	Company D	Company E
Plan	N/A	1,0	3,0	2,0	1,7
Do	N/A	1,6	2,3	1,7	2,1
Check	N/A	2,3	2,3	1,3	2,7
Act	N/A	3,0	2,0	1,0	1,0

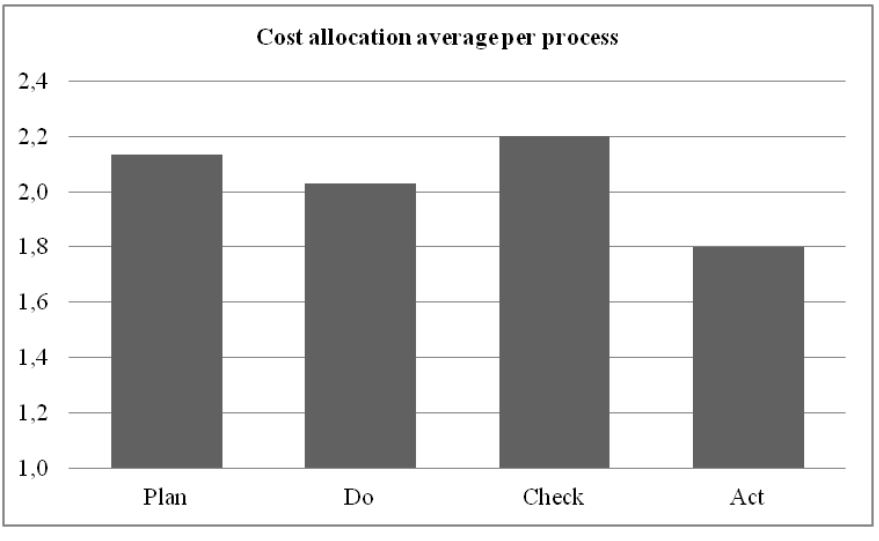


Fig. 5. Cost allocation average per process

Analyzing the implementation levels for the different processes, it shows that they are all at a quite similar levels. However only the process check is above the partially implemented level. Even though the distribution of maturity over the processes is quite similar, the distribution of maturity over the organizations is diverse. This also shows when looking at the actual averages per company and process in table 5. Where in some processes the characteristics are partially implemented, or the value is even higher than 2.0, in other processes values are much lower.

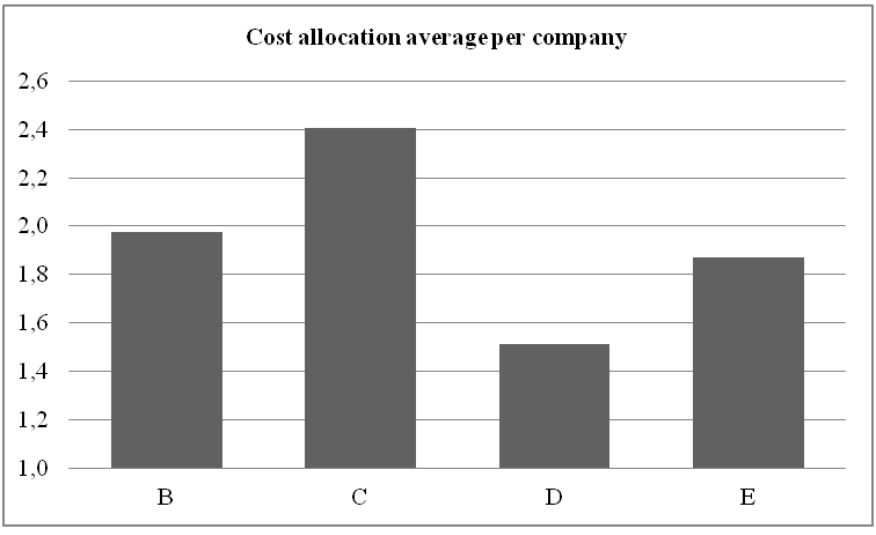


Fig. 6. Cost allocation average per company

4.4 IT Performance Management Average Per Company and Phase

This research has used a limited set of organizations, which results in hypothesis to explain the distribution of results as it is. Future research with larger samples may verify or falsify these hypothesis. Based on the IT performance management maturity levels for the companies and processes, presented in figure 7 and 8, the following hypothesis are made.

Differences in implementation level of the different processes seem equally spread, however with exception of the process Plan for performance management and cost allocation the implementation level of the processes is lower than two. This means that for almost all processes the characteristics are not even partially implemented. Reason for this might that organizations do not have a structured approach to get all processes on a similar level. Additionally it could indicate that organizations do not find all processes equally important or that they find it hard to implement some of the characteristics. Benefits that organizations are missing out on, are the full use of their collected results throughout the IT performance management cycle. Also results might not contribute to organizational goals anymore as they have not been correctly translated into actions during the phases, therefore not delivering the necessary details.

With regards to the organizations, differences in their IT performance management implementation levels lie in the fact that not all organizations have been engaged in IT performance management for long. Some of the organizations from the utility sector suffered from political demands, which resulted in the discrepancy between organizations that provide energy and the organizations that manage networks and transportation, and had to build their IT performance management again from scratch. Additionally there was also a company where the use of IT performance management results to support strategic goals was not pursued, therefore there was no priority in achieving high implementation levels.

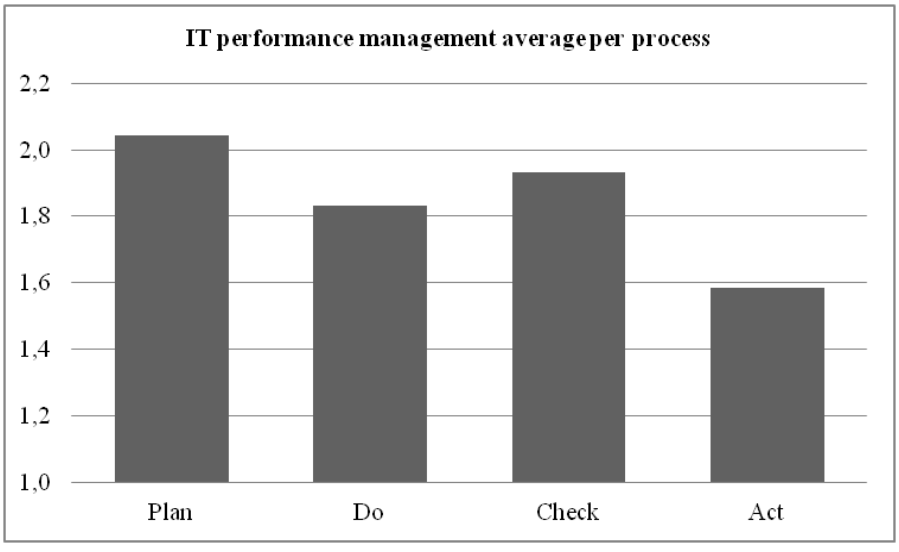


Fig. 7. Average IT performance management maturity level per process

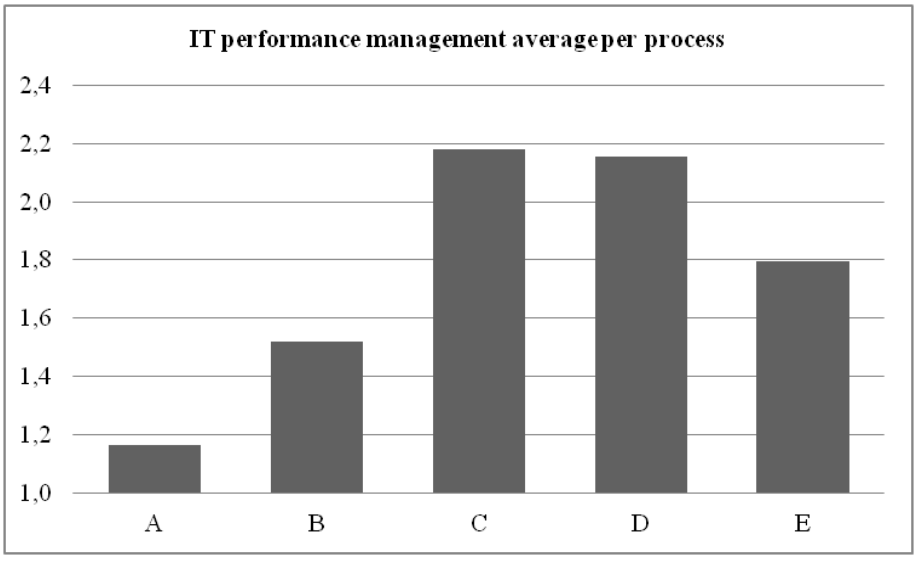


Fig. 8. Average IT performance management maturity level per company

In addition to the comparison of the IT performance management average per process against the IT performance management average per organization we calculated the correlation between IT performance management maturity versus the (total) number of employees per organization and between the IT performance

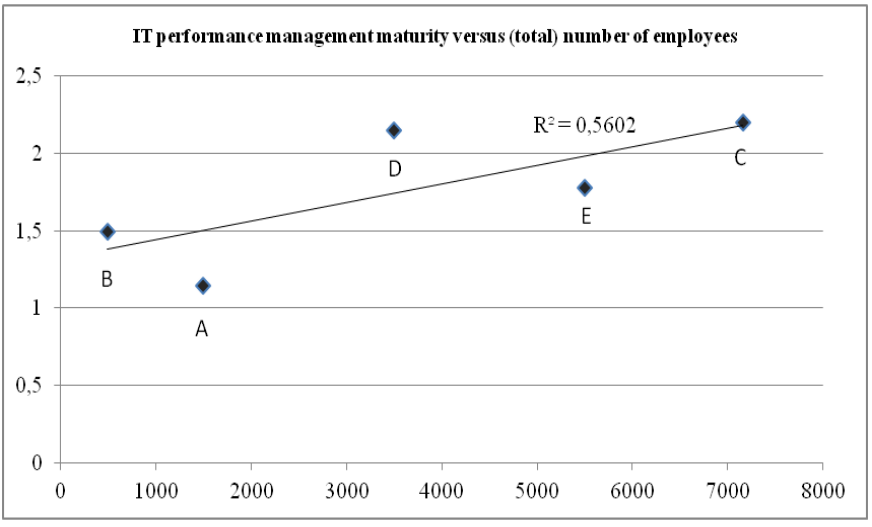


Fig. 9. IT performance management maturity versus (total) number of employees

management maturity versus the 2009 net revenue of the organizations. The correlation coefficient (R2) are presented in figures 9 and 10 and show values of 0,5602 and 0,516, respectively.

These results show that there is a weak correlation [37] between IT performance management maturity and the number of employees and that there is a weak correlation between IT performance management maturity and organizational net revenue. For example, in figure 9 it can be observed that organizations E and B score similar in terms of IT performance management maturity, even though there is a 10 times difference between their total number of employees. On the other hand, the weak correlation can be used to estimate the organization size where 'full maturity' (level 3) is achieved: by means of extrapolation one finds that from a company size around 14.000 employees an organization would have maturity level three ('all IT performance management characteristics are fully implemented').

The observation made with regards to figure 10 are similar to those for figure 9. For example, it can be observed that organizations D and C score similar on IT performance management maturity, even though there is an almost five times difference between their organizational net revenue.

Based on the interviews we found that the organizations are in different stages of the IT performance management implementation process. Obviously, the maturity is less dependent on organization size and depends on 'years of experience'. These findings seem to support the statement that it takes time to reach a certain maturity level: organizations C and D have multiple years of experience with IT performance management while some of the others just started with it. We have no reliable data on 'years of experience' for the organizations that participated in this research to prove our statement. This could be part of future research.

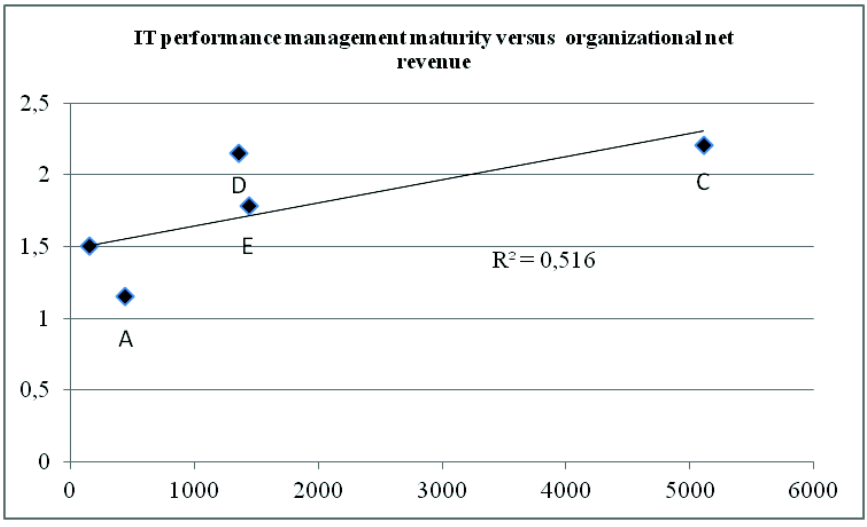


Fig. 10. IT performance management maturity versus organizational net revenue

5 Comparing Theory and Practice

The IT performance management approaches by the different organizations were collected and merged into a general performance management approach. This generic method contains an overview of the organization's IT performance management activities and allows for the comparison of the activities abstracted from literature, based on the requirements presented in section 2 research approach, with the activities organizations apply.

The method uses the process-data diagram notation, such as described in [38]. This notation divides the process into a left side that contains the activities of the method and a right side where the deliverables/concepts of the process are presented. To develop the diagram all IT performance management processes of the organizations were analyzed and their key activities were extracted. An example of this individual abstraction process is presented below. The abstracted activities were put into one overview, where similar activities were merged. This resulted in three activities on the process-data side of the process-data diagram:

1) Annual KPI cycle
2) Proactive monitoring
3) Reactive monitoring

Example **IT performance management activity abstraction – Company C**
The process-data diagram of the generic IT performance management approach (figure 11) contains the IT performance management activities of all participating organizations. The abstraction steps that have been applied to these organizations are presented below after which they are applied to company C, which will serve as an example.

Step 1. Interview summary
The first step in the abstraction process was the development of an interview summary. This resulted in two summaries per organization. The interviews were structured based on the subjects of the interview questions: context of IT, IT performance management goal, Key performance indicators, IT performance management location and governance and IT effectiveness/efficiency. Once the summary was finished it was send to the corresponding interviewee who could make changes and suggestions. These suggestions were processed after which the organizations would give their final approval for further usage of the summary.

Step 2. Activity abstraction
After the official approval was granted the interview summaries were compared and the organizational IT performance management process was abstracted. The first step was to determine the starting point of the IT performance management process. In most situations this was the yearly activity of setting the organizational IT direction. As the starting point of the management cycle was determined the subsequent steps were identified based on the interview summaries. The advantage of using two interview summaries was that in some situations they could complement each other, providing a more detailed organizational IT performance management process.

Step 3. IT performance management process visualization
Once all activities were identified they were named in a short descriptive way. Additionally the activities were analyzed to see which activities occurred in parallel. After the naming and process analysis was completed the IT performance management process was visualized. The shapes used for this visualization are presented here.

Mandatory shape, used to represent the starting point of the process: ●.

Mandatory shape which is used to represent an activity: (Activity A).
The name of the activity is put inside the shape.

Optional shape which is used to represent both the fork and the join: ▬.
A fork is used to indicate the starting point of activities that occur in parallel. The join is used to indicate from where activities no longer occur in parallel.

Optional shape which is used to represent a choice/decision point: ◇.
When there is a condition attached to an activity a diamond is put after that activity. From the diamond two lines emerge. One line is used for when the condition is met and one for when the condition is not met.

Mandatory shape used to represent the end of the process: ◉.

Company C example

Step 1. Interview summary

After the interview was conducted the data was summarized in an interview summary. This summary was send to company C, where the interviewees made several suggestions and changes. These were processed after which the summary was send to company C for final approval. Once this approval was granted the second step, activity abstraction, could be put in process. The interview summary will not be put in this section as it would violate the privacy agreements made with the participating organizations.

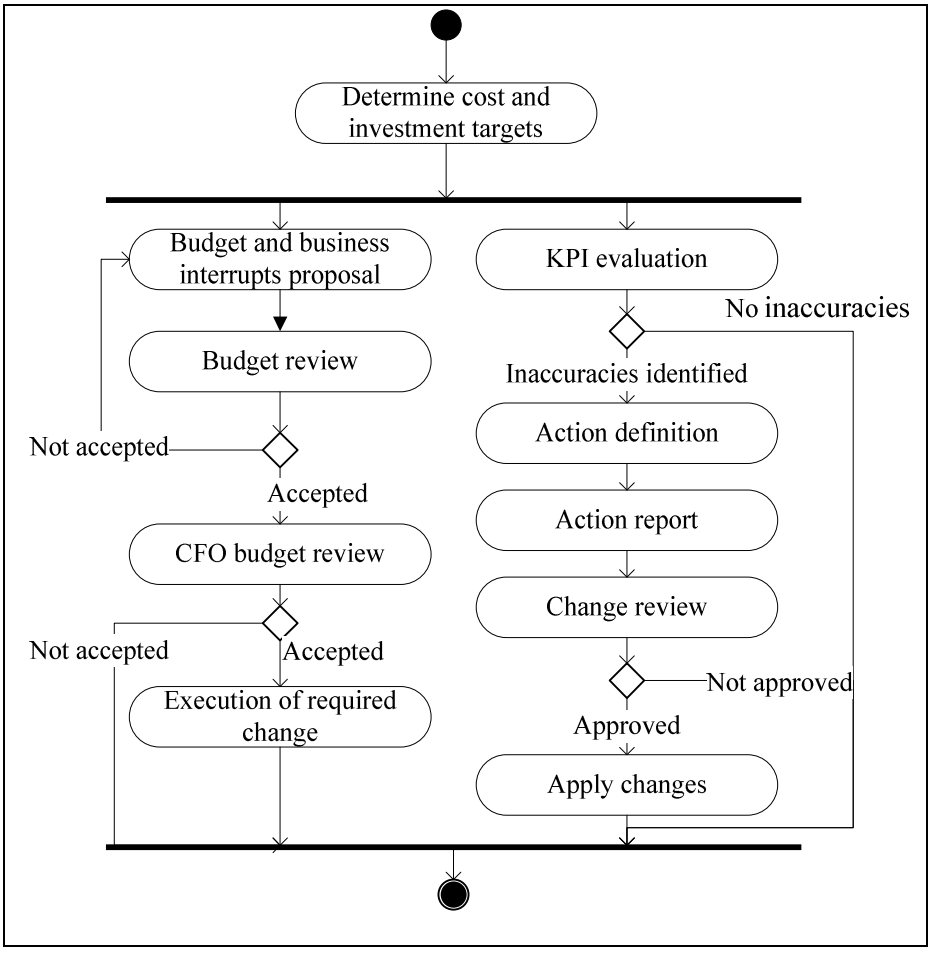
Step 2. Activity abstraction

During step two the interview summary was analyzed and the steps of the IT performance management process were extracted. As the process was not discussed activity wise the activities had to be extracted from the summary and named in a short an descriptive way, before they could be placed in the organizational IT performance management process. This step resulted in the following ten activities:

Activity Name	Description
Determine cost and investment targets	Yearly corporate business and IT management meeting during which the cost and investment targets are set.
Budget and business interrupts proposal	The domain manager proposes a budget for several KPIs.
Budget review	The Management Team will review the budget and either accept or reject the proposal.
CFO budget review	The Chief Financial Officer will review the budget and either accept or reject the budget proposal.
Execution of required change	The changes that are required after the approval of the budget proposal are executed.
KPI evaluation	A monthly evaluation of the KPIs to reveal potential inaccuracy of these KPIs.
Action definition	Employees define activities to overcome this inaccuracy.
Action report	Employees report the suggested activities to management.
Change review	Management reviews the proposed activities.
Apply change	Employees apply the proposed change.

Step 3. IT performance management process visualization

Once all activities were identified and described the right order of activities was determined. In addition it was analyzed if activities could occur in parallel and if decision points had to be entered into the process. This resulted in the following IT performance management process.



During the analysis the activities derived from the interview with the organizations were divided into these three activities. This decision was based on the fact that the organizations indicated that their IT performance management activities, could be divided in different phases. The names have been developed by the authors and are based on the objective of the sub-activities of the activities.

In the Annual KPI cycle, KPIs are developed based on the organizational goals. Some organizations have, in addition to organization wide also departmental KPIs. The organizational KPIs are constructed by CEOs, sometimes in collaboration with the CFO. In the proactive monitoring activity employees are allowed to define actions for improvement possibilities they identify. This starts with an employee that signals an improvement and defines actions that should be taken. These actions always need to be approved by the board, before they are implemented, due to governance and budget issues. After approval by the board the KPIs can be adjusted. However improvements can also be declined, which results in that they are not implemented. Another flow in this activity is that KPIs are adjusted during the year. Reasons for this

could be a business case, an possibility for improvement or a requested adjustment to one/more KPIs. This would also result in a request for change to the year plan, which will be approved/disapproved by the board. After this adjustment has been approved it will be implemented the same way as a KPI improvement by an employee.

In the reactive monitoring activity sub-activities are assigned to employees. In this activity there is no anticipation on trends or potential improvement possibilities. Employees do not propose changes to KPIs. Employees are evaluated based upon the KPI results for the KPIs they are responsible for. In some organizations the KPI evaluation reports with abnormalities can lead to KPI adjustment. However there are also organizations that keep on using the KPIs that are not measuring what they need to measure and only adapt them during the next annual KPI cycle.

The concepts per activity which can be found on the right side of the process-data diagram, are presented in table 6.

Table 6. Concepts of the Generic IT performance management approach

Concept	Description
KPI	An organizational, departmental or adjusted Key Performance Indicator.
KPI OWNER	An employee made responsible for a KPI
MONITORING RESULT	The results of the KPIs monitored
KPI EVALUATION REPORT	An evaluation report which contains the monitored KPI results
EMPLOYEE EVALUATOR	Someone in the organization who evaluated the performance of employees
EMPLOYEE EVALUATION REPORT	A report used by the EMPLOYEE EVALUATOR which contains the performance results of the employee
REQUEST FOR KPI CHANGE	A request for a KPI change which is suggested by employees. After approval by the board the REQUEST FOR CHANGE can become an approved REQUEST FOR CHANGE.
IMPROVEMENT	An improvement is defined by employees based upon a REQUEST FOR CHANGE

Comparing the process-data diagram (process) and the ITPM characteristics (framework) results in two remarkable findings. First of all the process-data diagram consists of three activities, which are Annual KPI cycle, Proactive monitoring and Reactive monitoring. Mapping these on the findings from literature it is apparent that organizations structure their IT performance management process differently. We identified, based on literature, that the IT performance management process consists of the areas IT performance management, IT performance measurement and cost allocation. This difference may result from the fact that there is little scientific literature available on IT performance management approaches and there is only a very limited amount of guidelines available for organizations on how to structure their IT performance management approach.

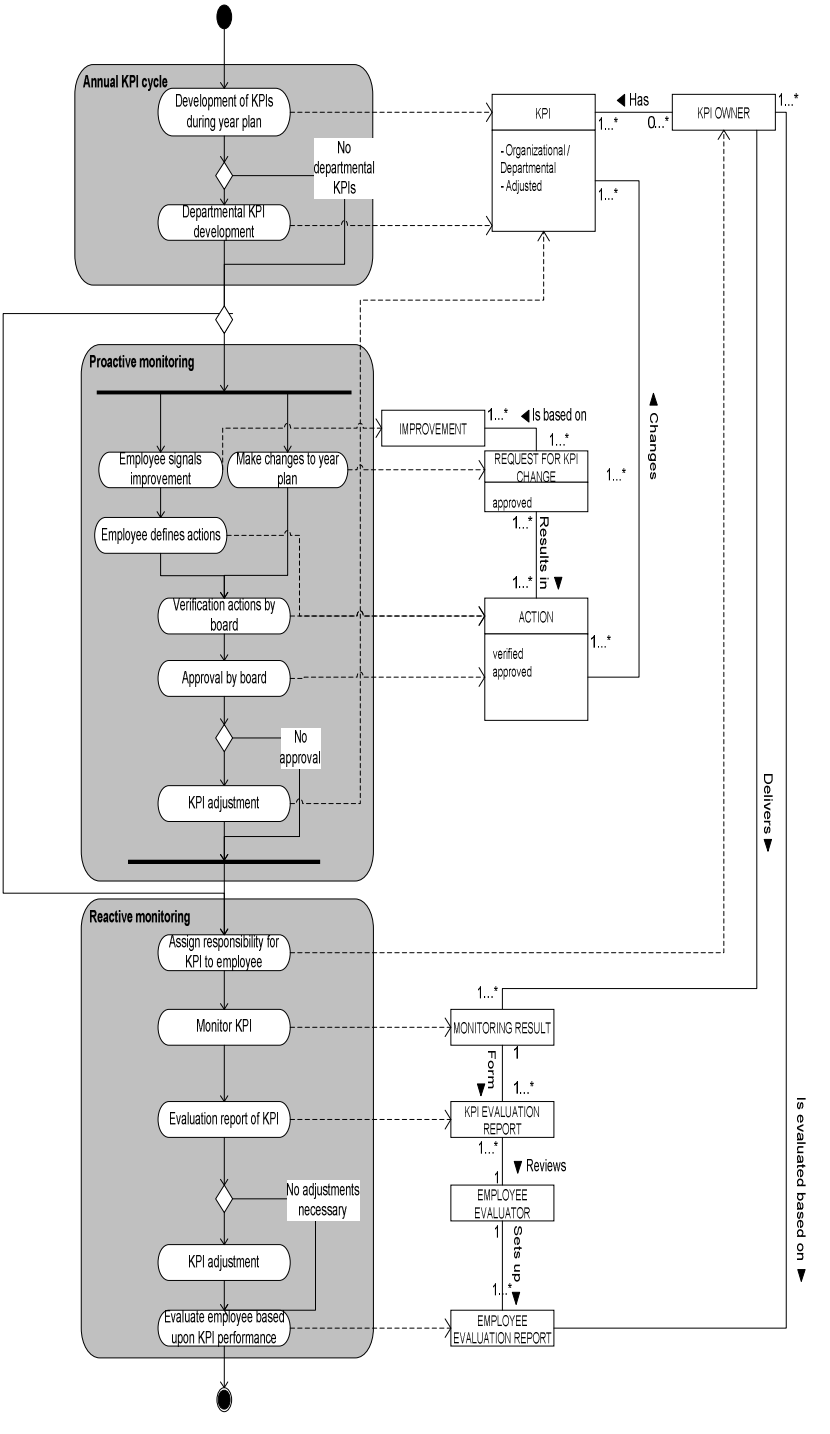


Fig. 11. Process-data diagram of Generic IT performance management approach

Second, even though the process-data diagram contains several characteristics from literature most of the characteristics are not applied. This could be due to the fact that literature analysis characteristics come from a scientific point of view and organizations have a more practical approach. There could be additional factors influencing the organizational IT performance management approach that were not considered in scientific research. Additionally in literature is it possible to have two totally separate phases, while in practice there often is a grey area in which phases overlap. Another possibility is that the processes and organizations do not have a high maturity level and therefore have not implemented all characteristics, as some might not be applicable for lower maturity levels.

6 Conclusion and Future Research

Based on data gathered during this research the research question: *"How do companies achieve and optimize the benefits of IT Performance Management?"* was answered.

The process-data diagram, shown in figure 9, presents the approach organizations use to achieve and optimize the benefits of IT performance management. The diagram shows that only a few of the organizations use performance results of the previous year when developing their new year plan. Even though using performance results could benefit the organizations when identifying problem areas or processes that could be improved. Organizations use performance results to achieve and optimize the benefits of IT performance management. Benefits of IT performance management can also be achieved when organizations use their performance results to develop new IT priorities. However currently none of the organizations use these results to increase their performance.

It seems that organizations are not aware that these results could also be used to improve IT performance management, or they do not know how to, as it has not yet been done. Organizations also try to achieve benefits from IT performance management by making employees responsible for KPIs and letting them identify and suggest improvements to KPIs. These improvements are always approved by the board of the organization to make sure are aligned to the organizational approach. Cost allocation is also used to achieve benefits of IT performance management by making employees aware of the costs involved with IT. Cost allocation can contribute to the benefit of IT performance management as budgets are allocated to departments forcing them to think about what IT can mean to their department and the plan required to realize these potential benefits.

Based on the dataset gather during this research it can be said that the organizations are not mature enough to optimize their IT performance management benefits. This because they have not yet been able to develop a loop back into the IT performance management cycle to use the results to their benefit. This also shows from the maturity levels presented in chapter 3.

6.1 Improvements and Future Research

Some points of improvement for the research are that the literature study and interviews should not have been conducted in parallel as this prevented the interview

to cover all characteristics in enough detail. Even though the researchers anticipated on this risk by having an elaborate interview, it turned out that even this interview did not cover all elements in detail. As the response time of several of the companies was long it was decided not to send out additional questionnaires. The second point of improvement could have been the time span and commitment to the interviews. A measure was taken to prevent that research questions had to be answered with an incomplete set of answers by sending the questionnaire back with the request to answer the remaining questions, however this was not successful as the response was slow and insufficient. The deliverables of this research are the starting point for follow-up research in the area of IT performance management. We recommend that future research addresses the development and identification of new IT performance management characteristics for more accurate determination of how organizations achieve and optimize the benefits of IT performance management. Additionally the IT performance management framework should be updated, so it can serve as a checklist for organizations to determine their IT performance management maturity level. More interviews can supplement the framework with characteristics that have proven to be of value for the organizational IT Performance Management process. This data can also be used to determine whether the characteristics from literature are applicable in real life situations, or only in situations presented in literature where there are no environmental factors that have an influence. If necessary characteristics from the framework may need to be removed. The data retrieved from more interviews can also be used to supplement the super method that was developed during this research. New activities can be identified and merged into the process-data diagram and the process-data diagram can be improved by deleting activities that prevent a good IT performance management approach. The combination of the IT performance management framework and the IT performance implementation levels could be used as the basis for a quantitative study to identify relations between the implementation levels of the different elements of the framework.

References

[1] Kang, H., Bradley, G.: Measuring the performance of IT services: An assessment of SERVQUAL. International Journal of Accounting Information Systems 3, 151–164 (2002)

[2] De Boer, J.C.: Het dashboard van de ICT-manager. Compact, 9–15 (March 2002)

[3] IT Governance requires performance measurement, http://www.kwaliteitopmaat.nl/Images/PSQT2004%20IT%20Governance%20requires%20performance%20measurement_tcm158-35896.pdf

[4] Van Grembergen, W., Van Bruggen, R.: Measuring and improving corporate information technology through the balanced scorecard technique. In: Proceedings of the Fourth European Conference on the Evaluation of Information Technology, pp. 163–171 (October 1997)

[5] Turban, E., McLean, E., Wetherbe, J.: Information Technology for Management – Making Connections for Strategic Advantage. John Wiley & Sons Inc., USA (2001)

[6] Tuten, P.M.: A Model for the Evaluation of IS/IT investments. dissertation (2009)

[7] Lomerson, W.L., Tuten, P.M.: Examining Evaluation Across the IT Value Chain. In: Proceedings of the 2005 Southern Association of Information Systems Conference, USA, pp. 124–129 (2005)
[8] Finding, K.: Measuring IT Performance. IT Alignment in Higher Education 3, 103–114 (2004)
[9] Remenyi, D., Money, A., Sherwood-Smith, M.: The effective measurement and management of IT costs and benefits. Butterworth-Heinemann, Oxford (2000)
[10] IT Governance Institute, http://www.isaca.org/Knowledge-Center/cobit/Pages/Downloads.aspx
[11] Lagsten, J., Goldkuhl, G.: Interpretative IS Evaluation: Results and Uses. Electronic Journal of Information Systems Evaluation 11(2), 97–108 (2008)
[12] Darke, P., Shanks, G., Broadbent, M.: Successfully completing case study research: combining rigour, relevance and pragmatism. Information Systems Journal 8(4), 273–289 (1998)
[13] Yin, R.K.: Case Study Research – Design and Methods. SAGE Publications, Thousand Oaks (2009)
[14] Zainal, Z.: Case study as a research method. Jurnal Kemanusiaan 9, 1–6 (2007)
[15] Shewhart, W.A.: Economic control of quality of manufactured product (1931)
[16] Basili, V.: The Experience Factory and its Relationship to other Quality Approaches. In: Advances in Computers, vol. 41. Academic Press, London (1995)
[17] Seddon, P.B., Graeser, V., Willcocks, L.P.: Measuring Organizational IS Effectiveness: An Overview and Update of Senior Management Perspectives. The DATA BASE for Advances in Information Systems 33(2), 11–28 (2002)
[18] Andra, S.: Action-Oriented Metrics for IT Performance Management. Cutter IT Journal 19(4), 17–21 (2006)
[19] Simonsson, M., Johnson, P.: Assessment of IT Governance – A prioritization of Cobit. In: Proceedings of the Conference on Systems Engineering Research, Los Angeles, USA (2006)
[20] Neely, A.: Performance measurement revolution: why now and what next? International Journal of Operations & Productions Management 19(2), 205–228 (1999)
[21] Wettstein, T., Kueng, P.: A Maturity Model For Performance Measurement Systems. Management Information Systems, 113–122 (2002)
[22] Son, S., Weitzel, T., Laurent, F.: Designing a Process-Oriented Framework for IT Performance Management Systems. The Electronic Journal Information Systems Evaluation 8(3), 219–228 (2005)
[23] Jack, A.T.: ValueMapping-A second generation performance measurement and performance management solution. In: Performance Measurement Association, Boston (2002)
[24] Martinsons, M., Davison, R., Tse, D.: The balanced scorecard: a foundation for the strategic management of information systems. Decision Support Systems 25(1), 71–88 (1999)
[25] Kaplan, R.S., Norton, D.P.: The balanced scorecard – measures that drive performance. Harvard Business Review, 71–79 (1992)
[26] ITPMG: IT Performance Management – What is your IT organization's readiness? IT Performance Management Group. Connecticut (2006)
[27] Keynote, http://www.keynote.com/docs/whitepapers/ScoreHighBusScore_MonitorwhatMatters.pdf

[28] Globerson, S.: Issues in developing a performance criteria system for an organization. International Journal of Production Research 23(4), 639–646 (1985)
[29] Eccles, R.G.: The Performance Measurement Manifesto. Harvard Business Review, 131–137 (January-February 1991)
[30] Swager, J.: Shared Service Center Charges – Een onderzoek naar het verantwoordelijkheidsgevoel van de business voor Shared Service Center Charge. Handboek Management Accounting, 1425–1425-33 (2008)
[31] Verner, J.M., Toraskar, K.V., Brown, R.: Information systems chargeout: a review of current approaches and future challenges. Journal of Information Technology 11(2), 101–117 (1996)
[32] Strikwerda, J.: Shared Service Centers - Van kostenbesparing naar waardecreatie. Van Gorcum/SMS, Den Haag (2003)
[33] Carlsson, E., Schurmman, A.: Management control in Shared Service Centers – How to influence people in the striving towards organizational goals. Linköpings University, Zweden (2004)
[34] Bhimani, A., Horngren, C.T., Datar, S.M., Foster, G.: Management and cost accounting. Pearson Education Limited, Essex (2008)
[35] Paulk, M.C., Curtis, B., Chrissis, M.B., Weber, C.V.: Capability Maturity Model, Version 1.1. IEEE Software 10(4), 18–27 (1993)
[36] Dymond, K.M.: A Guide to the CMM. Process Inc., US, Maryland (1997)
[37] Burns, A., Bush, R.: Principes van marktonderzoek, Dutch edn. Pearson Education Benelux, Amsterdam (2006)
[38] van de Weerd, I., Brinkkemper, S.: Meta-modeling for situational analysis and design methods. In: Syed, M.R., Syed, S.N. (eds.) Handbook of Research on Modern Systems Analysis and Design Technologies and Applications, pp. 38–58. Idea Group Publishing, Hershey (2008)

Enterprise Coherence Assessment

Roel Wagter[1,3], H.A. (Erik) Proper[2,3], and Dirk Witte[1]

[1] Ordina, Nieuwegein, The Netherlands
[2] CRP Henri Tudor, Luxembourg
[3] Radboud University Nijmegen, Nijmegen, The Netherlands
roel.wagter@ordina.nl, erik.proper@tudor.lu,
dirk.witte@ordina.nl

Abstract. In this paper, the authors discuss an instrument for the assessment of enterprise coherence (Enterprise Coherence Assessment, ECA). The term 'coherence' is used rather than the more common term 'alignment', since the latter is generally associated with bringing two concepts in line (typically 'Business' and 'IT'). The word coherence, however, stresses the need to go beyond this. Coherence involves connections and synchronisation between *all* important aspects of an enterprise. 'IT' and 'Business' just being two of these aspects.

The ECA instrument was developed as part of the ongoing GEA (General Enterprise Architecting) research programme, and has so far been applied in seven large Dutch organisations. The paper discusses the context in which the ECA instrument was developed, the instrument itself, as well as the results of the assessment study in which the instrument was applied.

Keywords: business-IT alignment, enterprise coherence assessment, enterprise architecture.

1 Introduction

Developments such as globalisation, the fusion of business and IT, the introduction of new technologies, novel business models, et cetera, pose many challenges to modern day enterprises [14]. As a result, enterprises need to cope with a rapidly changing environment, which means they need the ability to transform themselves (at least) as quickly as their environment does. Such enterprise transformations may range from changes in value propositions and business processes, via changes to the information systems used to support the business processes, to changes of the underlying IT infrastructures. They may be the result of a top-down (strategy driven) desire to change, but they can also consist of numerous bottom-up changes as a result of locally needed changes. Finally, the required/desired transformations will typically touch upon several additional aspects of the enterprise, such as human resourcing, finance, organisational structures, reporting structures, et cetera.

To make large enterprise transformations feasible and manageable, they are typically split into programmes and eventually into projects. Even more, larger enterprises typically do not just have one transformation programme but multiple, that all need to be coordinated with the enterprise's strategy. Therefore, a coordination mechanism is needed that connects the strategic considerations at the strategy level to the execution

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 28–52, 2011.

of the different projects involved in the transformation as a whole. This coordination generally also requires a further elaboration of the enterprise's strategy, since these tend to be too unspecific to indeed steer the programmes and projects within the transformation [14]. In addition, the needed coordination mechanism must allow the coherence between the different aspects of an enterprise to be guarded across the programmes and projects transforming the enterprise [14,30]. Traditionally, project management and programme management are put forward as being responsible for these coordination tasks [17,16]. However, these approaches focus primarily on the management of typical project parameters such as budgets, resource use, deadlines, et cetera. When indeed only considering the typical project parameters, one runs the risk of conducting "local optimisations" at the level of specific projects. For example, when making design decisions that have an impact, which transcends a specific project, projects are likely to aim for solutions that provide the best cost/benefits trade-off within the scope of that specific project, while not looking at the overall picture. Regretfully, however, in practice such local optimisations do not just remain a potential risk. The risk actually materializes, and consequently damages the overall quality of the result of the transformation [14]. This type of risk generally occurs when stakes with regards to general infrastructural elements of an enterprise collide with local short-term interests. This especially endangers the needed coherence/alignment between different aspects within an enterprise (such as business and IT, but also human resources, physical infrastructures, et cetera). As a result, more often than not, enterprises fail to actually realise the desired transformation even though it might be the case that all projects are finished on time and within budget.

Slot [22] has shown that a correlation exists between the performance of IT projects and the use of architecture to steer/coordinate these projects, i.e. projects being implemented *under architecture*. IT projects implemented under architecture result in 19% less budget over-runs. In principle, one might expect that such a positive effect would be discernable when working *under architecture* would be applied to enterprise transformations as a whole as well. Regretfully, however, in various assignments in practice[1], we have been confronted with the situation that transformation projects fail due to budget overruns, or a failure to meet objectives and expectations. Table 1, provides examples of issues and causes, which we have (informally) recorded during our own practical work in several organisations.

Our informal experiences and observations are also supported by the (Dutch) General Court of Auditors [3], who has produced a report on the cause of failures in ICT projects. The lack of enterprise coherence between several aspects is identified as a key cause in the failure of ICT projects (quotes translated from Dutch):

> "*ICT projects for the government seem to be much more expensive than anticipated initially, require more time than planned to complete, or do not deliver the desired results. This is a serious matter, since ICT projects of the government mostly involve the spending of public money. Furthermore the effects of projects that fail, to a larger or lesser extent, are often large-scale projects with profound social impact.*

[1] The authors either currently work for a consultancy firm, or have worked for one in the past.

Table 1. Informal observations

Issues	Causes
"*In this case, the same wheel has been discovered in several places!*"	There is no common shared view or approach to the organisation's design and layout. Solutions are problem specific and implemented locally. Connections with other questions arising from the same (sort of) problems are not made.
"*Our programmes and projects do not fit well with our strategy*"	Irrespective of how clearly the strategy is formulated and how widely it is accepted throughout the organisation, issues are not solved from the organisations mission, vision, core values and objectives, which creates a gap between what we are doing and what we aspire to achieve.
"*Strategic decisions are not followed up!*"	Decisions are not presented as sufficiently convincing to lower management levels, which offers too much room for individual interpretation – and personal interest.
"*We do not have sufficient grip on the progress of our change process.*"	Interdependencies between correlated paths are not properly identified thereby causing unnecessary delays.
"*Our decision-making process is slow and inconsistent!*"	Management lacks the overall picture and can't properly gauge the consequences of their decisions, creating delays as a result of indecision. Furthermore, the absence of an overview increases the risk of decisions not being in line with each other.
"*We are not innovative enough in solving our problems.*"	Not enough attention is devoted to the idea-forming process, which means that the full potential of new possibilities is not being used.
"*Our solution doesn't cover the issue.*"	The issue is not properly analysed and is thus approached from too few perspectives, which don't include the bottlenecks.
"*Over and over again there are endless discussions about the value and necessity of the decisions that have been taken*"	Management does not unanimously support the decisions, because they have been taken unilaterally or are insufficiently substantiated. Opposition in the organisation uses these divisions and uses its influence to delay progress.

> *The most important cause of the (partial) failure of ICT projects revealed by the first part of the research was that ICT projects for the government are often overly ambitious and too complex because of the combination of politics, organisational and technical factors. With these overly complex projects there is no balance between ambition, available people, resources and time.*"

In Op't Land et al. [14], the authors also provide a summary of possible causes for failures of strategic initiatives, as well as the need to develop a solution for them: "*The road from strategy formulation to strategy execution, including the use of programmatic steering, is certainly not an easy one to travel. Research shows that less than 60% of*

the strategic objectives in organisations are reached [23]. When considering the possible failures in strategy execution . . . an instrument is needed to support this process". In [8,9] Hoogervorst also argues in favour of using enterprise architecture as a means to govern coherence in enterprises.

Our own experiences, and the above discussed general insights, seem to indicate that maintaining/achieving coherence (by means of architecture) between different aspects of an enterprise is a crucial factor with regards to change processes, and therefore at least warrants a closer study of causes and potential solutions. The general concept of coherence is described in the MacMillan English dictionary [13] as: "*in which all the different parts fit together in a sensible or pleasing way*", while the Van Dale [26] dictionary describes coherence as: "*the extend in which several aspects are connected*". In line with these definitions, we define *enterprise coherence* as follows:

> *Enterprise coherence is the extend to which all relevants aspects of an enterprise are connected, to the extend necessary to let the enterprise meet its desired results.*

Since achieving, and/or maintaining enterprise coherence, seems to be an important issue (i.e. there is a potential positive correlation with the performance of an organisation), there is also reason to explicitly govern *enterprise coherence*. This insight triggered the multi-client *General Enterprise Architecting* (GEA) research programme [29][2]. The aim of this programme was to answer on the following research questions: *What factors influence/define enterprise coherence? How to govern coherence and improve the performance of an organisation?* The results of the first iterations of this research programme have been reported in [28][3]. Failure to adopt a holistic approach to key business issues, i.e. the frequent unilateral approach from a IT oriented angle, has been an important trigger for the research program GEA.

A fundamental first step in the GEA programme was the development of the *Enterprise Coherence Assessment* (ECA) to attain a clearer understanding of the challenges to *enterprise coherence* and its associated *governance of coherence* [28], as well as the impact of *enterprise coherence* on organizational performance.

The remainder of this paper is structured as follows. Section 2 positions how current approaches suggest how to govern enterprise coherence and contrasts this with the approach taken in the GEA programme. Section 3 then provides a discussion on the research context of this paper, in terms of the driving research question, and the research methodology used. In Section 4 we then continue with the presentation of the current

[2] During different stages of the GEA research programme, the members of the programme included: ABN AMRO; ANWB; Achmea; Belastingdienst – Centrum voor ICT ICTU; ING; Kappa Holding; Ministerie van Binnenlandse Zaken en Koninkrijksrelaties; Ministerie van Defensie; Ministerie van Justitie – Dienst Justitiële Inrichtingen; Ministerie van LNV – Dienst Regelingen; Ministerie van Landbouw, Natuur en Voedselkwaliteit; Nederlandse Spoorwegen; Ordina; PGGM; Politie Nederland; Prorail; Provincie Flevoland; Rabobank; Radboud University Nijmegen; Rijkswaterstaat; UWV; Wehkamp.

[3] For strategic reason, the initial target of the results was the Dutch language community, as most participating organisations where also based in the Dutch language area, while also having a national/local focus. In the near future, these initial results will be made available in English as well.

version of the ECA instrument. Section 5 continues with a report on the application of the instrument in the context of seven large Dutch organisations.

2 Governing Enterprise Coherence

As argued in [14,28], architecture offers a means for management to obtain insight, and to make decisions about the direction of enterprise transformations. As such, it should act as a means to steer enterprise transformations, while in particular enable senior management to govern coherence. We regard *enterprise architecture* as the appropriate means to make enterprise coherence explicit, as well as controllable or at least influenceable.

Effective governance of enterprise coherence requires an active involvement of senior management. This, however, implies two important requirements:

1. It is necessary to take the concerns, and associated strategic dialogues, of senior management as a starting point. In other words, the way in which architecture is integrated into the strategic dialogue should take the concerns, language, and style of communication of senior management as a starting point.

   When not doing so, it will be difficult to really involve senior management. Even more, the strategic dialogues provide the starting point for steering enterprise transformations and to guard coherence.
2. The power structures, be they of political, informal, or cultural nature, within an enterprise should be a leading element in governing enterprise coherence.

   As discussed in the introduction, an important reason for using architecture to steer and coordinate enterprise transformations is the fact that those design decisions which, in principle, transcend the interests of a specific project can be guarded/enforced that way. Doing so, however, also requires a strong commitment from senior management to these design decisions transcending projects. “Local business stakeholders”, such as business unit managers, who have a direct interest in the outcome of a project, may want to take projects a different direction (more favorable to their own interest) than would be desirable from an enterprise-wide perspective. Such divergent forces are also likely to lead to erosion of the desired enterprise coherence.

We argue that existing approaches and frameworks, such as, Zachman [24], DYA [30], Abcouwer [4], Henderson & Venkatraman[7], TOGAF [25], IAF [27], ArchiMate [11,10], take an “engineering oriented” style of communicating with senior management and stakeholders in general. The architecture frameworks underlying each of these approaches are very much driven by “engineering principles”, and as such correspond to a Blue-print style of thinking about change [5].

The above requirements, however, suggest the use of another style of thinking. In terms of stakeholder interests, formal and informal power structures within enterprises, and the associated processes of creating win-win situations and forming coalitions. In terms of De Caluwé [5], this is more the Yellow-print style of thinking about change. In the GEA programme, this line of thinking was taken as a starting point, by taking

the perspective that the actual political power structures/domains, and associated strategic dialogues, within an enterprise should be taken as a starting point, rather than the aspect/perspective frameworks suggested by existing architecture approaches.

In future research, we intent to position governing coherence in relation to the *Green*, *Red* and *White* "colors" as well. This does not imply that the existing Blue-print style frameworks and approaches are not useful. On the contrary, the engineering perspective is much needed. At the same time, it needs to be embedded in a Yellow-print oriented process. Architecture models produced from an engineering perspective potentially provide thorough underpinning of the views, sketches and models uses/created in the strategic dialogues with senior management. However, rather than structuring the models and views in terms of "information architecture", "application architecture" and "infrastructure", they would have to be structured based on those domains that are meaningful within the strategic and political dialogue in an enterprise. For example, in terms of "human resourcing", "clients", "regulators", "culture", "intellectual property", "suppliers", et cetera. Needless to say that this is also highly organisation specific.

This leads to the situation as suggested in Figure 1, where we find on the left hand side the Blue-print style of thinking and associated frameworks, and on the right hand side the Yellow-print oriented approach. Note the (tentative) position of the Zachman framework. More so than frameworks such as IAF, ArchiMate or TOGAF's content framework, the Zachman framework clearly suggest to tune the models and views to the interests/concerns of the stakeholders, and even suggests a classification of stakeholders. In our view, however, it still does so from a Blue-print thinking perspective and certainly does not take the power structures in an organisation into account.

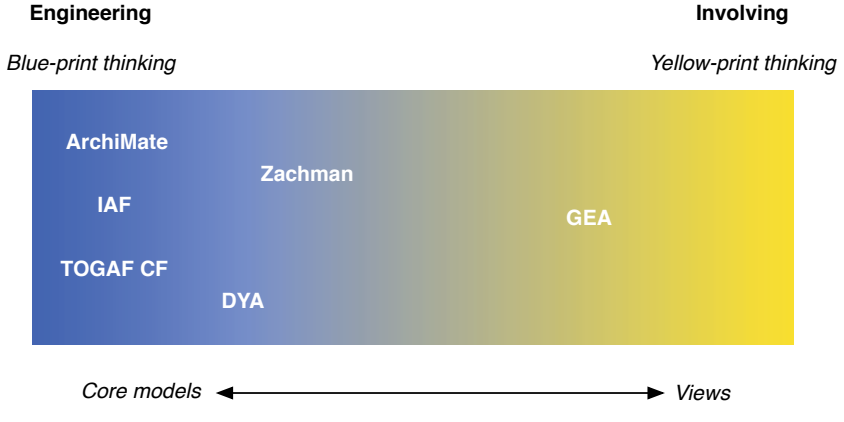


Fig. 1. Bridging Blue-print thinking to Yellow-print thinking

3 Research Context

As mentioned before, the development of the ECA is part of the ongoing GEA research programme. In this section we provide more background to this research programme, as well as the research method used in developing the ECA.

The GEA programme [29] is driven by four key research questions:

1. What are the core factors that influence/define enterprise coherence?
2. What is (in practice!) its impact on the performance of an enterprise?
3. How can an enterprise's level of coherence be expressed explicitly?
4. How can 'enterprise coherence' be governed?

More specifically, the research objectives of the GEA programme are:

1. Define the core indicators and factors defining/influencing enterprise coherence.
2. Identify the impact of enterprise coherence on an enterprise's performance.
3. Develop an instrument to assess an enterprise's level of coherence.
4. Develop instruments to guard/improve the level of coherence in enterprises during transformations.

The Enterprise Coherence Assessment (ECA) was developed to gain initial insight into the first two questions. On the one hand the answer to these questions determine if it is appropriate to carry out further research into the governance of enterprise coherence, while on the other hand providing a first refined definition of enterprise coherence and its practical impact on organisational performance.

The GEA programme took the perspective that if more than 50% of the organisations involved in the first ECA studies lack effective governance coherence, it was safe to assume that "*the lack of coherence governance in enterprise*" is indeed a relevant issue that needs further study as well as the development of a set of instruments (and underlying theory) supporting enterprises in governing coherence. The initial application of ECA involved seven large Dutch organisations (members of the GEA programme).

At the start of the GEA programme, the plan was to execute ECA assessment (for each of the participating organisations) at three stages:

1. A first assessment at the start of the programme, providing a baseline.
2. A second assessment once a shared understanding of enterprise coherence was reached. By comparing the results to the baseline, the effect of having an shared awareness of the forces that influence coherence should be measurable.
3. A final assessment once proper governance of enterprise coherence was put in place. By comparing these final assessment results to the earlier ones, the additional effect of coherence governance could be made explicit.

However, soon it became clear that doing these three assessments was not feasible. In the time needed for such longitudinal assessments, the composition of the involved organisations, as well as the people involved, would change so much that the results would no longer be comparable. We have therefore modified this idea to only implement the first assessment in the form of the ECA assessment instrument, while using a case-based research methodology [31] to further evolve the instrument. See Figure 2 and Figure 3.

The ECA that has been carried at the start of the GEA programme convinced the participants that there was enough evidence warranting the development of effective instruments to govern enterprise coherence. The resulting set of instruments, based on multiple additional case studies, is called GEA (General Enterprise Architecting) [28].

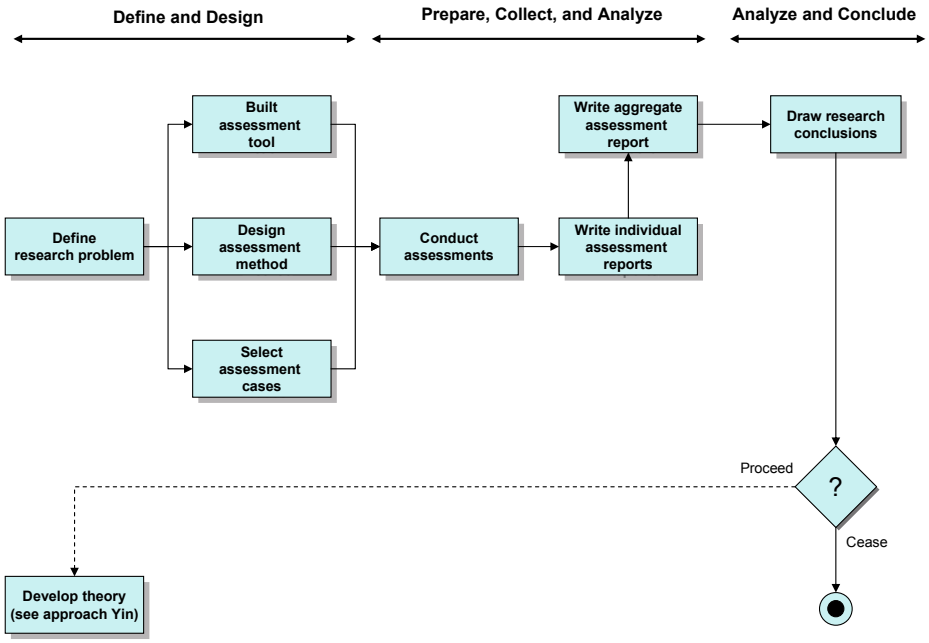


Fig. 2. Preliminary research approach for the development of the ECA, based on Yin [31]

4 The Enterprise Coherence Assessment Instrument

As a first step in the development of the ECA, a series of Metaplan [21] sessions was organized involving experts from eighteen organisations involved in the GEA platform [1]. The aim of these sessions was to gather an inventory of established characteristics for the success of coherence governance, from the perspective of experts from the field. The identified set of characteristics are shown in Table 2 and Table 3. As an additional source of input for the creation of the inventory, we also used characteristics of the Architecture Maturity Model embedded in the Dynamic Enterprise Architecture (DYA) [30] method.

The ECA instrument was not designed to carry out large-scale surveys in which all current rules in the field of statistics apply. ECA is specifically designed to make differences in the opinions of respondents of an organization explicit. This provides an explicit indication of the degree of governing coherence, while also providing a base to achieve a shared understanding of this level of coherence, and actions needed to improve it. At the same time, however, the ECA instrument has been designed in such a way that the results remain comparable across organizations. To reduce the variance that may result from different interpretations by the respondents, all respondents will be taken (by the interviewer) through a *joint* discussion of the questions and their further explanations (see Appendix A).

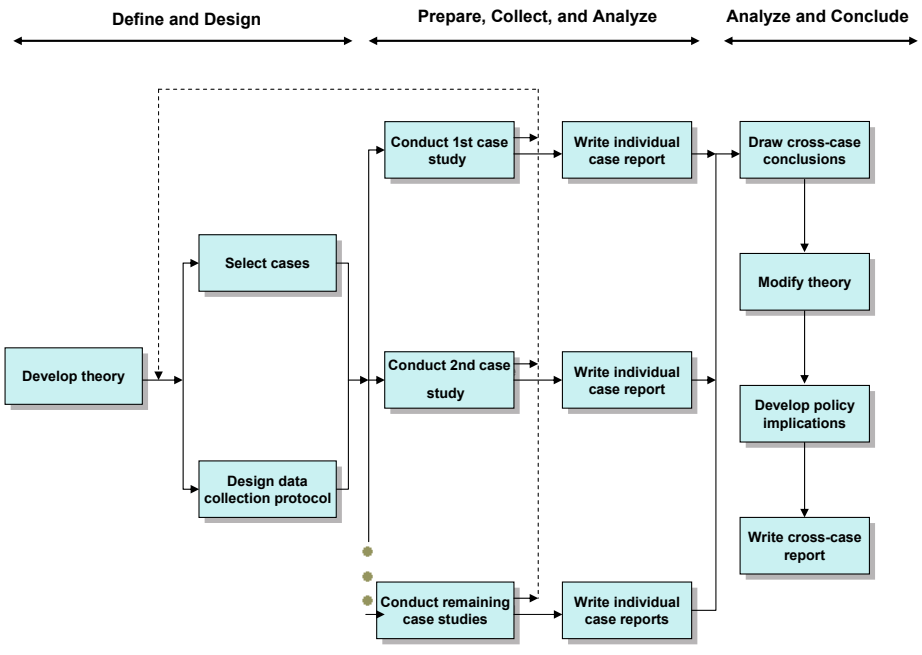


Fig. 3. Research approach adopted from Yin [31]

In future iterations of the ECA instrument, we intend to also include characteristics from additional sources, including the IT Architecture Capability Maturity Model [6], the Normalized Architecture Organisation Maturity Index (NAOMI) [18], the Enterprise Architecture Score Card [20] and the NASCIO Enterprise Architecture Maturity Model [2].

In the GEA programme, the inventory of characteristics led to the decision to develop the following parts of GEA: the EA-vision, EA-government, EA-processes, EA-products, EA-people and EA-means. The core of the ECA comprises of twelve key questions and their connections to these GEA parts. The resulting twelve questions are divided into two blocks of six questions each. The first block of six questions addresses the level at which an enterprise has developed a vision on the governance of its coherence. The second block of six questions concerns the extent of the application of the vision to the enterprise architecture practices.

The resulting set of questions are listed in the example questionnaire shown in Figure 4. The extent to which an organisation answers 'yes' to the questions determines its score. Before answering the questions, a process is carried out whereby the questions are weighted by the respondents. Some respondent can of course assign a higher importance to a specific question than another. Below we will return to the role of this weighting process.

To ensure that the assessment questions are answered as objectively as possible, the interviewer who is responsible for the collection of the answers is provided with a reference frame (see Appendix A). As mentioned before, before the respondents are

Table 2. Characteristics for success on the level of the development of an EA vision

Characteristic	*Description*
E.A. Vision	In order to be able to prove the value of EA one pre-requisite is that the top of the organisation holds a vision on EA.
Added value	The added value of EA as a strategic control tool should be recognised and promoted by all parties concerned. Also the added value of EA compared with other control tools that are in use.
Integral	To establish the EA function an integral approach to vision development, architecture processes and products, and the people and resources needed for EA is necessary.
Open	EA is an open model, managers control the number and the name of EA perspectives and the related components.
Customer orientation	The EA processes and products should support the control processes in a tailor made way, while supplying the results supporting these control processes.
Scope	There are never many principles. Its limitations illustrate the strength of the EA management tool because it means decisions can be made quickly. Therefore EA moves at a strategic level and gives direction to tactical and operational levels by means of frameworks.
Product distinction	From the point of accessibility and understanding it is necessary to distinguish between EA management products and EA specialist products. This means that it is possible to communicate with the right target groups and with the right EA products.

Table 3. Characteristics for success on the level of the application of the EA vision

Characteristic	*Description*
Allocating resources	Management must provide people with the necessary competencies, time, budget and resources for EA to realise the added value of EA.
Participation	Enterprise architects must possess access to managers and participate in the organisation's control processes
Directional	The EA management products require approval and control by the managers and provide direction to change programmes and the existing organisation.
Coherence	All business perspectives must be brought together coherently by the responsible managers.
Permanence	EA must be arranged as a continuous process whereby coherence is permanently adjusted to the dynamics of the internal and external environment.
Event driven	EA must be used as a management tool at the moment when major company issues arise in order to establish timely integral solutions and approaches.

asked to answer the questions, the interviewer will *jointly* take them through the list of questions and the associated reference frame. This frame ensures that the answers of all respondents are 'calibrated'. The relationship between the questions and the GEA parts

	Questions ECA instrument	Yes	No
1	We possess an EA vision agreed by the management.	○	◉
2	Our EA vision is the result of cooperation between the representatives of all stakeholders.	○	◉
3	Our organisation's vision, objectives and strategy are characterised by the various EA elements as perspectives, key concepts, guiding statements, principles, etc.	○	◉
4	Our EA vision is developed into EA processes, products, people and resources.	◉	○
5	In our organisation one or more control tools are used to rate organisational results in coherence.	○	◉
6	In our organisation one or more control tools are used to control change processes by coherence.	◉	○
7	Our EA architects are involved in setting up control processes at a strategic and tactical level.	○	◉
8	It is known whether all our change programmes were developed with or without 'EA'.	◉	○
9	In our managers' competence profile 'EA' is included as a competence.	○	◉
10	Our managers understand and use EA products in their control processes.	◉	○
11	At least once a year there is an updated version of the content of our EA framework.	◉	○
12	Those with end-responsibility for our change processes are accountable for time, money and quality as well as meeting EA principles and guidelines.	◉	○

Fig. 4. Example of a completed ECA questionnaire

are given in the table shown in Table 4. The numbers correspond to the twelve ECA questions in Figure 4.

The results of an ECA are reflected in a quadrant model, as depicted in Figure 5. This model is composed of two axes, the horizontal axis represents the level of development of the EA Vision and the vertical axis represents the level of the application of the EA Vision. These axes represent two dimensions of the governance of enterprise coherence, which correspond to the aforementioned GEA parts that need to be developed.

The axis 'EA vision development' describes the extent to which an organisation's body of knowledge concerning the governance of enterprise coherence has been made explicit. *Is there a vision about enterprise architecting? Has the vision been translated into a methodology and how an organisation wants to use it (is there an implementation plan)? Is there a real ambition for the application of EA?* The axis 'EA vision application' describes the extent to which an organisation actually operates the body of thought.

The correlation between the two axes results in four quadrants. Figure 6 provides a brief outline of the characteristics per quadrant. Below we will discuss the quadrants in more detail, while Section 5 provides anonymized real world examples of organisations and their positioning in relation to the quadrants.

Degenerating quadrant – If an organisation has no vision about enterprise architecting and also does not know how to apply this form of management then the organisation scores in this quadrant. Coherence in the organisation will continue to deteriorate with proportionate effects on the organisation's performance.

Table 4. Mapping GEA parts to the questions

GEA parts	Question
EA-Vision EA-Governance	1 – 6
EA-Processes EA-Products EA-People EA-Means	7 – 12

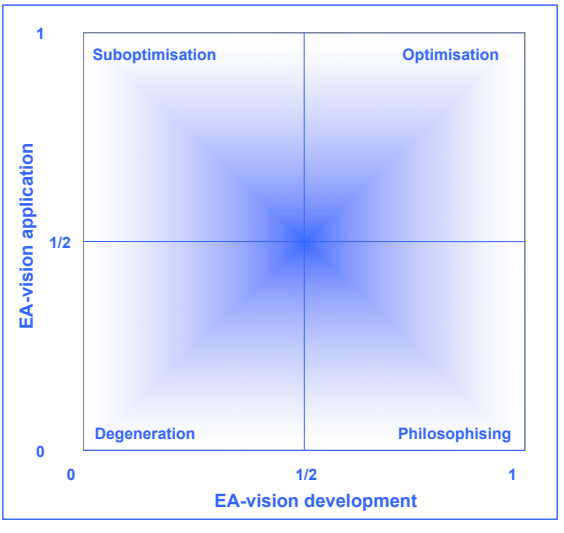


Fig. 5. Effects of ECA on the organisation

Characteristic aspects for this quadrant are:

- Coherence is not considered an important aspect.
- There is no synchronisation between representatives of the important aspects of the enterprise.
- No EA vision or activities.
- Strategy is not supported by EA.
- There is no awareness of EA.
- No people or resources are allocated to EA.
- Solutions are implemented without architecture.
- Decrease in effectiveness and efficiency.

Philosophical quadrant – There is a vision of enterprise architecting, this is also translated into how it should be implemented, but it is not developed beyond terms of 'paper' and 'goodwill'. It is not 'exploited', let alone implemented. The vision document seems to have disappearance in the well-known bottom drawer. There may be some basic increase in effectiveness. A basic level/awareness of governance of

enterprise coherence may be developed. Therefore, there is an increased likelihood that things move in 'the right direction'.

Characteristic aspects for this quadrant are:

- Coherence is considered to be a strategic aspect throughout the organisation.
- There is regular synchronisation between representatives of the important aspects of the enterprise.
- There is an integral EA vision, limited EA activities in the enterprise's operations.
- EA is integrated in the organisation's strategy.
- EA is inspired especially by third parties.
- A limited number of people and resources has been allocated to EA.
- Some solutions are implemented with architecture.
- Increase in effectiveness, not in efficiency.

Suboptimal quadrant – Organisations positioned in this quadrant will be organisations with do-ers, with individuals with their own vision and ideas about enterprise architecting, who have taken their own local actions. Models have been designed that perhaps offer the most potential for reinforcing governance of coherence throughout the organisation. However, these are not synchronized/aligned and are formulated in their own jargon. The biggest flaw is that the managers, who should be the customers of these products, do not know that they exist or they do not know how to include them in management processes. The application of EA is the next stage but not on an enterprise level. A number of things are done well, but these are not good things by definition. Throughout the organisation there is some increase in efficiency.

Characteristic aspects of this quadrant are:

- Coherence is only experienced as a enterprise aspect locally and in different ways.
- There is no synchronisation between representatives of the important enterprise aspects.
- Local EA visions and activities are on the agenda.
- EA is integrated in one or more department strategies.
- EA is applied particularly by third parties.
- Local and frequent temporary allocation of people and resources to EA.
- Local solutions are implemented with architecture.
- Not effective, increase in efficiency.

Optimisation Quadrant – In this quadrant, vision and action go hand in hand. The organisation has a detailed view of enterprise architecting and knows how to use it to its advantage. The managers take strategic decisions from their integral and current knowledge about the meaning and design of the organisation. The organisation works on optimising management and implementation processes that are supported by EA processes and products. The good things are done well, in other words efficiency and effectiveness go hand in hand.

Characteristic aspects for this quadrant are:

- Coherence is experienced as an important aspect and governance of coherence is applied throughout the organisation.

- There is frequent synchronisation between representatives of the important aspects of the enterprise.
- There is an integral EA vision and activities that as a framework give direction on a strategic, tactical and operational level.
- EA is integrated in the organisation's strategy.
- EA is internalized in the thinking and action of its own leaders and managers.
- There is talk of structural allocation of people and resources.
- Integral solutions for major issues are implemented with architecture.
- Structural improvements in coherence within the organisation is on the agenda.
- There is high effectiveness and efficiency.

When the questions from the questionnaire have been answered, then the respondents' scores offer a good starting point for follow up actions to improve the governance of enterprise coherence. In particular, by using the following questions as drivers:

- How can the (possible) differences in the positioning of the respondents be explained?
- Which steps for improvement can be made in connection with the positioning at an organisational level (average of the respondents' scores)?

The discussion arising from the first question may lead to the employee adjusting their views, which would have provided a very different score. Or if not, it may lead to

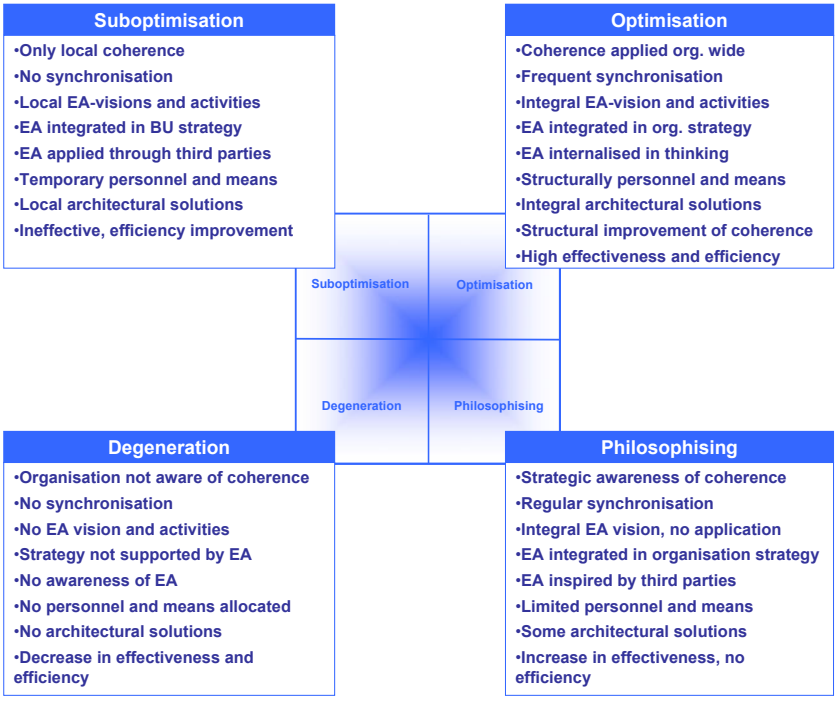


Fig. 6. Characteristics per quadrant

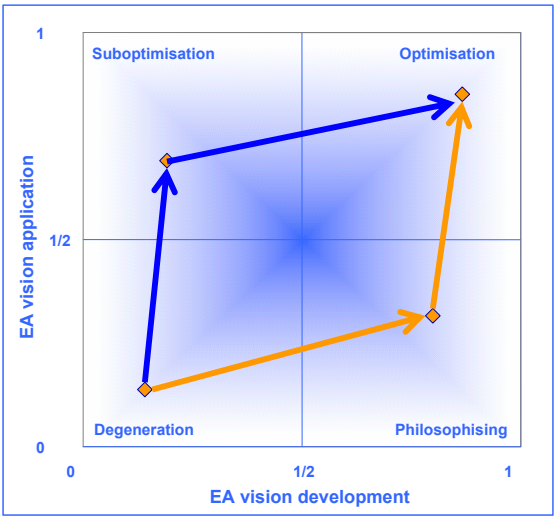


Fig. 7. Development scenarios

new concepts for the whole group. The organisation's score is an average of the given scores from the individual respondents. However, as we will see in the next Section, the average is not just computed, but rather determined in a joined session with all the involved respondents. During such a session, individual respondents may change their scores in response to improved insights into their understanding of the actual situation in the organisation and/or insight into the question itself.

If the results of the organisation's score are in the optimisation quadrant then people will be reap the rewards of applying coherence governance. It is important to maintain this optimisation and to stay alert so as not to fall back into old habits. If the positioning falls in one of the three following quadrants: degenerative, philosophical or sub-optimisation, then this offers greater possibilities for improvement. If the score falls in the degenerative quadrant this means that one must first take a step to the right as well as directly upwards, before the step can be made towards optimisation (see Figure 7). These approaches correspond to organisation's management styles. One organisation first wants to consider it properly, as a supporter of the Design School and the other organisation wants to first initiate experiments, as a supporter of the Learning School [15].

5 Using the ECA Instrument

In this Section we are concerned with the use of the ECA instrument in practice. We start with a discussion of the steps involved in applying the instrument, followed by the discussion of the application of the ECA instrument in seven large organisations in the Netherlands.

The ECA instrument uses the following steps to position an enterprise:

1. Determine the relative weight of the questions: "*rank the questions in order of importance*".
2. Gather responses to situational questions: "*which questions do, or do not, apply to your organisation?*"
3. Process the answers and feedback of the positioning per respondent.
4. Analyze the differences between the individual positions.
5. Aggregate the individual positions to determine the organisation's positioning.

Step 1: *Determine the weight of the questions*

As a first step, for each question the respondent indicates its relative importance to the organisation. See Figure 8. A question's importance is determined according to the situation, no two organisations are the same. Where, for example in a more hierarchal organisation it is an absolute must for the management to have agreed a vision on enterprise architecting, in a different organisation it may be much more important for all the stakeholders to be involved in formulating the vision.

The weighting of the questions is conducted using the "Pair Wise Comparison" method [19]. Pair Wise Comparison ranks the twelve questions by pairing them by comparison. By bundling the results of this weighting it is subsequently possible to divide the research population into segments, who have the same standpoints regarding the questions.

The actual comparison was computed using Analytic Hierarchy Process (AHP) [19]. In the ECA case, two dimensions and twelve questions are used. These are respectively (D)evelopment and o(P)eration, resulting in two times six questions (D1 to D6 and P1 to P6). The questions take the form of a statement that does (factor 1) or does not (factor 0) apply to an organisation. The weighted averages for D and P are calculated from the weighted average of weight × factor (1 or 0), for each statement. The weights are determined by comparing the questions, separately for D and P. D1 is compared with D2 to D6, then D2 with D3 up to D6, and so on. Then we ask the question: is

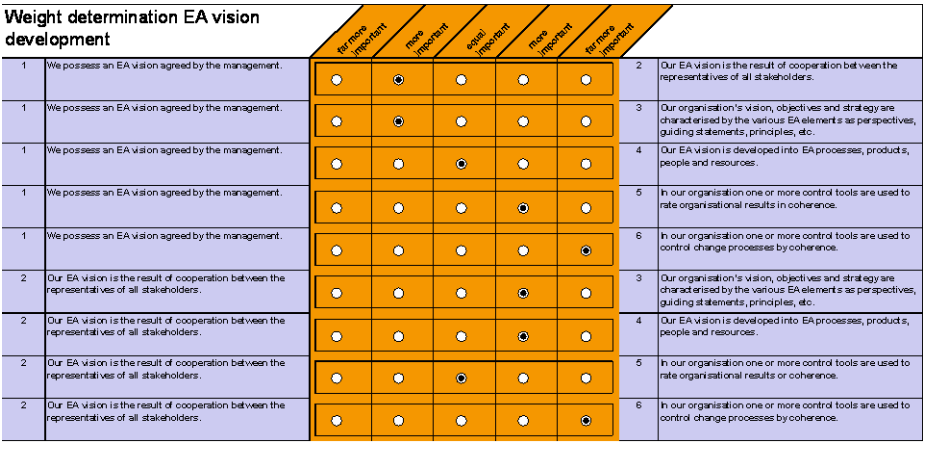


Fig. 8. Part of the process to determine the weight of the questions [19]

D1 much more important than D2, if so, then D1 4 and D2 are awarded a $1/4$ point, if more important than 2 respectively $1/2$ and if equally important each 1 point. The total number of points per statement determines the weight.

Step 2: *Answer situational questions*

After determining the weight of the twelve questions, the form with the situational questions is completed with yes/no. See Figure 4.

Step 3: *Process and provide feedback and position each respondent*

The details of each respondent are entered into an application that calculates the individual position. See the diagram in Figure 9 for the person in question.

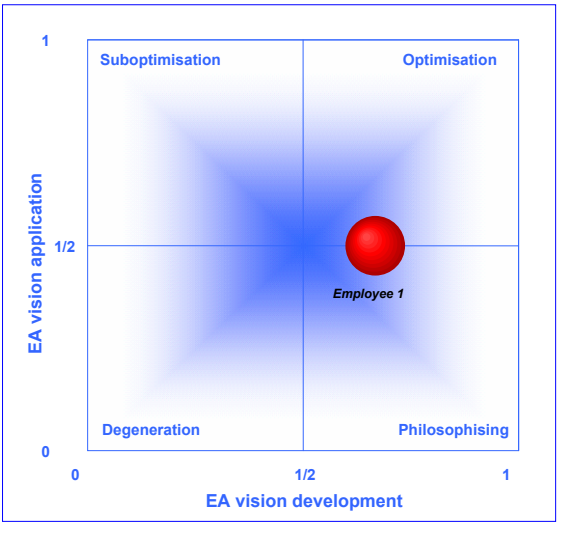


Fig. 9. Diagram of individual positioning

Step 4: *Analyse the differences between individual positions*

Analysis of the differences between the individual positions can be carried out in one of two ways: on the aspect of the weight determination that is known for the questions and on the aspect of the situation. Making these differences explicit can result in interesting discussions and may already lead to adjustments in opinions of individuals or groups of people at this stage of the process. See Figure 10, which illustrates the positioning of individual employees as well as the position of the organisation.

In Figure 10 we illustrate the scores of three employees from one organisation, resulting from the assessment carried out by the participants of the growth platform, as well as the organisation's total score. Significant differences are apparent in the ratings completed by the employees. We can also deduce that the related organisation scores 0.545 on the EA development axis.

Analysis of the responses reveals that there is a vision but it is not developed into an implementation plan. And there is also no ambition to use any tools to strengthen coherence governance. A further analysis of the score 0.241 on the EA vision application

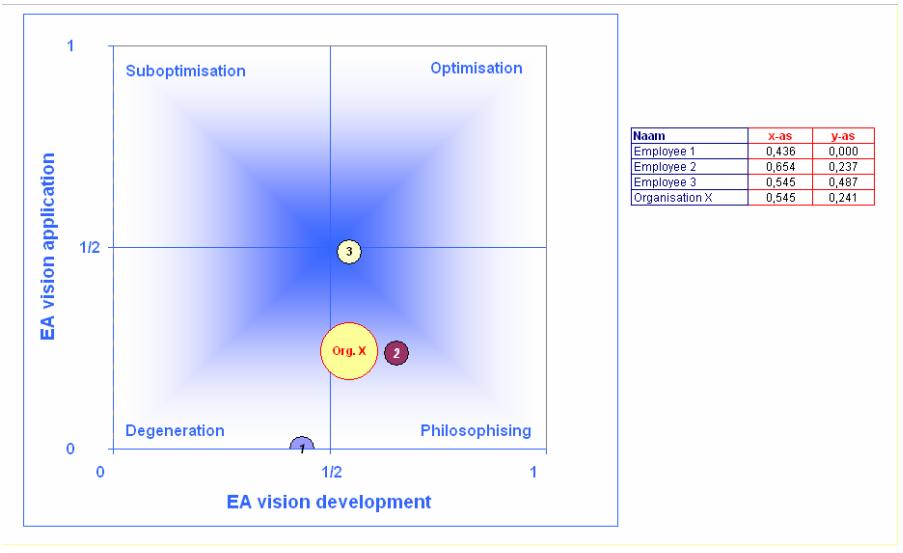


Fig. 10. Individual and organizational positioning

axis reveals that ideas about the application of enterprise architecting are implemented in a fragmented manner.

Step 5: *Aggregate the individual positioning at an organisational level*

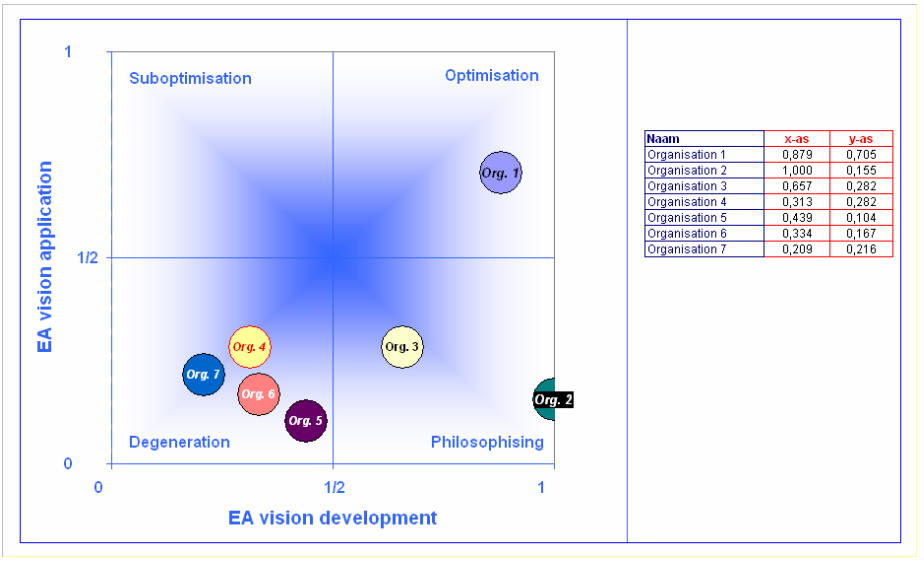
After the relative weighting of the situational questions has been performed and the questions have been answered, the model automatically provides the position of the total organisation in the matrix (see Figure 10). This yields an initial average of the individual scores of the respondents.

These results are then discussed in a joined session with the involved respondents. These discussions may lead to insights about the actual situation in the organisation and/or better interpretation of the questions. This may, on its turn, lead respondents to want to change their individual scores, and eventually the aggregate scores for the organisation. Using this joint discussion, undesired variance due to misinterpretations and/or incomplete knowledge about the organisation, is reduced.

With the help of this positioning the starting points can be identified for a development and implementation strategy for enterprise architecting in the organisation. Figure 6 can be helpful at this stage. Is it necessary, for example, to first develop a vision, translate it into workable concepts and subsequently develop an implementation strategy? Or can one already get to work because sufficient homework has already been done on developing a vision et cetera? In this case it is perhaps necessary to first establish a communication offensive.

In the context of the research question "*how does governance of enterprise coherence work in organisations?*", we applied ECA to seven large organisations in the Netherlands, involving twenty-five participants.

Before the assessment we set the condition that if there was a lack of governance of enterprise coherence in more than 50% of the researched organisations, that the problem 'lack of coherence governance in organisations' is present. If this is proved then



Naam	x-as	y-as
Organisation 1	0,879	0,705
Organisation 2	1,000	0,155
Organisation 3	0,657	0,282
Organisation 4	0,313	0,282
Organisation 5	0,439	0,104
Organisation 6	0,334	0,167
Organisation 7	0,209	0,216

Fig. 11. ECA positioning of several organisations

the conditions are met for further research and the development of a theory. We claim there to be a general lack of governing of enterprise coherence, if less than 50% of the assessed organisations scores in the optimization quadrant.

The results of the assessment are summarized in Figure 11. This provides an initial overview of the situation of participating parties and differences. More specifically, the assessment resulted in the following comments/feedback from the respondents:

– Organisation 1 is characterized by the fact that its vision on enterprise architecture and its management are developed by themselves, while being based on methods used in the market, such as DYA, GEA and TOGAF. Therefore it has been accepted and supported at board level.

  The relationship between the level at which meaning is assigned and the tactical/operational levels of the organisation is also well defined in the enterprise architecture.

  Those who developed the enterprise architecture vision and management have also managed to implement the enterprise architecture processes and allocate people and resources on this basis.

  Given this process orientation all change processes are implemented ‘under architecture’. This is achieved by consistently developing program start architectures (PSAs [30]) that can be used as an effective steering instrument for transition before initiating change process transitions. The fact that the relevant directional frameworks that apply at the level of the organisation at which meaning is assigned are also incorporated in the PSAs bridges the gap between the strategic and tactical levels.

  In short, the coherence of the organisation is made explicit, updated on an ongoing basis and used to develop integral solution options and approach choices for

major issues. This ensures that the coherence and therefore the performance of the organisation are continuously improved.

- Organisation 2 is a large Dutch government organisation, which consists of several divisions that operate with a high degree of autonomy.

  Within this organisation, the high quality architecture products have been designed at concern level in the form of an extensive enterprise architecture vision, business process models, use cases, etc. However, because the divisions operate with such a high degree of autonomy the enterprise architecture function can only attempt to elicit 'architecture behavior' and has achieved only limited success in this respect. In practice, only a few of the divisions are prepared to work with reference models that describe situations encountered in daily practice.
- Organisation 3 is a large executive agency from the Dutch government, which was created out of a merger of several similar organisations. This background made that the development of an enterprise architecture was not an easy job. There was a strong emphasis on producing an enterprise architecture as a product. One did not succeed in developing architecture processes and embedding these in the merged organisation.

  Therefore, the first large project that was supposed to be implemented 'under architecture' failed miserably. Even tough there were other contributing factors as well, the architecture was held as the primary cause of this failure. The enterprise architecture was written off as unreceptive and the architecture function was largely dismantled.

  Although those involved were actually on the right track with the enterprise architecture vision, etc., they were a long away from implementing and applying it in the organisation.
- Organisation 4 is a large construction organisation in which the enterprise architecture vision is still very limited and 'hidden' in several documents. The situation is also complicated by the fact that people throughout the organisation describe themselves as 'architects', while applying architecture in their own personal way.
- Organisation 5 is a large Dutch government agency with many offices located throughout the country. At a corporate level, the organisation has made considerable progress in articulating their enterprise architecture vision. Those involved have produced ample architectural models that have, however, a strong IT focus. The process orientation in their architectural thinking is also lagging behind. The architecture processes have also not been identified, described and implemented.
- Organisation 6 is a large Dutch transport organisation. The architecture function is this organisation involves a relatively small architecture group. The situation is characterized by the fact that the organisation does not employ separately managed change processes. In other words, all organizational changes are directly implemented by line managers and/or business unit managers.

  Despite the fact that the architects are doing their best to develop an architecture vision and get it supported at board level they have no sway with the managers who implement the changes. The fact that 'working under architecture' sometimes requires investments in the interest of the greater whole is a complicating factor. The line and business unit managers are not prepared to authorise such investments,

partly because of the way in which financial management and accounting are organised.

– Organisation 7 is a large association with several million members. The association organises activities in five different domains. The organisation structure reflects these domains.

  Several attempts have been made to determine the form and content of the architecture function. Due to several external influences, the organisation entered a troublesome time in its existence. Its survival was indeed threatened. Major cost cuts that have been made as a result of this situation, have resulted in the architecture function not being developed further, while architectural initiatives have been reduced considerably.

  Regretfully, the board did not realize that during such cost cutting periods, an enterprise architecture can provide adequate support for effective cost cutting.

One party is clearly on the right track with coherence governance. This involves one of the largest Dutch pension funds that by 'working with architecture on an enterprise level' has already managed to halve its ICT costs within five years [12].

Two organisations score in the philosophical quadrant and four organisations in the degenerative quadrant. This overview of the relative positions have been discussed and validated in a meeting with the involved parties, with the aim to identify actions that will lead to the improvement of their respective governance of enterprise coherence. The situational questions that were indicated as irrelevant, combined with a relatively high importance, form the first indication for measures to reinforce governance of coherence.

Since 85.7% of the organisations do not score in the optimisation quadrant, an important conclusion that can be drawn from this assessment is that it clearly demonstrates the need for further research into the governance of enterprise coherence, in particular the development of a theory for the governance of enterprise coherence.

It is also interesting to note that organisation 5 suffers from similar problems as Organisation 2, in terms of the autonomy of divisions and offices. This resulted in the provisional conclusion that it might be more difficult to implement enterprise architecture in organisations with divisions that operate with a relatively high degree of autonomy than in more centrally managed organisations. We recommend further research on this point.

6 Conclusion

In this article we explored the ECA instrument and a coherence assessment that was carried out in seven large organisations in the Netherlands. This tool provides individual organisations with a simple measure for positioning itself on an EA vision development level as well as its level of application. Situational differences can be taken into account. In particular, respondents can define the relative importance of the questions. Principles, design, procedure and backgrounds to the tool were also discussed. Our research revealed that in a substantial number of the assessed organisations there was a lack of governance of enterprise coherence. The results of the assessment offer organisations the tools to begin discussions, about the use of enterprise architecture as an instrument to achieve better governance of enterprise coherence.

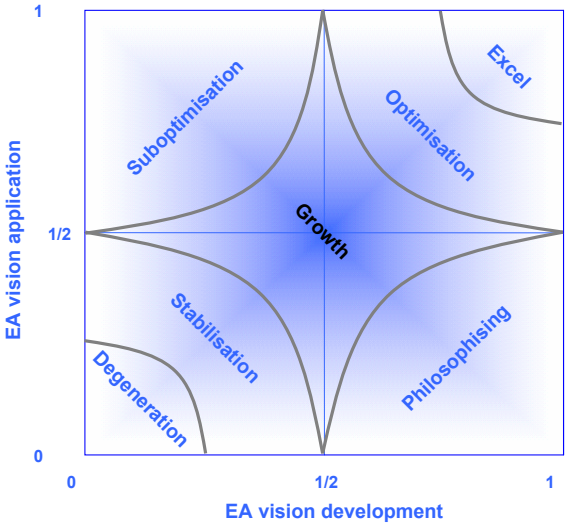


Fig. 12. Main stages of organisational development in GEA positioning

In our subsequent research we will refine the ECA instrument by developing more questions for the enterprise architecting dimensions EA vision, processes, products, people, resources, method and management. We also intend to also include characteristics from additional sources, including IT Architecture Capability Maturity Model [6], the Normalized Architecture Organisation Maturity Index (NAOMI) [18], the Enterprise Architecture Score Card [20] and the NASCIO Enterprise Architecture Maturity Model [2]. In this refined positioning process we will also include the relationship with the organisation's maturity. Figure 12 gives an impression of the desired result. The main diagonal in Figure 12 shows the known main stages of organisational development.

A Elaboration of the ECA Questionnaire

1. *We possess an EA vision agreed by the management.*
   If one participates in enterprise architecture (EA) then we assume that a vision of EA is articulated in a document and subsequently agreed to by management.
   With regards to the content aspect, which are reflected in the vision, we consider:
   - Whether EA is defined in terms of what it is?
   - Why are we doing it?
   - Who does it, how and with what do we do it?
   - What solves it, what are the desired effects, et cetera?
   - Are several management theories included in the vision's principles?
   - Are EA's success factors established?
   - Is there a clear degree of urgency?
2. Our EA vision is the result of cooperation between the representatives of all stakeholders.

One of the EA factors for success involves the situation of whether all (representatives of) important organisational components cooperate in the design.
In your view is this the case?

3. *Our organisation's vision, objectives and strategy are characterised by the various EA elements as perspectives, key concepts, guiding statements, principles, et cetera.*
To identify the correct correlation and concepts for solving important problems with the help of EA, the organisational vision, objectives and strategy, EA elements such as perspectives, principles, key models and relevant relationships are extracted. These perspectives are the ways in which an organisation is viewed and can be controlled, such as Products, Processes and Culture.

   Is there such a characterisation of perspectives, key concepts, principles, et cetera, in your organisation or corresponding concepts?

4. *Our EA vision is developed into EA processes, products, people and resources.*
To translate this vision into effective actions it must be elaborated into processes, products, people and resources.
This includes:
   - The application of executive EA processes that deliver EA control process related products and EA specialist products such as Programme Start Architecture and Key Models respectively.
   - Managing EA, including maintaining EA, resulting in EA governance products such as EA development plans or evaluation reports.
   - Profile and competencies of the enterprise architects.
   - Tools such as an EA framework (e.g. Zachman, DYA, Architect or Aris).

   Is this kind of translation present in your organisation?

5. *In our organisation one or more control tools are used to rate organisational results in coherence.*
Does your organisation use control tools that measure integral coherence and on what basis are adjustments made as a result of the ratings? Examples would include the Balanced Score Card, INK, EFQM, et cetera.

6. *In our organisation one or more control tools are used to control change processes in coherence.*
Does your organisation possess control tools, which control integral coherence during preparation phases of important change processes, such as Prince II, business cases, programme start architectures?

7. *Our EA architects are involved in setting up control processes at a strategic and tactical level.*
Questions that arise are:
   - Are concrete company problems the reason for involving enterprise architects in control processes?
   - Are all relevant company components represented?
   - Are all named EA success factors met in the vision?
   - Is EA used as an integral control tool?
   - Are EA control products such as principle analyses, scenario analyses and integral business solutions used as a guide for decision-making?
   - Is the involvement of enterprise architects (with the EA control products) structurally embedded in organisational control processes?

   - Are the EA control products concrete in terms of usability, readability, clarity, composed of the correct level of detail, et cetera?
8. *It is known whether all our change programmes were developed with or without 'EA'.*
   Are the change programmes actually tested by means of EA control mechanisms, to ascertain whether they comply with architecture principles laid down in a Programme Start Architecture (PgSA)? Are established procedures followed for necessary deviations from the PgSA?
9. *In our managers' competence profile 'EA' is included as a competence.*
   If one wants to actually apply EA as 'coherence governance' one condition is that managers are familiar with it and can apply it. Is equipment for managers structurally organised with regards to knowledge and skills in the area of EA?
10. *Our managers understand and use EA products in their control processes.*
    Is EA actually embedded in the control of the organisation and not just something belonging to the 'ivory tower architects'?
11. *At least once a year there is an updated version of the content of our EA framework.*
    This statement raises the following questions:
    - Is the EA maintenance process well organised?
    - Do the specialist EA products (also called the EA building block products) meet quality criteria such as being up to date, consistent, et cetera?
    - Is input from the EA application processes consistently regulated?
    - Does the EA controller possess the necessary competencies and are the tools used of a professional level? For examples, tools to capture and leverage enterprise architectures.
12. *Those with end-responsibility for our change processes are accountable for time, money and quality as well as meeting EA principles and guidelines.*
    This statement is based on the idea that if this situation applies then:
    - Solutions and choices of approach are developed from an integral view of the organisation.
    - All responsible parties (direct and indirect problem owners) are actively involved in developing company solutions.

References

1. GEA Groeiplatform (in Dutch)
2. Enterprise architecture maturity model version 1.3. Technical report, National Association of State Chief Information Officers (NASCIO), United States of America (2003)
3. Lessen uit ICT-projecten bij de overheid, Deel B. De Algemene Rekenkamer (2008) (in Dutch)
4. Abcouwer, A., Maes, R., Truijens, J.: Contouren van een generiek model voor informatiemanagement. Primavera working paper, Universiteit van Amsterdam (1997) (in Dutch)
5. De Caluwé, L., Vermaak, H.: Learning to Change: A Guide for Organization Change Agents. Sage publications, London (2003)
6. Department of Commerce, Government of the USA. Introduction - IT Architecture Capability Maturity Model. Government of the United States of America (2003)
7. Henderson, J.C., Venkatraman, N.: Strategic alignment: Leveraging information technology for transforming organizations. IBM Systems Journal 32(1), 4–16 (1993)

8. Hoogervorst, J.A.P.: Enterprise Architecture: Enabling Integration, Agility and Change. International Journal of Cooperative Information Systems 13(3), 213–233 (2004)
9. Hoogervorst, J.A.P.: Enterprise Governance and Enterprise Engineering. Springer, Heidelberg (2009)
10. Iacob, M.-E., Jonkers, H., Lankhorst, M.M., Proper, H.A.: ArchiMate 1.0 Specification. The Open Group (2009)
11. Lankhorst, M.M., et al.: Enterprise Architecture at Work: Modelling, Communication and Analysis. Springer, Berlin (2005)
12. Lugtigheid, R.B.: Architectuur bij PGGM; een praktijkvoorbeeld. Technical report, Ordina (2007) (in Dutch)
13. Macmillan English Dictionary for Advanced Learners. Hueber, 2nd edition (2010)
14. Op't Land, M., Proper, H.A., Waage, M., Cloo, J., Steghuis, C.: Enterprise Architecture – Creating Value by Informed Governance. Enterprise Engineering Series. Springer, Berlin (2008)
15. Pettigrew, A., Thomas, H., Whittington, R.: Handbook of Strategy & Management. Sage Publications, Thousand Oaks (2001)
16. Project Management Body of Knowledge. Technical report, The Project Management Institute (November 2001)
17. Managing Successful Projects with PRINCE2. The Stationery Office (2009)
18. van der Raadt, B., Slot, R., Van Vliet, H.: Experience Report: Assessing a Global Financial Services Company on its Enterprise Architecture Effectiveness Using NAOMI, p. 218b (2007)
19. Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)
20. Schekkerman, J.: How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework. Trafford Publishing, Victoria (2006)
21. Schnelle, E.: Neue Wege der Kommunikation. Spielregeln, Arbeitstechniken und Anwendungsfälle der Metaplan-Methode. Number Heft 10. Hanstein, Königstein/Taunus, Germany (1978) (in German)
22. Slot, R.G.: A method for Valuing Architecture-Based Business Transformation and Measuring the value of Solutions Architecture. PhD thesis, Universiteit van Amsterdam (2010)
23. Smit, R.: Manieren om strategie te verknallen. Financieel Dagblad (February 23, 2007) (in Dutch)
24. Sowa, J.F., Zachman, J.A.: Extending and formalizing the framework for information systems architecture. IBM Systems Journal 31(3), 590–616 (1992)
25. The Open Group – TOGAF Version 9. Van Haren Publishing, Zaltbommel, The Netherlands (2009)
26. Van Dale Groot woordenboek van de Nederlandse taal. Van Dale, 14th ed. (2010) (in Dutch)
27. Van't Wout, J., Waage, M., Hartman, H., Stahlecker, M., Hofman, A.: The Integrated Architecture Framework Explained. Springer, Berlin (2010)
28. Wagter, R.: Sturen op samenhang op basis van GEA – Permanent en event driven. Van Haren Publishing, Zaltbommel (2009) (in Dutch)
29. Wagter, R., Nijkamp, G., Proper, H.A.: Overview 1th Phase - General Enterprise Architecturing. White Paper GEA-1, Ordina, Utrecht, The Netherlands (2007) (in Dutch)
30. Wagter, R., Van den Berg, M., Luijpers, J., Van Steenbergen, M.: Dynamic Enterprise Architecture: How to Make It Work. Wiley, New York (2005)
31. Yin, R.K.: Case Study Research – Design and Methods, 4th edn. Sage Publications, Thousand Oaks (2009)

PROASIS: As-Is Business Process Model Maintenance

Nuno Castela[1,2], Marielba Zacarias[1,3], and José Tribolet[1,4]

[1] Center for Organizational Design and Engineering,
INESC INOVAÇÃO, Rua Alves Redol, nº 9, 1000-029 Lisbon, Portugal
[2] Polytechnic Institute of Castelo Branco, Av Empresário, 6000-767 Castelo Branco, Portugal
[3] Research Center for Spatial and Organizational Dynamics, University of Algarve,
Campus de Gambelas – Building 9, 8005-139 Faro, Portugal
[4] Dep. of Information Systems and Computer Science, Instituto Superior Técnico,
Technical University of Lisbon, Campus Alameda, Av. Rovisco Pais,
1, 1049-001 Lisbon, Portugal
ncastela@ipcb.pt, mzacaria@ualg.pt, jose.tribolet@inesc.pt

Abstract. Business process modeling must offer a trustworthy, reliable and updated representation of different enterprise concerns. Nonetheless, it is acknowledged that model maintenance is a difficult task and most of the times modeling efforts in companies are limited to specific projects occurring at a particular time. After that, models just "sit on the shelf". This paper defines an "as-is" model continuous updating process that uses the annotation mechanism to create interaction contexts enabling business actors (1) to communicate and explicit their knowledge about processes and about their own work, and (2) to discuss existing process representations. To support the as-is model updating process in real organizational environment a prototype tool has been developed. This approach has demonstrated that organizational actors, since provided with a process and supporting tool, can act as active updaters of business process models by comparing the modeled with actually executed activities, becoming themselves organizational modelers.

Keywords: Enterprise Engineering, Business Process Dynamic Updating, Annotation, Collaborative Negotiation, Organizational Knowledge, Organizational Self Awareness.

1 Introduction

Enterprise engineering [1] puts together concepts, methods and technologies which allows to understand, model, develop and analyze business strategies, processes and supporting information systems, with special focus on their dynamics and inter-relationships. Enterprise architectures are enterprise engineering frameworks that allow representing organizations from different perspectives where the most commonly used include strategy, process, applications and technology perspectives [2]. Within enterprise architecture frameworks, business process models are used to communicate, document and understand the activity of organizations [3]. The model representing business processes at a particular time, is defined as the "as-is" model. In

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 53–82, 2011.

contrast the "to-be" model reflects future changes in processes resulting from process analysis made in the scope of enterprise initiatives (e.g., Business Process Management (BPM), Total Quality Management (TQM), Business Process Reengineering (BPR)). The goals of building as-is models include:

- Redesigning or improve the organization [4], [5], [6], [7];
- Improve enterprise integration [8], [9], [10];
- Act as a starting point to information systems architecture [11] and to requirements gathering in information systems development [12];
- Control the running processes (using workflow management systems, model based process control or business activity monitoring) [4], [7];
- Act as knowledge repository of the organization.

Achieving the latter goal enhances the ability of organizations to become learning organizations [13], [14]. Processes and activities contain all the information about how, when and who does the work flow [15] and constitute the only basis truly verifiable and understandable by organizational actors [16]. This is particularly true if business process models are built by gathering actual actor actions and interactions composing business activities, which in turn are orchestrated in business processes.

Organizational knowledge about activities and processes can be made explicit through business process models built with the contribution of the knowledge of all the individuals working in organizations [17]. This highlights the need of promoting and facilitating the use of this type of models. Business process model repositories may also allow incorporating new knowledge in an iterative and incremental fashion by becoming the subject of conversations among organizational actors. To achieve this it is necessary to understand the interaction dynamics between organizational actors and the interaction contexts created by such interactions.

Business process models can also have a key role in enhancing organizational self awareness. Human beings are self-aware by nature. Organizational self awareness is the result of a conscious process that involves (1) efforts of each individual member of an organization of making sense of their environment and (2) discussions and negotiations among those individual members in order to reach agreements around their "sensemaking" of the environment [18]. At an organizational level, self-awareness is only achieved by a proper understanding and management of the interactions among organizational actors [19]. In order to properly support organizational self-awareness, business process models need on one side to accommodate individual, group and organizational views emerging within personal action contexts, as well as inter-personal and group interaction contexts [20]. On the other side, they must guarantee the consistency of the whole model [17].

Despite being acknowledged as an important asset for knowledge management purposes, empirical studies have shown that the as-is business process model is not continuously updated, because the maintenance of this representation is not straightforward [21]. Currently, the depiction of business process models is limited to efforts including BPM, BPR, and TQM among others, and then "sit on the shelf" [22].

The work presented in this paper aims at defining a process to streamline and automate the continuous updating of business process models in order to maintain their alignment with the actual processes and activities being executed, using (1)

annotation and (2) collaboration mechanisms. This work argues that if as-is models could be updated continuously by incorporating individual knowledge and enabling discussions and negotiations among the contributing individuals, they could be used as a permanent repository of organizational knowledge and therefore, could be a basis to support and enhance organizational self awareness.

More specifically, this work has the following research goals:

- Definition of the updating process model of as-is business process to keep the alignment between the model and actual execution, recognizing organizational actors as key players in the process to reduce the gap between the actual of operation of the organization and its representation.
- Development of a supporting tool, suitable for use in real organizational environment.
- Definition of an mechanism for making explicit the misalignments found
- Extending the previous mechanisms by allowing a communication channel between the actors (the part) and organization (the whole). In order words, allowing the discussions of private and incoherent views expressed in graphical or textual representations of the organization to enable the construction of shared representations.

The remainder of this paper is structured as follows. Section 2 describes the theoretical background supporting our approach. Section 3 summarizes related work on other collaborative business process modeling tools. Section 4 defines the as-is enterprise model dynamic updating process. Section 5 presents the supporting tool prototype developed. Section 6 presents a case study set to use the defined process and tool in real organizational environment. Section 7 presents our conclusions and future directions.

2 Background

The present research aims at giving contributions to enterprise architecture and business process modeling disciplines, and is supported on theoretical and practical concepts coming from other disciplines including organizational, management and social sciences, as well as software engineering. This section summarizes the state of the art on enterprise architectures and business process modeling, the contemporary paradigm of organizational science and some related theories, Haberma's theory of human communication and an enterprise modeling approach supported by them. The annotation mechanism used in updating software engineering processes is presented. Finally, related work about collaborative business process modeling tools is summarized.

2.1 Enterprise Architectures and Business Process Modeling

The concept of enterprise architecture has a key role in giving the meaning of the organizational whole, while respecting the independence of the various perspectives of its constituent parts. According to Schekkerman [2], enterprise architecture is a full

expression of the companies that acts as a collaborative force between business aspects: operational, organizational, structure, business terms, processes, data and automation.

To Lankhorst [23], Enterprise Architecture is a comprehensive set of principles, methods and models used in the design and implementation of organizational structure, business processes, information systems and technology infrastructure.

There are several frameworks for Enterprise Architecture, some more specific and some more general as it may be applied to any organization. The CEO framework [11] and the Zachman framework [24] are examples of the latter ones.

One of the most important features of Enterprise Architecture is the possibility it gives to minimize the misalignment between the various components of the architecture, thereby minimizing enterprise misalignment as it relates to strategy, processes and activities, organization, information, systems and technology [2].

According to Whitman [25], the business process model is a symbolic representation of the organization and things that the organization is dealing with. Fox [26] gives a more detailed definition because he defines the business process model as a representation of the structure, activities, processes, information, resources, people, behavior and restrictions of an organization. The model of business processes allows a view of the organizations at the operational level. The business process model of an organization, according to Giaglis [27], focuses on identifying the activities that compose each process by defining the workflow between them and where they are performed and identifying the resources used in each activity.

The elicitation of business process models means to extract information from current business procedures and existing applications and encode them formally. A business process model should be able to provide several pieces of information to its users. These elements include, for example, which activities compose the business process, who performs these activities, how and why are implemented, and what informational elements manipulate. Any modeling technique should be able to represent one or more of the following modeling perspectives: functional, behavioral, organizational and informational [27].

From the organizational knowledge point of view, the business process model can be characterized as:

- A facilitator of knowledge creation and sharing [14].
- An enabler of the evolution of tacit knowledge (individual) to explicit knowledge (organizational), as it allows the use and sharing of knowledge, driving the creation of new knowledge [28].
- An implementation of the organization media, through private images and public maps and a representation of both the individual organizational image and the whole organization, allowing individuals to know their place in organizations [29].
- A simulation of hypertext organization [28], since it can be implemented with multiple navigable layers, including the layer of individual views (business system layer), the layer where these individual views are articulated and fit (project team layer) and the layer of the general model, containing a unique global view of the organization (knowledge layer). These layers must be aligned in real time to ensure consistency and integrity in representation.

- A representation of two theories of action of Argyris and Schön, the professed and implemented (and therefore have to have sensors that capture the action implemented, that collect and analyze the consequences, allowing the redesign of governance variables and strategy of action) [29].
- A basis for monitoring the very complexity of the model using simple modeling primitives (Entity, Role, Context, Activity) [30].
- A platform to promote the updating of the model itself, by detecting the misalignment between reality and representation of reality using annotations [31].

There are several notations and methodologies to model business processes. BPMN and DEMO are, respectively, examples of a notational language to model business processes and a methodology to model organizations (and business processes as well).

BPMN. Business Process Modeling Notation (BPMN) is a standard graphical notation for representing business processes and was originally developed by Business Process Management Initiative (BPMI) which merged later with the Object Management Group (OMG), an entity that now manages its development cycle.

The main objective of BPMN is to provide a standard notation that can be read and understood by all stakeholders of organizations, including business analysts, programmers and managers [32]. To achieve this objective, BPMN was created as a lingua franca, bridging the gap between process design and implementation. Another objective of BPMN is to ensure that the language (XML based) developed for the execution of business processes can be viewed with a business oriented notation.

DEMO. DEMO (Design and Engineering Methodology for Organizations) meta-ontology is used to develop ontologies that must meet some requirements that cover important properties of ontologies [33], of which the following can be highlighted: It should clearly distinguish, based on solid theoretical grounds, between the world (states and events) and the causes of change in this world (actors and acts). Ontology is about the comprehension of the essence (which means the nature or being of things) of something. Nowadays and beyond the original meaning, enterprise ontology have also a practical goal because it serves as a basis of common understanding in a particular area of interest [34]. So, enterprise ontology is the specification of the conceptual model of the organization essence, independently of its implementation [33]. This conceptual model should be coherent, where the several aspects of the model can be distinguished, without losing the notion of the whole, comprehensive, in which all relevant issues are covered, consistent, where the models are free of contradictions and irregularities, concise, containing no superfluous questions and only showing the essence of the operational enterprise model, avoiding all the implementation and realization questions [35], [36]. For each organization there is only one ontology that shows the essential activities of business, the players involved and the products and services that they are dealing with.

The complete organizational ontology is composed by four models. The construction model (CM) specifies the composition, environment and structure of an organization. The CM is expressed by the actor transaction diagram (ATD) and by the transactions result table (TRT). The action model (AM) specifies the action rules that

govern the actors in order to handle their agenda. The state model (SM) specifies the legal states of the c-world and p-world. The SM is expressed by the object fact diagram (OFD) and by the object property table (OPT). The process model (PM) specifies the legal event sequences in both worlds by stating the atomic process steps and their causal and conditional relationships [33].

The only concern of ontology should be on the essential aspects of production and communication in an organization and not on how actors communicate (matters relating to implementation). The notion of ontology system aims to understand the essence of the construction and operation of an organizational system.

2.2 Organizational and Social Theories

Classic approaches of organizations are positivist, i.e. they hold an objective view of reality. Nevertheless, this classical paradigm is being replaced by a new one that regards reality as something that is negotiated and constructed by people's interpretations of what happens around them [37]. Consequently, the contemporary position held on the organization is identified as constructivist. In constructivism, reality is neither completely objective nor subjective. Rather, is it objectified, that is, it is constructed in a way that makes it seem objective. Contemporary views are centered on how organizational agents continually (re)create and change the organization. Constructivist theories argue that organizations exist largely in the minds of organization members in the form of cognitive maps, or images. In talking about organizations and designing maps of it, they are reified, that is, they are made real. Hence, the existence of shared maps requires social agreement and cooperation.

Organizations are also currently regarded as complex systems [38]. Bohm [39] argues that in every complex system there are hidden processes below the surface of reality, which explain the world stage at any time. The author addresses the study of systems that exhibit non-linear behavior (as opposed to cause-and-effect behavior). Complexity introduces notions such as self-organization and emergence (as opposed to deterministic motion), chaos and unpredictability (as opposed to command and control), or sensemaking and understanding (as opposed to rationalizing and predicting).

Another important concern of the constructivist paradigm is organizational evolution. Axelrod and Cohen [40] have taken the principles of complexity and evolution and have put together a conceptual framework for analysis and (re)design of social, political, and organizational systems. The authors call it a 'population approach to complex adaptive systems' given the special emphasis it places on populations. Change in populations result from assessing the performance of the strategies of the agents in the population, according some to measure of success. Whereas successful strategies will be repeated (or copied from another agent), unsuccessful ones will be changed or eliminated altogether. Selection is the result of mechanisms such copying, changing or combining strategies. When a selection process leads to improvements, then selection leads to adaptation.

Finally, the new paradigm emphasizes the notion of agency. In order to be fully understood, agency must be regarded at collective and individual levels. The conceptual framework of Axelrod and Cohen [40] addresses agency at organizational and societal levels. Structuration theory [41] explains the role of agency in the

(re)production of societies. Activity theory (AT) [42] analyzes the formation and evolution of activities, and human consciousness. The organizing unit in AT is the activity, and agents are individuals and groups.

Innovation and knowledge creation can emerge from the interaction of organizational actors groups [43]. It is noteworthy that this position shares the basic assumptions of the contemporary paradigm of organizational science, which highlights the importance supporting interactions among human agents in managing any business process.

Regarding human interactions it is essential to understand the assumptions and rules they use to communicate. According to the theory of communicative action the inherent aim of language is to reach understanding and bring about consensus among individuals [44]. The primary function of speech is to coordinate the actions of several individuals, and to allow interactions to unfold orderly. Speech fulfills this function because the meaning of utterances rests on reasons. This view is called, 'the validity basis of meaning', where validity means a close relation between reasons and consensus. Validity claims are always understood to have been made in the act of speaking. When the hearer rejects the validity claim of the speaker, the communication breaks down and the communicating agents change from action to a discourse situation. Discourse is communication that reflects on the disrupted consensus in an action situation, and aims at reaching rationally motivated consensus. Discourse picks up in the common practice of argument and justification interwoven in everyday life. Discourse is the default mechanism in regulating everyday conflicts within modern societies. In short, Habermas argues that the social order of modern societies rests on the basis of communicative action and discourse.

This theory is important to understand how organizational actors can reach agreements and consensus on representations of their activities, expressed through business process models. Zacarias [17], proposes a conceptual model based on the aforementioned theories centered on agents and their contexts as a complementary perspective to the current perspectives of enterprise architecture (process, information, applications and technology) to enable the alignment between the structure and behavior of enterprises defined in existing perspectives with the actual behavior of its human agents. This conceptual model uses the three agency layers defined in agent architectures (execution, coordination, change/learn) and defines contexts according each agency layer. In the execution layer, contexts are regarded in terms of recurrent actions as well as action and interaction patterns. At the coordination layer contexts are regarded in terms of commitments resulting from agent interactions. At the change/learn layers contexts are regarded in terms of the rules that both enable and restrict the changes that can be made to action and interaction patterns. This work also makes and instrumental use of the concept of context at the execution layer in defining a methodology for the alignment between agent and business process perspectives, The proposed methodology encompasses (1) capturing agent actions and interactions, (2) discovering personal and inter-personal contexts by grouping related actions and interactions, (3) uncovering action and interaction patterns within such contexts and (3) assessing the alignment between contextual actions and interaction patterns with activities composing business process models.

2.3 Annotation Mechanism

In general, annotations are an addition of information on a particular section of a document or other informational entity. Annotations have specific uses in distinct areas: in biology for genome annotation [45], in law science for annotated versions of legislation books, in language science for linguistic annotations, in programming languages like Java [46], in modeling languages like UML [47], in Web pages for analysis of documentation [48] and for adding comments, explanations or other external reference [49], in hypertext for establishing new connections, interpret materials and promote the creation of structure or content, increasing the body of inter-related material [50]. Annotations should capture the activities, resources and the context involved. The continuous improvement of processes requires that the experience captured should be continuously incorporated into business processes and portrayed in the as-is model. The systematic capture and storage in the context where the experience was captured has three major benefits [31]: the experience can become explicit; the experience may be incorporated in the description of the process; the analysis of the experience can be reused in other processes for process improving.

The annotation mechanism used in software engineering to capture the changes (and their justifications) for software projects from the implicit knowledge of development teams [31], seems well suited to be employed in enterprise engineering, namely in annotating business process models.

2.4 Collaborative BPM Tools

According to Borghoff [51], the widespread use of personal computers and associated networks meant that these resources began to be used not only for distributed data processing, but also to work collaboratively, so that, in the literature, the designations Computer Supported Cooperative Work (CSCW) and groupware (software used by groups) were introduced. The term supported refers to the shared data access in a non-coordinated way and to the synchronized modeling of relationships and group interactions. The CSCW designation refers to the research field that was created, and groupware refers to the solutions and tools designed to support collaborative work in practice.

The role of individual members of groups is an important aspect in the development of CSCW tools, because the roles help to structure the interactions between team members and to define the functionalities and access rights of the group. The roles define the social function of individuals in relation to group process, to group organization and relatively to other group members. The roles define rights and obligations in relation to group process.

Ellis [52] defines groupware as "computer-based systems that assist groups of people who share a common task (or common purpose), which provide an interface to a shared environment". The goal of groupware is to support communication, collaboration and coordination of group activities. The group activities are potentiated by a proper and delicate balance between social processes and properly structured technology.

A business processes modeling tool is an automated system that provides capabilities to build business process models [46]. Research on collaborative BPM tools have been addressed by a number of authors.

Gonzalez [53] identifies a set of features necessary to collaborative tools for modeling business processes and analyzes a set of commercial tools to check whether these features identified are implemented. In general most of the tools analyzed provide collaborative features, but many of the features are not present unless, in some cases, all modules have to be purchased, or in other cases, some modules have to be purchased from other vendors.

The requirements of collaborative tools for modeling business processes can be classified according to spatial-temporal matrix of groupware [51] because the interaction can happen at the same time (synchronous) or at different times (asynchronous) and the participants in interaction can be in the same location or different locations. This classification is important to analyze the characteristics of a collaborative tool, since it serves to determine whether the tool covers these requirements or not. After reviewing the available tools, three categories of business process modeling collaborative tools were identified [52]:

- Web tools with modeling support (WS).
- Client / server local tools (CS).
- Export / import Documents (I / E).

Gonzalez [52] analyzed 35 tools from the market for evaluating the collaborative features considered most important, which include: Web Publishing, Model Viewer, Information Reports, Version Control, User Profiles, Comments and Notes, Ability to Disaggregation / Aggregation and other collaborative features not included in the analysis: Notifications to alert the participants that changes were made to the model during the asynchronous work and chat for discussion among participants.

Rittgen [54] suggests that while the modeling literature is abundant, the majority describes the use of notation in a descriptive way, instead of a prescriptive way, because the most common problems that people experience during the modeling process are not issued. Of the descriptive approaches, only a few are dealing with collaborative modeling based on groupware systems. All the others assume the scenario where only one modeling expert creates a formal model. However the following issues should be taken into account:

- The development of a model is rarely done by just one modeler, but by a team that may involve business representatives and people from outside the enterprise.
- The problem domain of business modeling is unstructured and formal languages have a limited use.
- The objective of providing a tool for collaborative modeling requires the identification of detailed stages involved in the modeling process.

Rittgen [54] argues that the process of enterprise modeling involves negotiation conversation type, involving teams (possibly from inside and outside the organization). The modeling process can be seen as a collaborative modeling process, absorbing the benefits of the group decision support systems.

For Rittgen, modeling is a conversational negotiation. To prove this point of view, a modeling experience from which some conclusions were extracted was set up, considering the top four steps of organizational semiotics ladder [55]: syntactic, semantic, pragmatic, and social.

The conclusions drawn from the social and pragmatic levels, through the observation of individuals involved in the business modeling process over three years, are the following:

- At the social level, it was found that social norms among the modeling team are mainly made of rules to determine whether a proposal is accepted or rejected. At this level there are two types of rules: majority and seniority.
- At the pragmatic level, two types of behavior were discovered, and each one can be classified into two subcategories: (1) Understanding, which is concerned with the text of the description of the case or with the modeling language; (2) Organization of the modeling process, which involves two types of activities: setting the agenda and negotiation. Agendas are a tool for the rigorous structuring of modeling sessions, but can be adapted, if necessary.

Most activities at pragmatic level are associated with negotiation. An analysis of workflows at pragmatic level revealed a structure that goes beyond the simple identification of generic activities, so the negotiation process follows a certain pattern (Figure 1).

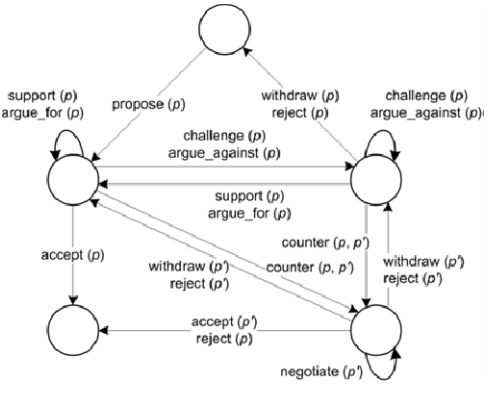


Fig. 1. Negotiation pattern [29]

This pattern consists of an initial and rejection state at the top, in a state where acceptance is favored (top left), a state where rejection is favored (top right), a sub-recursive state to negotiate a counter-proposal (below right) and a state of acceptance (below left). Each of the states allows a certain set of activities that drives the pragmatic negotiations to different states. The parameters that concern the modeler performing the activity and the argument (if present) were left out. In general, any modeler can perform any operation but there are rules that must be observed:

- A modeler that makes a proposal implicitly assumes that he supports it;
- The modeler who withdrew a proposal is the same modeler that originally made it;

- A counter-argument is made by a different modeler;
- A counter-proposal can be done by a modeler or the modeler who made the initial proposal (to accommodate counter-arguments).

Groupware systems for collective sense-making address an important issue in collaborative modeling and can be used as the core of a support modeling system There is a need to implement a component of negotiation that facilitates the structuring of arguments and decisions regarding modeling choices. The model shown in the figure 1 can serve as the initial workflow that controls the negotiation component.

3 As-Is Business Process Model Dynamic Updating Process

The As-Is Business Process Model Dynamic Updating Process proposed in this work is based on a set of assumptions, which result from the reviewed literature on organizational management and organizational sciences, enterprise architecture and business process modeling, that is summarized below.

The operation in organizations is described in terms of processes and activities where the processes are composed of activities flows. Activities are an abstraction of what the organization does. The definition of activities in these areas involves shared understandings about their objectives, resources consumed and produced, roles and procedures involved. Enterprise architecture has several perspectives (strategic, organizational processes, information, application, technology). In the process perspective there are several levels of representation (operational, management, knowledge). The as-is process model represents the operational level of the organizations at current time. This model consists of business processes model and the organizational model (organizational chart).

Organizations are complex adaptive systems created and maintained through the interactions of its organizational actors. Organizational actors are the individuals, which can form groups of individuals, working in organizations, which in turn are also complex and adaptive entities with capacities to act, monitor, analyze, learn and change (themselves and the organization). The implementation of activities is constrained by human-social factors such as needs and motivations of the actors involved, the tools as mediators between the actors involved, and shared socio-cultural rules. In addition, activities are subject to constant change. The creation and maintenance of operational processes models involve communication and negotiation processes between the actors involved. A single organizational actor may have several behaviors. Due to their multi-tasking, each actor can have multiple personal contexts and participate in various group or inter-personal contexts. Their behavior is determined by the role played under certain context.

The idea to explore in defining the AS-IS business process model dynamic updating PROcess (PROASIS) is based on the detection of misalignments between the shared model and ongoing executed processes. Misalignments are detected by the organizational actors executing activities belonging to a particular business process. PROASIS allows these actors to use annotations as a mechanism to collect the updates that they want to make. The language used to represent business processes is BPMN due to its simplicity and widespread use [32]. PROASIS is supported by a

groupware prototype tool that distributes the business process model to the organizational actors, and supports the gathering of annotations as well as the underlying negotiation for refining and approving the annotations. This tool supports also the designing of new versions of the models updated due to the annotations, by the organizational actors themselves.

3.1 PROASIS Definition

PROASIS is a support process performed by people (organizational actors) that work in operational business process and share a common representation of that same process. In PROASIS, annotations are used to build updating proposals to the model of a particular operational process model in order to align it with the process perceived by each organizational actor. These proposals aim to make the corrective maintenance of the business process model and can have two objectives: to correct the model or to increase its detail.

After making an annotation on a modeling element (which depends on the level of granularity), a negotiation with the actors who eventually share the same action context may exist. This negotiation/discussion will be made by all stakeholders of the annotated element in order to clarify the original purpose of the annotation. All actors involved in this review should declare the agreement or disagreement with the annotation made to the model element.

After the review of the annotation, the annotation should be evaluated by the actors enabled to do so, having some degree of responsibility on the executed activities or on the organizational actors involved. If the evaluation of the annotation (and any reviews made to it) results in an approval, the changes requested in the annotation could be incorporated in the new version of the process model by the modeler.

Figure 2 shows the structure that (1) identifies the generic activities in PROASIS and (2) shows the negotiation pattern involved in review and evaluation steps.

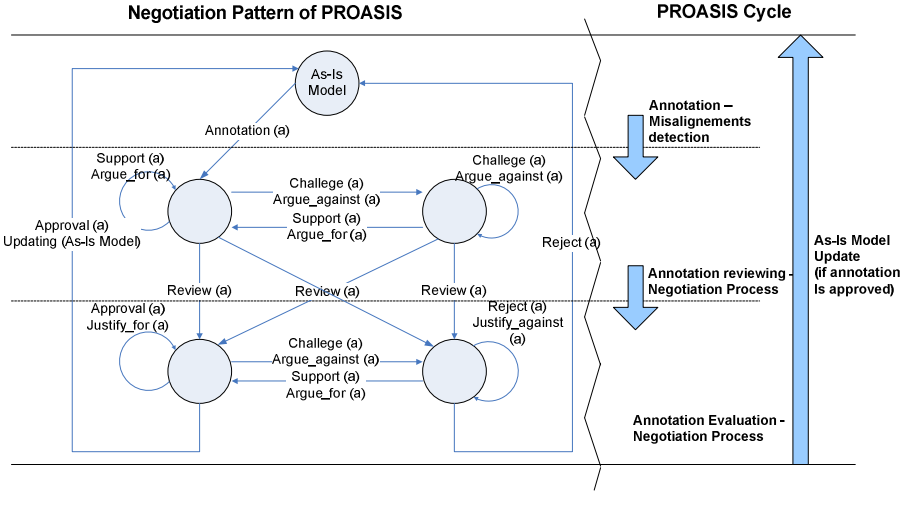


Fig. 2. PROASIS Negotiation Pattern and Steps

This pattern consists of an initial state associated with the as-is model, which can be changed if an annotation is made and then approved in the evaluation step of PROASIS cycle, which involves negotiation. Before evaluation and after the annotation be made, an intermediate negotiation step will be carried out, called review, which should be taken into account in the evaluation step. If the result of evaluation is an approval, it may lead to an updated as-is model through the creation of a new version of the distributed as-is model. This pattern is based on the negotiation pattern of Peter Rittgen [54] used for collaborative business process modeling and which is adapted here for the collaborative updating of business processes (PROASIS).

To define a dynamic update process whose use is as comprehensive as possible, it was necessary to consider the various levels of granularity that a business processes model can provide

The levels of detail proposed by Zacarias [20], which derive from the contexts of the operational model, are considered to support the as-is business process model updating (Figure 3) [56]:

- Process (organizational context).
- Activity (group context).
- Individual actions and interpersonal (individual and interpersonal contexts).

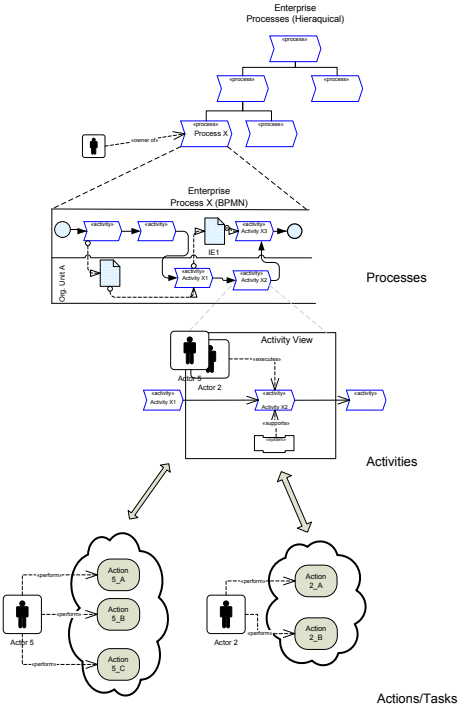


Fig. 3. Granularity Levels of Business Process Model [56]

These levels of detail of the model are related to the architecture of actors (actor, a pair of actors, a group of actors and organizational unit). In this work, the organizational unit is considered an additional level of detail, because the organizational chart is part of the enterprise modeling, and also because the organizational units intersects the representation of activity flows in BPMN diagrams, in which swimlanes typically represent organizational units.

The action level of detail, although considered in this work, is not subject to annotations for updating, because its representation depends on the personal discretion of each organizational actor. In addition other factors were taken in consideration in this decision:

- It is not usually represented in business processes models (e.g. BPMN).
- The context of action represents the set of actions that each actor plays in the organization in order to perform one or more activities, which not corresponds to a shared and homogeneous vision with the same granularity.
- Two or more actors can perform the same activity performing different personal actions, but with the same common goals.
- There may be repetitions of operations of the same actor in the implementation of different tasks, so the cardinality of the relationship between activities and actions is many-to-many.

However, this level of detail is important because it represents the view that each individual actor has about the work he executes in organizational context, and may therefore contain within itself the motive that leads each of the actors to propose organizational changes and updates to common shared models (activity, process). PROASIS do not update the action level of detail, mainly because this level is not a single coherent representation, which can be distributed without ambiguity by all stakeholders of organizations. However, the individual actors who act in the personal and interpersonal action contexts have to monitor the common parts of model and propose changes to implement and incorporate their individual vision in the greater levels of detail. In this way they may give their personal contribution through discussions generated by reviewing and evaluation of the annotations, involving the pairs of actors, groups and organizational units in the updating discussions, strengthening the model role as a common vision at every level of detail presented.

The set of modeling elements and the roles that are considered as standard annotators in PROASIS, can be extracted from figure 3 (table 1).

Table 1. Levels of detail, modeling elements and actors roles in PROASIS

Model	Level of detail	Modeling element	Annotator role
Operational	Process	Process	Process owner
	Activity	Workflow, Activity Informational Entity, Support Information System	Executor
Organizational	Organizational unit	Organizational unit	Organizational unit responsible

The organizational roles of each model presented in the table can take on different roles on PROASIS depending on the modeling element annotated. Note that an annotation is always done by individual initiative in a particular context, involving various actors in the later stages of reviewing and evaluation. This means that the updating context (PROASIS) captures the actors involved in the action context (operational level), consisting in a subset of actors of the operating model - people who participate in the reviewing and evaluation of the annotations.

3.2 Levels of Detail

The levels of detail considered for PROASIS are the same as the as-is business processes model distributed in organizations have. This model is usually composed by the levels of detail described in Table 1. The modeling elements considered in this model and that may be subject to updating proposals through annotations are those shown in Figure 4 marked from A trough F. Depending on the modeling element annotated in each level of detail, the various actors who play different roles in the operational model (which are also represented in figure 4) may play different roles in the PROASIS, as annotator of the model, and reviewers or evaluators of the annotations [56]. Examples of the relationships between these roles at each level of detail considered are depicted in the following subsections of this document.

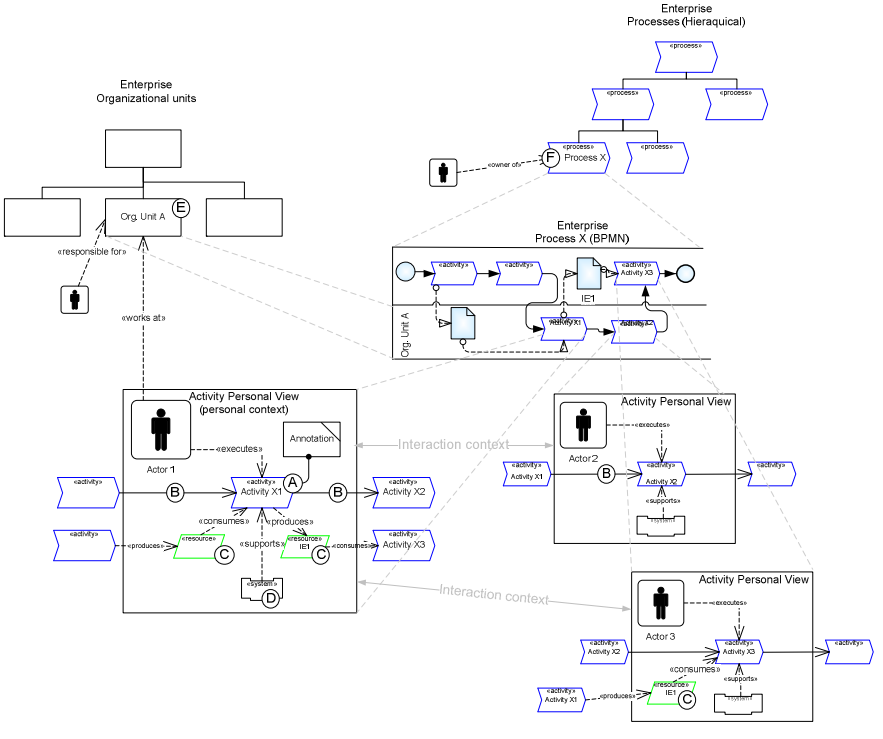


Fig. 4. Operational Model Contexts and Actors [56]

Figure 5 shows the set of modeling elements that can be considered for an actor to make an annotation at the activity level of detail: 1 - Activity; 2 – Workflow; 3 – Informational Entity and 4 – Information System.

An actor can annotate activities and other modeling elements attached to the activities since he is an executer of the activity in the operational model.

The review done at this level of detail involves all the actors who, according to the model, share the modeling element that was annotated. Table 2 shows the actors who are considered reviewers for each annotation type. The reviews are used to express agreement or disagreement with the annotation made.

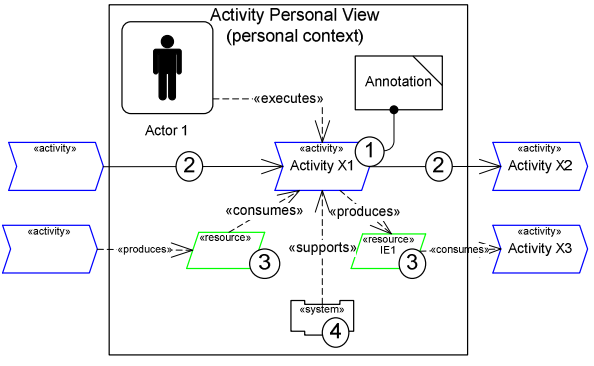


Fig. 5. Personal Activity View (Personal Context)

Table 2. Reviewers and Modeling Elements

Annotated modeling element	**Reviewers**
Activity	Executing actors of an annotated activity
Work Flow	Actors who executes an activity which is the origin or arrival of workflow
Informational Entity	Actors who executes an activity which creates or reads an informational entity
Information System	Actors who executes an activity which is supported by information system

The evaluation of the annotations will be performed by actors who have operational responsibility to the annotated element and hierarchical responsibility before the annotator, so whatever the element annotated at the activity level of detail, the annotation will be evaluated jointly by the owners of the process containing the activity and the heads of organizational units of actors involved. The evaluators may approve or disapprove the annotation. An annotation could be considered as a basis for updating the as-is model by the modeler only if the result of the evaluation of the original annotation will be the joint approval by all the evaluators.

Annotations made by the process owners to any modeling element belonging to the processes that they owns, or by the organizational unit responsible for any activity

executed (or for any modeling element manipulated in the execution of the activities) by the organizational actors that belong to that organizational unit, can be considered optionally. If this option is considered, these actors also belong to the reviewers group.

At Process level of detail, the PROASIS standard annotator is the process owner (figure 6). The process owner can make annotations to the process that he owns as a whole, and this annotation can be reviewed by the organizational units responsible that are responsible for actors who execute the activities that comprise the process.

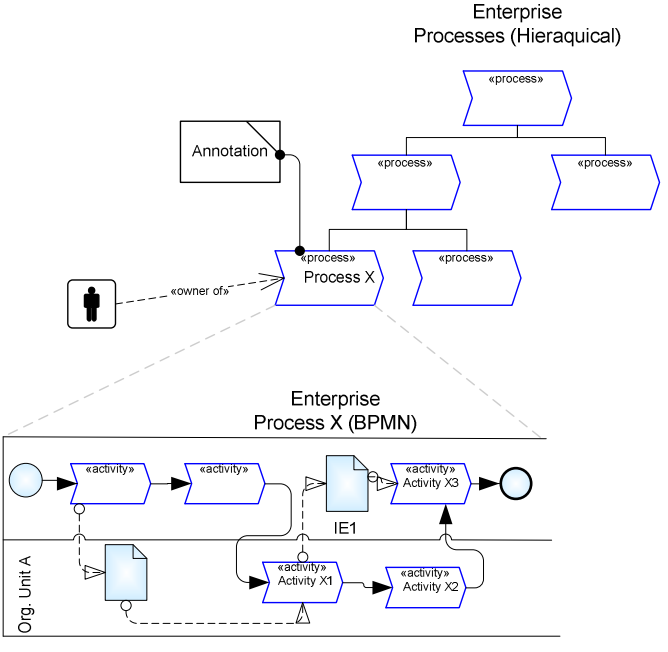


Fig. 6. Process Annotation

The evaluation of the process annotation is made jointly by those involved in its reviewing (process owner and organizational units responsible involved in the process).

Optionally, the frontier process may be considered as modeling elements that can be annotated. In this case the modeling element that will actually be annotated is the workflow that connects the two activities that are on the border between the two processes. If a process owner makes an annotation to a frontier process, the reviewing and evaluation of the annotation will have to involve, in addition to those in charge of organizational units that have responsibilities in both processes, the owners of both processes.

If the model is constructed only up to the process level of detail, only the processes owners, the organizational units responsible and the set of executer are known. It can be assumed that either the process owner or the executors of the activities that

comprise the process can make annotations. The set of executers, the process owner and in addition, the organizational units responsible, can participate in the reviewing of annotations. Consequently, the evaluation is made by the owner of process and/or the organizational units responsible involved. If the original annotation is approved, can lead to a new as-is model version.

At the organizational unit level of detail, the typical annotator of the organizational unit annotated as a whole is the organizational unit responsible (Figure 7).

The review of this kind of annotation involves the organizational unit responsible that originated the annotation and the process owners whose processes have activities performed under the responsibility of the annotated organizational unit.

The Evaluation involves the same actors involved in reviewing the annotations, requiring a joint annotation approval to be considered valid for the modeler in order to update the model, generating a new version of the as-is model.

Optionally, and considering the hierarchical characteristics of organization diagram, which represents organizational units in a tree structure model, the subunits responsible can also be annotators. In this case those responsible can also participate in the annotation review, but cannot participate in the annotation evaluation. Moreover, also the organizational responsible of the higher units can become annotators of the lower level units, but in this case, this also implies the participation in the evaluation.

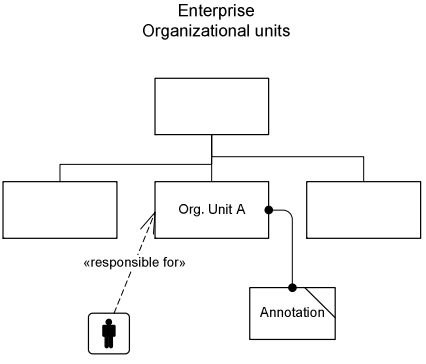


Fig. 7. Organizational Unit Annotation

3.3 Annotation Categories in PROASIS

One of the goals in defining PROASIS was to approach as much as possible its collaborative updating process to the problem domain of business processes modeling. To achieve this, some options were taken.

The annotation stays attached to the annotated modeling elements. Consequently, reviews and evaluations stay attached to the annotation made.

Some categories of annotations were created to restrict the universe of discourse. These categories were derived from the work of Becker-Kornstaedt [31] that recognized that the integration of annotations in the models could be classified as adaptive maintenance (to capture context changes in activity or process execution),

perfective maintenance (to capture better and more information about execution) or corrective maintenance (to capture modeling corrections). Thus the following categories of annotation were created: correction, detail augmentation and adaptation. The annotations can contain, in addition to the category, a textual explanation or diagram made by the annotator, which allows the actors to annotate the model through a draft model diagram containing the proposed corrections. In this case, the universe of discourse is restricted because this diagram must comply with the notational language used for process modeling.

In the review of annotations, the actors should express agreement or disagreement regarding the notation made, complemented with text.

In the evaluation of the annotations, the actors should express approval or disapproval regarding the annotation made, complemented with text.

3.4 Modeling PROASIS with DEMO Methodology

The notion of ontology aims to understand the essence of the construction and operation of an organizational system. The following text expresses the essence of PROASIS:

“The "client" of PROASIS (corresponding role of the operational model that detects misalignment between the model and "reality") wants to update the model, so he makes an annotation (update request). This update request is received by the modeler (which is who actually update the model if the annotation is approved) and by the reviewers. When reviewers receive the annotation, they can begin the review of the annotation (which is optional). The evaluation of the annotation is made based on the analysis of the annotation (update request) and reviews. If the annotation (request update) is approved, the model will be updated and delivered to the "client"”.

In the text above, an independent transaction (T1) is identified, corresponding to the delivery of a final product to the environment, which is in this case, the delivery of an updated enterprise business process model. The production fact of this transaction is the delivery of a particular update of the model. The following transaction result table (table 3) shows the transaction T1 and the dependent transactions T2, T3 and T4.

Table 3. Transaction Result Table of PROASIS

Transaction	**Result**
T1 – Model Update	R1 – The model M is updated
T2 - Annotation	R2 – The Annotation AN is created
T3 – Revision	R3 – The Revision R is made
T4 - Approval	R3 – The Evaluation AV is made

The process structure diagram shows the structure of PROASIS (figure 8): after an actor of the operational model making an update request (annotation), it can be seen that to deal with the promise of T1 (T1/pm) the modeler performs two acts: the coordination act T3/rq (which means that he promises to update the model based on the annotation made only if there is an approval of the evaluator) and the execution act of T1 (which will only be executed if the evaluator approves the annotation). At

the same time, when the actor of the operational processes requires an update to the model (T1/rq), this same actor "transposes" to PROASIS as an annotator, and T1/rq leads to T2/rq, and after executing T2, also leads to T3/rq, since it requires the review of the annotation that he did. The transactions T3 and T4 both imply negotiation processes. The new instance of the model produced in T1, reflects the changes expressed in the annotation, and the set formed by annotation/reviews/approvals became part of it.

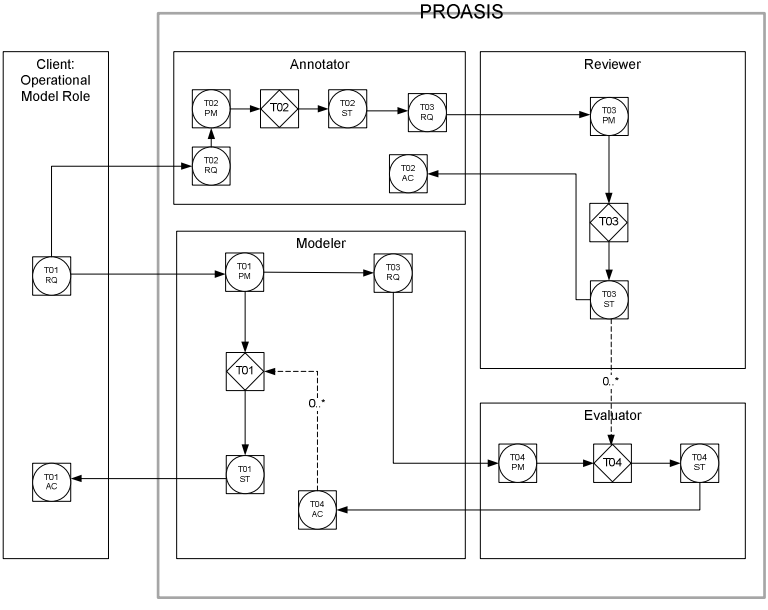


Fig. 8. Process Structure Diagram of PROASIS

Figure 9 shows the association between the two models, the operational (that one that is being updated) and the PROASIS (that one that is used to update the operational model, represented with the DEMO actor transaction diagram), modeled with the Actor Transaction Diagram (ATD) of DEMO. This association is expressed by the dynamic relationship among the roles of each model. The initiation of the transaction T1 (in PROASIS) is made by the operational actor role that makes an annotation, which could by any of the roles defined in the operational model, depending of the annotation context. Consequently, the review and evaluator roles are dynamically assigned because they depend of the associations among the modeling elements of the operational model. PROASIS modeling with DEMO shows and emphasizes: The essential elements of PROASIS, the organizational roles involved (annotator, reviewer, evaluator and modeler) and the transactions of PROASIS (annotation, review, evaluation and modeling), and their relationship with the roles of the operational processes, which act as initiators of the transactions on PROASIS.

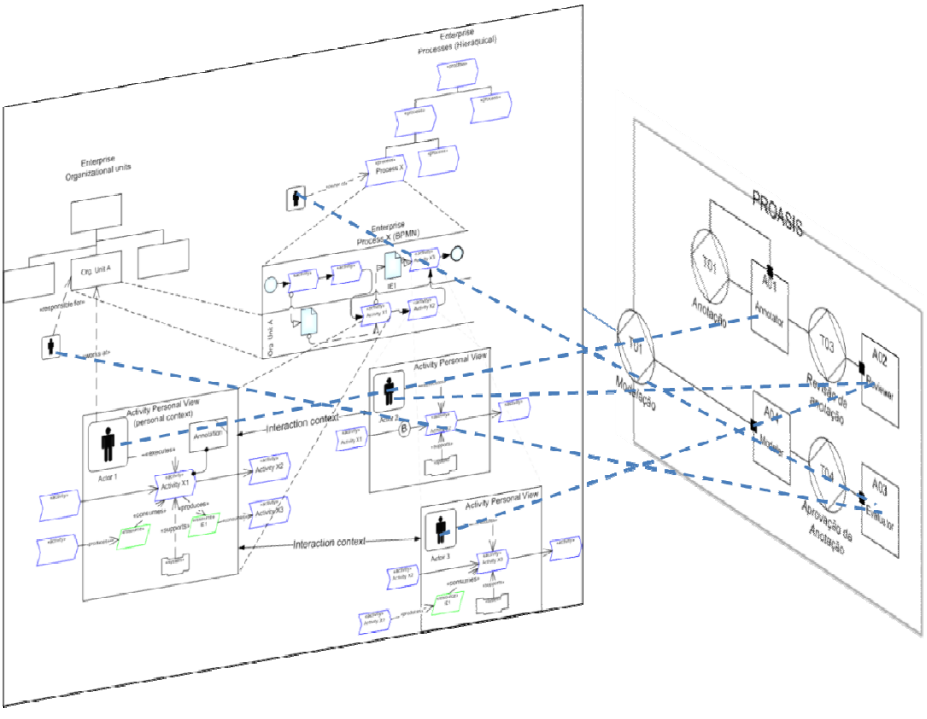


Fig. 9. Relationship between operational model and PROASIS

4 MAPA: PROASIS Supporting Tool

The PROASIS supporting prototype tool, named MAPA (Monitoring and Annotation of Processes and Activities), was defined and developed as a groupware Web based tool that allows the dynamic update of the business process models in a collaborative way, with the following general requirements:

- Annotation editing functions: actors need support to make immediate annotations in the context where the experience occurs. Therefore, an annotation creation, modification and deleting system must be created to be used by organizational actors.
- Different levels of granularity: it should be possible to annotate any object (process, activity, role, resource, relation) in the process model as well as any attribute of each object.
- Selective distribution of diagrams and modeling elements: users only access information that concerns to them, depending on the role played (executor, process owner and organizational unit responsible).
- Access rights: to protect the authors of the annotations, different levels of access rights should be addressed. Only the author of an annotation must be able to delete or modify it.
- Ability to save the entire history of models and their annotations, and the corresponding reviews and evaluations.

- Mapping annotations to entities: is essential to know to what object corresponds each annotation.
- Notification mechanisms: to warn organizational actors about the need to participate in the reviewing and/or evaluation of annotation made and to warn about changes made to the diagrams due to the approved annotations.
- Diagramming capabilities: to allow annotator actors to make graphical annotations with proposed changes to models or to allow modeler actors to directly change the diagrams if the proposals for changing the model was approved.

In developing this platform two versions were created (v1 and v2). MAPA v1 works at the activity detail level. In this version the diagrams are static, so they have to be modeled in a separate tool and then uploaded to MAPA database. Version 2 has emerged as an evolution of version 1. The major implemented change is the direct editing of diagrams. This feature allowed to deploy graphical annotations and to transform the organizational actors in modelers (in this version, any organizational actor can be an active modeler if he plays the modeler role in PROASIS).

This version was designed to operate at the process level of detail, since this was the level to be used in most of the organizations where the MAPA tool was tested. However, the MAPA v2, though not fully implement the activity detail level, allows the annotation of activities and other modeling elements that exists in the process model, either individually or grouped.

The interaction in MAPA v2 allows the direct diagrams handling, so it presents a palette of BPMN modeling artifacts. The interaction at this level can be carried out by annotators who propose changes to the diagrams, and either by the modelers which change the diagrams, updating them. Figure 10 shows the screenshot of the MAPA v2 tool. On this screen, in the left is the process navigation area, in the upper central area is the toolbar (which varies depending on the user's role) and in the center is the drawing area (where the diagrams are presented, and can be changed using the palette of the BPMN modeling artifacts presented in the left of the drawing area).

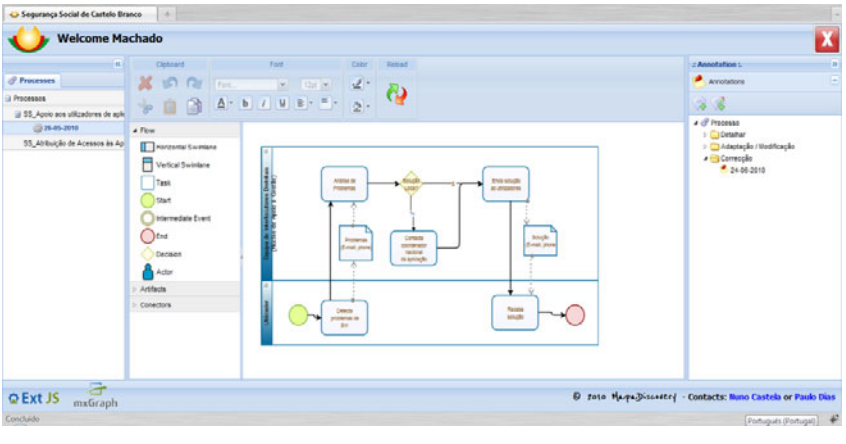


Fig. 10. MAPA v2 screenshot

The interaction with several end users of the case studies organizations provided information to refine the initial requirements of the tool, which lead to the incorporation of new features: incorporation of BPMN specific elements (event types, activity types, etc.); improved diagram visualization in order to allow the comparison of the update proposal diagram (graphic annotation) with the current diagram; improvement of notifications sent by e-mail, indicating the actor who did the action, the annotated process, the annotation/revision/evaluation type and a direct link to the tool (Web site); introduction and improvement of administration functionalities (creation of processes, users, user groups, roles, etc.); introduction of options that enhance the interaction, such as the ability to record a diagram without sending a notification and without generation of history, which allows the incremental recording of diagrams, only sending notification when the diagram is complete.

5 Case Study

The PROASIS and MAPA tool were implemented in several different real organizations in order to test the applicability of the defined process and supporting tool. In this section the case study developed in Social Security Center of Castelo Branco is described.

The Social Security Center of Castelo Branco District (SSCCB) is part of the Social Security Institute. The district centers are the basic organizational and administrative basis of Portuguese Social Security System, responsible for implementing the necessary steps for the development, implementation and management of the benefits under the Social Security System. The organizational goals are timely payment of benefits, combating social exclusion and fraud and supporting workers, families, children, elderly and companies.

In this context and under a perspective of documentation, distribution and analysis of the organizational business processes, it was decided to begin the process of obtaining the information necessary for business process modeling in three organizational units: (1) Financial Management, (2) Administration and Assets and (3) District Interlocutors.

The initial SSCCB business process model was constructed from information gathered from organizational actors, using a methodology that combines the bottom-up and top-down approaches, involving organizational operational actors, who will then use the MAPA tool to maintain the as-is model updated over time.

From the information collected, the organization chart and macro processes hierarchical diagram were built. Both diagrams keep the association of each organizational actor to the organizational units where they work and to the business processes that have activities they execute.

There are currently 19 processes modeled in the tool, belonging to the three organizational units. The activities of these 19 processes are executed by a total of 12 organizational actors. Two of them are responsible of the organizational units.

Figure 16 presents as an example, one of the diagrams that were distributed to capture updates through the annotations, in this context. This figure shows the process “Application Users Support”.

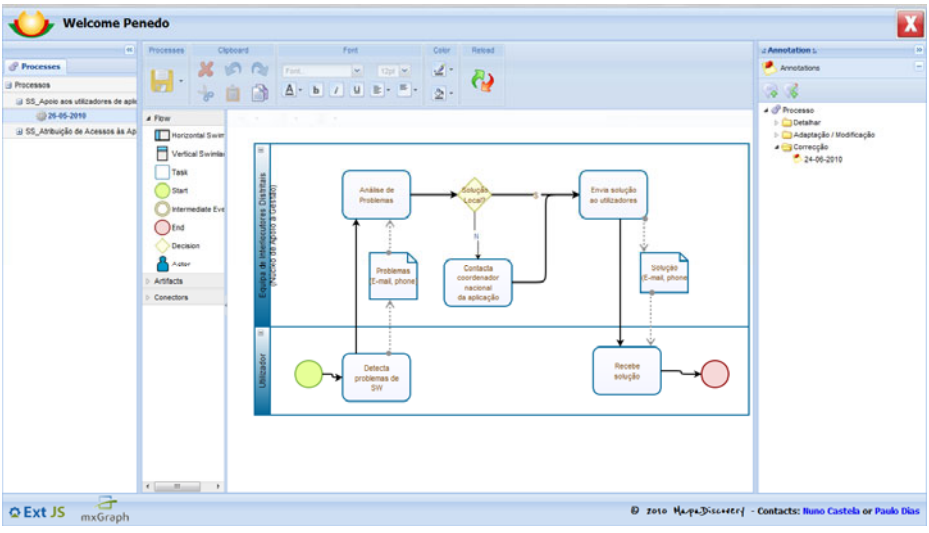


Fig. 11. Process "Application Users Support"

In following figures, a sequence of annotation, review, evaluation and update of the business process diagram "Applications Users Support" is showed. The original version published in the MAPA tool that the actors who belong to this business process (3 executers and 1 process owner) had access, is that one showed in figure 11. In this context, a graphical annotation was made by the organizational actor Proença (figure 12), who have categorized the annotation as a correction and added the following textual description: "This Annotation corrects some flows and create some activities related to EasyVista (support for the resolution of incidents of applications), and turns the notification mechanisms provided by EasyVista visible (including notifications by e-mail)". Figure 13 show the new model diagram proposed in this graphical annotation made directly in the tool based on the diagram published and distributed by MAPA tool.

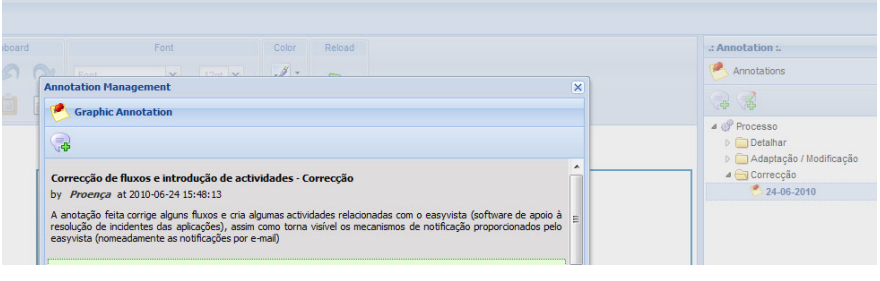


Fig. 12. Annotation made to process "Application User Support"

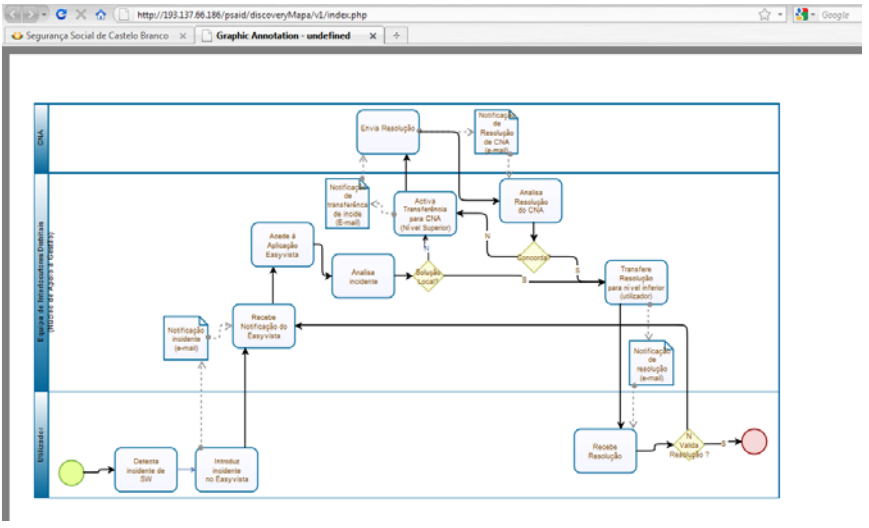


Fig. 13. Alternative model proposed in the scope of annotation made

In figure 14, the reviewing and later evaluation of the annotation described in figures 12 and 13 is made. The annotation had a review made by the organizational actor Domingos, who categorized his review as an agreement and added the following textual description: "I agree, but I think that would be clear if the possibility of the CNA to transfer the problem solving to the II, because we can see, in EasyVista, the state that, in many cases, is in analysis in the II". In figure 14, also can be seen the evaluation made by the organizational actor Penedo, who have approved the annotation and said "The analysis is correct". This approval, together with the other approvals of the annotations made to this version of the diagram shown in figure 11, resulted in a new version of the business process diagram that is shown in figure 15. Note that the full history of the previous version is saved and can be accessed by selecting the process navigator from the left side of the screen.

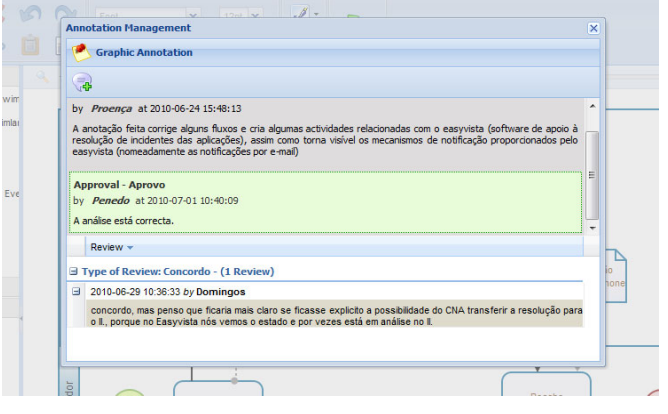


Fig. 14. Reviewing and approval of the annotation made

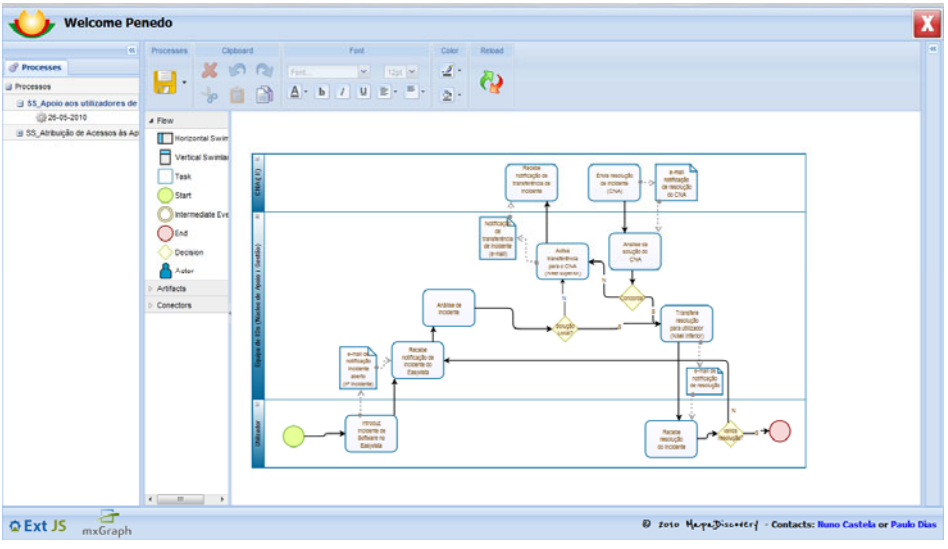


Fig. 15. New Diagram Version of process "Application User Support"

In this case study, which is still ongoing, the first updating cycle with PROASIS using MAPA have started with a total of 19 processes, from which 13 were annotated with a total of 26 annotations (from which 13 were textual annotations and 13 were graphical annotations). There were 37 annotations reviews (from which only one was a no agreement, all others were agreement). The total of annotations evaluated is now in number of 21 (what means that 5 annotations continue with the discussion open). The total number of new versions of the diagrams created due to the notes is 10, meaning that nine of the cases have not yet closed the first cycle of Annotation-review-evaluation-modeling, which culminates with a new version of the diagram initially distributed.

This case study using MAPA to test PROASIS in real organizational environment, has reached its goal: encourage people to discuss their work using a common representation. Also managed to involve the organizational units responsible and processes owners through the approval or rejection of proposals in order to update the model.

There was a great ease in the recognition of the process models within the MAPA tool mainly because the executers and leaders were involved in the initial modeling process. This conclusion drawn from the SSCCB case study is relevant when compared with the initial findings of the Huf Portuguesa case study (a multinational automotive manufacturing company), where the operation of the MAPA tool was initially tried with business processes modeled within the Department of Information Systems, revealing to prove fruitless due to the non recognition of processes by their executors and leaders. In the SSCCB case study, the organizational actors were involved in the modeling process, which began by a bottom-up approach through collecting the individual actions of organizational actors, which validated the abstraction made to define the activities, which in turn were grouped into business processes defined by the organizational responsible by a top-down approach.

The organizational actors showed easiness in interacting with the tool and a great acceptance in using graphical annotations instead of textual annotations. This acceptance is related to the bottom-up methodology used to build the business process model, which directly involved the executers through the gathering of actions, from which the activities were abstracted.

The possibility to assign the modeler role to the process owner provided a considerable improvement in updating processes. This would obviate the possibility of misunderstanding among the purposes of the evaluators and better understanding of what has to be translated to diagram from the evaluations, because both roles could be played by the same organizational actor.

This case study allow, through the cooperation with the users of MAPA v2, to refine some features of the tool and fix some bugs that are normally detected with intensive use.

The interest showed by the national leaders of the SSCCB can be attested trough the decision to extend the case study to all of the district centers of Portugal.

6 Conclusions and Future Work

The tool to support the business process model updating process is currently being used, beyond the case study presented in this paper, in two universities, in a multinational automotive manufacturing company and in a theatre company.

With the available results it is considered that the organizational actors annotated corrections and updates to the model that are usually related to the validation of the diagrams produced, yet allowing the actors directly involved (executers, process owners and heads of organizational units) in the validation (and subsequently update) of the model to try to align it with the reality in an interactive and shared way. By the facts observed in case studies, the annotations, and its extensions (reviews and evaluations) satisfy the requirement of being appropriate mechanisms to support the conversation between the actors and their representation.

The DEMO methodology has revealed appropriated to model the essential meta-process defined in this work and to easily demonstrate what is the dynamics of PROASIS role assignment to actors executing the operational processes in organizations. It was found that the dynamic assignment of roles depends on the context in which the annotation is done at the operational level. This characteristic distinguishes this work from a simple construction of a process and tool to collaboratively update something.

The introduction of MAPA tool in real organizations revealed that it may have an important role, not only in gathering the information needed to update the model (beyond the first important role in validating the model constructed initially), but also because it allowed the opening of a communication channel that encourages the collection and sharing of knowledge about organizational activities. MAPA also demonstrate that actors can play an active modeler role in a collaborative and distributed way.

PROASIS is important in the growth of individual self awareness, because provide explicit representations to the organizational actors that are left with a better sense of what they do and the surrounding context. It was also important to increase the self

awareness of the groups around processes and activities. The organizational self awareness gained from the contribution and explanation of group and individual knowledge through the creation of the historical evolution of the business processes (versions of the process diagrams), which contains all the annotations history (and its negotiation/discussion) that culminated, at certain moments in time, with the proper evolution of the modeled processes, aligning them with their implementation in practice.

Future work, in operational terms, will focus on consolidation of the case studies presented and the refining of PROASIS and MAPA tool.

The refining of PROASIS will focus on better defining the annotations categories, in order to find annotations patterns to improve the update statement made by organizational actors.

MAPA tool prototype will be further developed. The main aim is to provide it with a number of features that enhance the frequency of use by organizational actors. This can be done including links to real artifacts that are required during the execution of processes and activities. This objective will provide access to quality manuals in context (e.g. the work instructions of activities can be accessed), access to user manuals for the computer systems needed to support each activity and access to the documents templates needed in the processes.

References

1. Tribolet, J., Winter, R., Caetano, A.: Editorial Message: Special Track on Organizational Engineering. In: ACM Symposium on Applied Computing, Santa Fe, New Mexico (2005)
2. Schekkerman, J.: How to Survive in the Jungle of Enterprise Architectures Frameworks, 3rd edn. Trafford Publishing, Canada (2006)
3. Caetano, A., Silva, A., Tribolet, J.: Object-Oriented Business Process Modeling with Roles. In: 7th International Conference on Information Systems Implementation Modeling (ISIM 2004), Czech Republic (2004)
4. Bernus, P.: Enterprise models for enterprise architecture and ISO9000:2000. Annual Reviews in Control 27, 211–220 (2003)
5. Davenport, T.H.: Process Innovation: Reengineering Work Through Information Technology. Harvard Business School Press, Boston (1993)
6. Hammer, M., Champy, J.: Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness, New York (1993)
7. Vinturella, J.: Management Briefing Business Process Management (BPM): Must Have? (2005), `http://management-briefings.jbv.com` (published in April 30, 2005)
8. Goul, M., Corral, K.: Enterprise model management and next generation decision support. Decision Support Systems 43, 915–932 (2007)
9. Kim, H.M., Fox, M.S.: Using Enterprise Reference Models for Automated ISO 9000 Compliance Evaluation. In: Proceedings of the 35th Hawaii International Conference on System Sciences (2002)
10. Chapurlat, V.B., Kamsu-Foguema, F.: Prunet: Enterprise model verification and validation: an approach. Annual Reviews in Control 27, 185–197 (2003)
11. Sousa, P., Caetano, A., Vasconcelos, A., Pereira, A., Tribolet, J.: Entreprise Architecture Modeling with Unified Modeling Language. IRM Press (2005)

12. Loucopoulos, P., Kavakli, E.: Enterprise Modelling and the Teleological Approach to Requirements Engineering. International Journal of Intelligent and Cooperative Information Systems (1995)
13. Boudreau, M.-C., Robey, D.: Coping with contradictions in business process re-engineering. Information Technology & People 9(4), 40–57 (1996)
14. Magalhães, R.: Fundamentos da Gestão do Conhecimento Organizacional, Edições Sílabo (2005)
15. Magalhães, R., Tribolet, J.: Engenharia Organizacional: das partes ao todo e do todo às partes na dialéctica entre pessoas e sistemas. In: Costa, S.G., Medeiros, L., Vieira, Rodrigues, J.N. (eds.) Capítulo do livro "Ventos de Mudança", Fundo de Cultura, Brasil (2007)
16. Tribolet, J.: Organizações, Pessoas, Processos e Conhecimento: Da Reificação do Ser Humano como Componente do Conhecimento à "Consciência de Si" Organizacional. In: Amaral, L. (ed.) Sistemas de Informação Organizacionais, Edições Sílabo (2005)
17. Zacarias, M., Pinto, H.S., Magalhães, R., Tribolet, J.: A 'context-aware' and agent-centric perspective for the alignment between individuals and organizations. Information Systems 35(3), 271–374 (2010) (in press)
18. Weick, K.E.: Sensemaking in Organizations. Sage Publications, Thousand Oaks (1995)
19. Zacarias, M., Magalhães, R., Caetano, A., Pinto, H.S., Tribolet, J.: Towards organizational self-awareness: An initial architecture and ontology. In: Rittgen, P. (ed.) Ontologies for Business Interactions, pp. 101–121. IDEA Group, USA (2008)
20. Zacarias, M.: Conceptual Framework based on Agents and Contexts for the Alignment between Individuals and Organizations. Dissertação para obtenção do grau de doutor em engenharia informática e de computadores. IST, Lisbon (2008)
21. EMIPA-SIG: A state-of-the-art analysis by the Special Interest Group (SIG) on Enterprise Modeling: Issues, Problems, and Approaches. In: International Conference on Enterprise Integration Modeling Technology (ICEIMT), South Carolina (June 6-10, 1992)
22. Castela, N., Tribolet, J: Representação As-Is em Engenharia Organizacional. In: 5ª CAPSI, Lisboa, Portugal (Novembro 3-5, 2004)
23. Lankhorst, M.: Enterprise Architecture at Work: Modelling, Communication and Analysis. Spinger, Germany (2005)
24. Zachman, J.: A framework for information system architecture. IBM Systems Journal 26(3), 84–92 (1987)
25. Whitman, L., Ramachandran, K., Ketkar, V.: A Taxonomy of a Living Model of the Enterprise. In: Proceedings of the 2001 Winter Simulation Conference (2001)
26. Fox, M.S., Gruninger, M.: Enterprise Modeling. AI Magazine, 109–121 (Fall 1998)
27. Giaglis, G., Paul, R., Serrano, A.: Reconciliation of Business and systems Modelling Via Discrete Event Simulation. In: Proceedings of the ACM's 1999 Winter Simulation Conferences (1999)
28. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company. Oxford University Press, Oxford (1995)
29. Smith, M.K.: Chris Argyris: theories of action, double-loop learning and organizational learning. The encyclopedia of informal education (2005), http://www.infed.org/thinkers/argyris.htm (last update: January 28, 2005)
30. Matos, M.: Organizational Engeneering, an overview of current perspectives, Dissertação para obtenção de grau de Mestre em Engenharia Informática e de Computadores (2007)
31. Becker-Kornstaedt, U., Reinert, R.: A Concept to Support Process Model Maintenance through Systematic Experience Capture. In: SEKE 2002, Ischia, Italy (July 15-19, 2002)
32. White, S.: Introduction to bpmn. Technical report, IBM Corporation (2004)

33. Dietz, J.L.G., Habing, N.: A Meta Ontology for Organizations. In: OTM Workshops 2004, pp 533–543 (2004)
34. Dietz, J.L.G., Hoogervorst, J.A.P.: Enterprise Ontology and Enterprise Architecture - how to let them evolve into effective complementary notions. GEAO Journal of Enterprise Architecture 2(1) (March 2007)
35. Dietz, J.L.G.: Enterprise Ontology: Theory and Methodology. Springer, Heidelberg (2006)
36. Dietz, J.L.G.: The atoms, molecules and fibers of organizations. Data Knowl. Eng. 47(3), 301–325 (2003)
37. Hatch, M.J., Cunliffe, A.L.: Organization Theory, 2nd edn. Oxford University Press, Oxford (2006)
38. Magalhães, R.: Organizational Knowledge and Technology. Edward, Elgar Publishing, Aldershot (2004)
39. Bohm, D.: Wholeness and the Implicate Order. Routledge, London (1980)
40. Axelrod, R., Cohen, M.D.: Harnessing Complexity. Basic Books, New York (2000)
41. Giddens, A.: The Constitution of Society. University of California Press, Berkeley (1984)
42. Engeström, Y., Miettinen, R., Punamäki, R.: Perspectives on Activity Theory. Cambridge University Press, Cambridge (2005)
43. Heylighen, F.: Self-Organization, Emergence and the Architecture of Complexity. In: Proceedings of the 1st European Conference on System Science, pp. 23–32. AFCET, Paris (1989)
44. Finlayson, J.G.: Habermas. Oxford University Press, Oxford (2005)
45. Stein, L.: Genome annotation: from sequence to biology. Nat. Rev. Genet. 2, 493–503 (2001)
46. Flanagan, D.: Java in a Nutshell. O'Reilly, Sebastopol (2005)
47. Unified Modeling Language Specification, Version 1.4.2, formal/05-04-01 (also available as ISO/IEC 19501) (2005)
48. Oinas-Kukkonen, H.: Towards evaluating knowledge management through the 7C Model. In: Proceedings of the European Conference on Information Technology Evaluation, (ECITE 2005), Turku, Finland (September 29-30, 2005)
49. Yang, S.J.H., Chen, I.Y.L., Shao, N.W.Y.: Ontology Enabled Annotation and Knowledge Management for Collaborative Learning in Virtual Learning Community. Educational Technology & Society 7(4), 70–81 (2004)
50. Marshall, C.C.: Toward an ecology of hypertext annotation. In: Proceedings of ACM Hypertext 1998, Pittsburgh, PA, pp. 40–49 (1998)
51. Borghoff, U., Schlichter, J.: Computer-supported Cooperative Work: Introduction to distributed Applications. Springer, Germany (2000)
52. Gonzalez, P.P., Framinan, J.M.: Tools For Collaborative Business Process Modelling, Business Information Systems: Concepts, Methodologies, Tools and Applications, vol. 4, pp. 636–648. Information Resources Management Association, USA (2010)
53. Ellis, C.A., Gibbs, S.J., Rein, G.L.: Groupware: Some Issues and Experiences. Communications of the ACM 34(1) (1991)
54. Rittgen, P.: Negotiation Models. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 561–573. Springer, Heidelberg (2007)
55. Gazendam, H.W.M., Liu, K.: The evolution of organisational semiotics: A brief review of the contribution of Ronald Stamper. In: Filipe, J., Liu, K. (eds.) Studies in organisational semiotics. Kluwer Academic Publishers, Dordrecht (2005)
56. Castela, N., Zacarias, M., Tribolet, J.: Business Process Model Dynamic Updating, Communications in Computer and Information Science (CCIS). In: Proceedings of the International Conference, CENTERIS 2010, Viana do Castelo, Portugal, vol. 110 Part (2), pp. 158–167. Springer, Heidelberg (2010)

Ontology Construction: Portuguese Air Force Headquarters Domain

Carlos Páscoa[1,2], H. Sofia Pinto[2], and José Tribolet[2,3]

[1] Department of University Education, Portuguese Air Force Academy, Sintra, Portugal
[2] Department of Computer Engineering, Instituto Superior Técnico,
Universidade Técnica de Lisboa, Portugal
[3] CODE - Center for Organizational Design & Engineering, INOV,
Rua Alves Redol 9, Lisboa, Portugal
cjpascoa@gmail.com, sofia.pinto@dei.ist.utl.pt,
jose.tribolet@inesc.pt

Abstract. Capturing knowledge has always been an objective although known to be costly and time consuming. Ontologies, being "an explicit specification of a conceptualization", have tried to capture knowledge through concepts that, in some cases, are used to represent a domain entity, relations between the domain concepts, functions, axioms and instances.

This paper describes an ontology for the Portuguese Air Force Headquarters (*Estado Maior da Força Aérea* - EMFA), reporting also on the ontology building process, its life cycle, applied methodologies, decisions taken and results achieved.

The sources of information, used in the knowledge acquisition phase, consisted mainly of the *EMFA organization* book, the Internet, and text analysis techniques, as well as interviews, brainstorming and cross-validation sessions. Conceptualization consisted on the identification of concepts and classes while building classification trees. The resulting EMFA ontology comprehends six main modules covering the key concepts of the military organization domain. All modules were cross-validated in several meetings and informal competency questions were used in order to verify the usefulness of the ontology.

Keywords: Ontologies, Enterprise Modeling, Enterprise Ontology, Military Organizations, Large Organizations.

1 Introduction

"*An organization begins with a person who has an idea … He or she hires people to do the basic work of the organization … As the organization grows, it acquires intermediate managers … The organization may also find that it needs two kinds of staff personnel … the analysts who design the systems … the support staff, providing services to the rest of the organization …Put these five parts together",* Strategic Apex, Operating Core, Technological Structure and Support Staff, ―*and you have the whole organization …"* [37].

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 83–109, 2011.

The actual dynamics of generating added value to organizations, the nature of the world characterized by completeness, agility and flexibility of the management related instruments and the organization's structure in conjunction with information systems, essential for success, led to the emergence of new disciplines, combining the hard and soft business components.

This applies to the discipline of Organizational Design and Engineering (ODE) which, clustering principles and theories of soft paradigms, such as The Actor Network Theory (ANT) or hard[1] theories such as Systems Theory, advocated the joining of the information systems with organization's strategy that must be fit for the purpose it seeks to achieve.

The discipline of ODE is different from traditional organizational engineering disciplines because it tries to combine knowledge of social sciences with engineering sciences, enabling the design of the social organization component (individuals, groups, values, culture, etc..) combined to rigor associated to the tools of the engineering disciplines [36].

Enter the computer in the organization, as an essential tool to achieving the strategy of the areas of managing their own information and decision support, requires the determination of Architecture Information Systems (AIS), which consists of five sub-architectures [38].

- Enterprise Architecture. Deals with aspects of the organization that are not directly related to the specific business and its operations, such as 'Mission', 'view', 'strategy', and 'organizational goals';
- Business Architecture. Deals with the materialization of business strategy, defined in the , in business processes, representing the objects 'business process' and 'business purpose';
- Information Architecture. Deals with what the organization needs to know to perform the operations, as defined in business processes, characterized in EA and provide an abstraction of the information needs of the organization, regardless of technology; contemplates the objects' feature ',' actor ',' observable state 'and' activity';
- Application Architecture. Deals with the needs of applications in data management and support of business, being independent of the software used to implement the different systems and includes the objects 'component SI' ('block IS') and 'service';
- Technological Architecture. Handles all the technology behind the implementation of the applications as defined in the AP, as well as the necessary infrastructure for the production of support systems, business processes, and considers concepts such as 'IT component' ('IT block') and 'IT services' (IT - Information Technologies).

One of the components architectures deals with business processes that embody the generation of value-added businesses and are decomposed into activities that require a set of resources, human and material, in a defined time, which could contribute to their achievement.

[1] Hard Sciences is an expression that refers to engineering disciplines, while the Soft Sciences expression refers to organizational and management sciences [36].

The inability to run the processes, depending on its nature, leads to impairment of the success of an organization resulting in a set of incalculable damage that may even lead to loss of competitiveness and market exit.

The set of resources required for execution of normal procedures, human resource is constituted by its complexity, as the most specific and most critical and valuable organization. The human resource, given its complex nature is also one that may fail in a more logical response to a request of competence by an activity.

One of the paradigms of the construction of the AIS is the organizational architecture (including the organization's structure). The structure of the organization, like the other elements that define its macro-strategic (e.g. mission, vision, goals and objectives [39]) is to be formed in such a way that can contribute to achieving the desired result, creating added value in business.

Organization design has undergone several mutations according to the various organizational theories that have emerged over time. Some, especially the organizational structures of state administration entities are still heavily marked by theories that advocate principles related to authority and responsibility, strong hierarchy and division of labor.

These principles, however, may not be compatible with new paradigms applied to the world of work, e.g., speed of execution, flexibility, agility and clustering of skills. These principles applied to business processes, by itself, result in more human resources available, with more training and then, with greater ability to solve organizational problems.

The traditional way of organizing public administration, for example, implies the existence of a group of entities arranged in a hierarchical manner that meets a set of principles. When creating views of the organization, it is essential to take into account the structure of it and how it is organized.

Organization view generation includes knowing what one should know, generating the necessary elements to "*view the cockpit of the organization*" [40], in real time and in adverse conditions to ensure the necessary situational awareness to ensure agility and flexibility.

The question arises: **what are the views translated by competence questions that the organization should have that would permit to replace a resource, even with a less degree of competence?**

Enterprise modeling allows the build up of models that can represent the enterprise and enterprise ontology allows capturing knowledge about a domain.

A project developed in 2007 by the Portuguese Air Force aimed at setting the baseline for answering the question and add flexibility to the Organization while running business processes.

The main idea behind the project was to develop an organization (enterprise) ontology for the Estado-Maior da Força Aérea (EMFA) [1], that could store the organizational and tacit knowledge in such a way that it could facilitate to the Air Force all the necessary information in an abstract level, separating Job Positions from individuals and answer most common questions (identified in section 3) that represent views to the organization and providing the answer for the question of competency replacement.

The development methodology used was based on the METHONTHOLOGY [2] that, allowing building ontologies from scratch, encompasses the following phases: specification, knowledge acquisition, conceptualization, integration, implementation and evaluation.

The paper is structured as follows:

- Section 2 introduces the ontology concept, its types, evolution, representation and development methodologies. It also presents related work on the enterprise ontology;
- Section 3 outlines the EMFA structure and describes the way it is organized presenting also some problems in the existing reference document;
- Section 4 describes the current situation, since this project started in 2007;
- Section 5 presents the way ahead, including areas of research and future improvements;
- Section six concludes.

2 Ontologies

Capturing knowledge has always been an objective to organizations, although known to be costly and time consuming [3]. Additionally, successfully acquiring and representing the knowledge for a particular domain does not mean that the knowledge can be reused in a new system [4].

The first ontologies made available were built from scratch. By that time no methodologies or guidelines were available to lead or ease the building process. After some experiences, Gruber [5] introduced some design principles. Gruber's work was the first to describe the role of ontologies in supporting knowledge sharing activities, and presents a set of guidelines for its development. The ontology-building process became clearer, with the continuous development of several other ontologies. As a consequence, the first methodologies for building ontologies appeared in 1995, leading to the emergence of the ontological engineering field [6].

2.1 What is an Ontology?

The term "Ontology", dated circa 1721, in its abstract philosophical notion can be defined as "a branch of metaphysics concerned with the nature and relations of being", "a particular theory about the nature of being or the kinds of things that have existence" [7].

The term "Ontology", in Artificial Intelligence, is used to refer to the shared understanding of some domain of interest which may be used as a unifying framework to identify important underlying concepts, define them, assign terms to them and note their important relationships, improving shared understanding and communication and has the potential to be reused or shared [8]. "*An Ontology is an explicit specification of a conceptualization*" [5] is one definition. There are other definitions. According to Guarino [9], each definition has advantages and disadvantages.

Ontologies provide a formal specification of a knowledge domain [10]. Usually, an ontology is composed of five components [5]: *Concepts*, representing domain entities

(tasks, functions, strategy, etc.); *Relations*, representing a connection between the domain concepts with cardinality n:n; *Functions*, a special case of relations with cardinality n:1; *Axioms*, which represent true statements; *Instances*, that represent domain elements.

2.2 Ontology Types

An ontology can take many forms, from a simple catalog identification (informal) to a theory including general logical constraints (formal) [11].

Informal ontologies may take the form of a controlled vocabulary (a catalog for example), a list of terms together with meanings (a glossary for example) or a list of information terms with associated synonym relationships (a thesaurus for example) while formal ontologies are represented in formal knowledge representation languages with verifying expressiveness, therefore providing a set of more powerful applications.

Simple ontologies are more common, with less cost and provide less completion and interoperability support. On the other hand, structured ontologies are less common, more expensive and provide more completion and interoperability support.

2.3 Building an Ontology

The process of ontology building can be categorized into two main categories: either from scratch or by means of reuse. While the first of the two categories is concerned with building ontologies from scratch using methodologies, reusability is mainly addressed by methodologies. Starting an ontology entails handling two major concerns: language and environment [11].

Since no approved methodology exists, as concluded by Pinto and Martins [6], [12], at the moment "*ontology building is more of a craft than an engineering task*", the process of building an ontology, varies from author to author.

METHONTOLOGY is a structured methodology for building ontologies from scratch, based on the experience gained building the Chemicals Ontologies, that comprehends six phases (specification, conceptualization, formalization, integration, implementation and maintenance), four activities (planification, knowledge acquisition, documentation and evaluation) and states organized into an "*Ontology Life Cycle*" [2]. The evolving prototype life cycle allows the ontologist to go back from any state to a previous one if some definition is missed or wrong. "*So, this life cycle permits the inclusion, removal or modification of definitions anytime of the ontology life cycle. Knowledge acquisition, documentation and evaluation are support activities that are carried out during the majority of these states…*" [2]. The states and activities are detailed below:

- Planning entails knowing its objective and defining a set of steps that will lead to attaining the defined objective;
- Specification includes identifying ontology's purpose, level of formality and scope [2],[14];

- Knowledge Acquisition uses several techniques such as non-structured and structured interviews with experts, informal and formal text analysis. As a result domain components can be identified and represented [2];
- The term *"Formal refers to the fact that an ontology should be machine-readable"* [15]. Therefore the formalization state consists of formalizing the ontology concepts to the desired degree;
- Conceptualization includes "structuring the domain knowledge in a conceptual model (…). The first thing to do is to build a complete Glossary of Terms. Terms include concepts, instances, verbs and properties." [2];
- Integration includes reusing definitions already in other ontologies, identifying them and developing an integration document [2];
- Implementation comprehends using an environment that supports importing and representing the meta-ontology and ontologies selected in the integration phase;
- Documenting is an activity to be done during the whole ontology development process and intends to register ontology development documents in the several stages;
- Evaluation is another important task to be taken into consideration. One important proposal for ontology evaluation is OntoClean [16], which assures an appropriate and consistent hierarchical structure;
- Maintenance involves all the necessary activities to maintain the ontology a living thing including adding, updating and deleting concepts and relations.

2.4 The Enterprise Ontology and Ontology Representation

The Enterprise Ontology is a collection of terms and definitions relevant to business enterprise modeling that can be used as a basis for decision making [13]. The major role of the Enterprise Ontology is to act as a communication medium; in particular, between: different people, including users and developers, across different enterprises and different computational systems [18].

The Enterprise Ontology presents several sections: Meta Ontology, Time, Activity, Plan, Capability, Resource, Organization, Strategy and Marketing comprehending associated concepts and relations between them. The TOVE Enterprise Modeling project [19] aimed at creating the next generation of an Enterprise, a Common Sense Enterprise Model. TOVE authors consider an organization to be a set of constraints on the activities performed by agents.

Ontologies engineering processes begin with the definition of the ontology's requirements; usually in the form of questions that the ontology must be able to answer referred as the competency questions. The second step is to define the terminology of the ontology – its objects, attributes, and relations [19].

There are several languages available to ontology representation: CYCL [22], Ontolingua (http://www.ksl.stanford.edu/software/ontolingua/), F-Logic, CML etc.; Web language standards like: OIL (Ontology Inference Layer) [23], DAML+OIL (DARPA Agent Markup Language) [24], RDF(S) [25], XOL (Ontology Exchange Language), SHOE (Simple HTML Ontology Extensions) [26], XTM (XML Topic Maps) and OWL (Ontology Web Language) [27, 28, 29, 30].

Protegé, from the Stanford's Medical Informatics Group, was the tool chosen to build the EMFA ontology. Stanford University holds a considerable amount of information at http://protege.stanford.edu/ on ontology design and Protegé usage.

3 The EMFA Ontology

The EMFA organization exists to support the Air Force Commander delivering staff studies, policy, guidance and system's requirement definition while assessing the Air Force daily activities and tasks.

EMFA is headed by a general and is composed of five Divisions: Personnel, Intelligence, Operations, Logistics and Planning. Each Division is divided in several branches. Divisions are leaded by a full Colonel and Branches are commanded by a Lieutenant-colonel. Each branch has several staff officers, ranked Lieutenant-colonel or Major, that deal with matters regarding the branch mission. The Organization composed of *Organizational Entities* that are filled by personnel that occupy *Job Position*s. The description of the Organization and related job descriptions was described in the Air Force document RFA 303-2 [1].

The *Organization Entity* has its own attributes. In between parenthesis (1) means a one-to-one relation and (n) a one-to-many relation. The attributes are:

- ***Description*** (1). Organizational Entity Name (example: "*Operations Division*");
- ***Level*** (1). Organizational Entity Level within the Organization (example: "*Two*");
- ***Competences*** (n). *Organizational Entity* set of Competences within the Organization (example: "*To develop and Present the Division Activity Plan*");
- ***Themes*** (n). Organizational Entity themes related to its competences (example: "*Operations*");
- ***Job Position*** (n). Working units with attributes that fill the organizational units in order to do the Job (example: "*Advisor for Operational Research*"). In this case, this field is a unique identification to *Job Position* codes.

Job Position has its own attributes:

- **Code (1)**. Unique identification code (example: "*EM400*")
- ***Description* (1)**. *Job Position* description (example: "*Advisor for Operational Research*");
- ***Activities*** (n). Set of activities expected to be performed by the *Job Position*;
- ***Competence*** (essential) (n)**.** Essential competences required to perform the *Job Position* related activities;
- ***Competence*** (desirable) (n). Essential competences required to perform the *Job Position* (example: "computer skills in document processing);
- ***National security classification*** (1). Required to access National documents needed to *Job Position* related activities ;
- ***NATO security classification*** (1). Required to access NATO documents needed to *Job Position* related activities;
- ***Standard Language Proficiency*** - *English* (1). Required to deal with *Job Position* related activities;

- ***Standard Language Proficiency*** – *French* (1). Required to deal with *Job Position* related activities;
- ***Specialization*** (1). Derivates from the organization of careers and indicates the need of special skills deemed necessary to perform the *Job Position* (example: "Navigator");
- ***Remarks*** (1). Comprehensive information about the *Job Position*;
- ***Initial date*** (1). Initial date that indicates when the *Job Position* was filled by a determined person;
- ***Name*** (1). Name of the person that fulfils the *Job Position*;
- ***Telephone number*** (1);
- ***Theme*** (n); Set of themes related to the *Job Position* (example: "Operations", "Logistics").
- ***Responsible_for*** (1). Indication of *Job Position*s that this *Job Position* has authority upon and therefore is responsible for;

An analysis of the existing information in the EMFA's manual and the way it is organized reveals some problems:

- **Inexistent agreed semantics.** There is no formal agreed terminology. However, tacit knowledge across the organization accepts that there are *Organizational Entities* (that have competencies) and *Job Position*s (that have functions and qualifications – also called competences).
- **Low consistency.** Inexistence of horizontal and vertical verification of activities and competences (the first level organizational entity can have a competency that is not on the immediate level) with repetitions with a different text (for example: "participate in working groups" and "integrate working groups" or "produce reports" and "write reports")
- **Regulations on paper.** Paper regulations are very heavy, with a lot of sheets, little flexibility, hard to read, expensive due to the heavy spending of ink and paper, with high economical and environmental adversities.
- **Inefficiency.** Paper distribution limits the desired dematerialization, essential to smooth flow and process facilitation.
- **Slow access to documents.** Access to regulations is hard and slow.
- *Metrics inexistence*. Inability to know the working hours associated to *Job Position*s.
- *Organization based on people*. Management is based on people instead of *Job Position*s with the consequent inability to count persons.
- *Difficult and expensive upgrades*. Organization changes imply changing manuals, printing and distributing numerous copies.

Although the Air Force has its Information Technology (IT) System with some information about people that are occupying a certain *Job Position*, the EMFA, itself, has no electronic database that can act as a repository of information allowing for rapid query (and quick reference) of some important issues like:

- What are the agreed upon semantics for the Organization? (inexistent);
- What is the representation of the Organization? (drawn on paper);

- Who works for whom? (available on paper, does not answer to matrix groups that are created inside the Organization with specific, time-limited tasks);
- What are the requirements, essential and desirable needed to occupy a *Job Position*? (information exists on paper, however it lacks consistency between *Organizational Entities*);
- What is the relevant information that one has to know about each *Job Position* (like, telephone number, job code, hierarchal dependency)? (inexistent);
- What are the competencies needed to occupy a *Job Position*? (information exists on paper, however it lacks consistency between *Organizational Entities*);
- What are the functions of each *Job Position*? (information exists on paper, however it lacks consistency between *Organizational Entities*);
- What are the subjects that each *Job Position* deals with? (available on the *Job Position* only with a very limited sentence);
- Who deals with a specific subject? (available, but time consuming).

Developing an ontology seemed to be a correct approach to establish an electronic repository of information accessible to all while providing answers to the previous questions.

3.1 Ontology Structure

A military headquarters structure is not familiar to common people. However the author has experience acquired from working as staff officer for a considerable number of years. The domain knowledge was taken from other staff officers. To better plan the project, the first step was a brainstorm session among the officers with the objective of building a general snapshot view of the domain. Six key areas were identified: *Organizational Entity*, *Job Position*, *Staff Officer* and *Content* which comprehends four auxiliary modules (*Theme*, *Competence - Essential*, *Competence - Desirable* and *Activity*), *Library* and *Organization Structure*.

Organization is divided into two main areas: i) the *Organizational Entity* (with its own attributes) and *Job Position* (also with their own attributes). The *Organizational Entity* represents the organizational working units and the *Job Position* fills the *Organizational Units*. Some important requirements were identified.

- The organization has to be inserted in the ontology and preferably displayed by the use of a graphical interface;
- People should be categorized displaying relevant information such as: *Job Position*, branch, essential and desirable competences, job tasks, telephone number, leadership position and identification of subordinates (by *Job Position*),

All modules were cross-validated in several meetings and informal competency questions were used in order to check the usefulness of the ontology. These questions addressed specifically each one of the previously identified areas.

The enterprise ontology [18] was used as a base for some of the attributes, both in the designation and the explanation of semantics (for example, "*organizational unit*"). The first action was to agree on a glossary of terms. Table 1 shows an example.

Table 1. Dictionary of terms (sample)

Term	Definition	Source
Air Force	Military organization that has the primary responsibility for conducting air warfare. The air force must gain control of the air, support ground forces (e.g., by attacking enemy ground forces), and accomplish strategic-bombing objectives. Its basic weapons platforms are fighters, bombers, attack aircraft and early warning and control aircraft.	http://www.answers.com/topic/air-force
Division	An organizational part of a headquarters that handles military matters of a particular nature, such as personnel, intelligence, plans, and training, or supply and evacuation	DOD Dictionary of Military Terms
Senior Officer	Military ranks: Major, Lieutenant-Colonel and Colonel	Portuguese Law
Organization	The central concept in this section is that of an ORGANISATIONAL UNIT, the main structural element of an organization	Extracted from the Enterprise Ontology
LEGAL ENTITY	(which includes a PERSON, CORPORATION etc.) and a MACHINE, all of which themselves may correspond to a single OU	Extracted from the Enterprise Ontology
PERSON	a human being (for the purposes of this Ontology, PERSONS are of interest for their capacity to play various Actor Roles in an enterprise (e.g. perform ACTIVITIES).	Extracted from the Enterprise Ontology
Organizational Unit (OU)	An Entity [with a defined identity] for MANAGING the performance of ACTIVITIES to ACHIEVE one or more PURPOSES. An OU may be characterized by: • the nature of its PURPOSE(S); • one or more PERSONS working for the OU; • RESOURCES allocated to the OU; • other OUs that MANAGE or are MANAGED-BY the OU; • its ASSETS; • its STAKEHOLDERS; • being LEGALLY OWNED; • its MARKET (if it is a VENDOR)	Extracted from the Enterprise Ontology

Some of the competency questions identified are:

- ***Job Positions*.** What is the telephone number for *Job Position* P? What activities is *Job Position* P responsible for? Which *Job Position* needs rank R?
- ***Organizational Entities*.** Who works on the *Organizational Entity* O? Who is the boss of *Job Position* P?
- ***Themes*.** Whose *Job Positions* handle theme T? Whose *Job Position* handles theme T but not theme A?
- ***Competences*.** What are the *Job Positions* that need *Competence* C?

3.2 Ontology Development

This section presents the EMFA ontology development process defining its objective, requirements, structure and other related and relevant information. The base line was setup using the concepts presented in the Enterprise ontology by reuse. We further extended and instantiated it.

The development of the EMFA ontology followed METHONTOLOGY and was strongly influenced by the ideas presented by [14]. It is also important to mention the study presented by [6], as a reference to understand how methodologies have evolved and to identify and comprehend the key steps in ontology development and reuse.

The next paragraphs describe the activities related to specification, knowledge acquisition, conceptualization, integration, implementation and evaluation phases that compose METHONTOLOGY.

Specification Phase. The objectives of the EMFA ontology are:

- Agree on a common semantic (the meaning of concepts) across the organization;
- Store all the relevant information about EMFA in a convenient manner allowing elimination of redundant data while providing appropriate query capabilities and a more efficient way of working;
- Improve consistency by introducing horizontal and vertical activity verification that allows finding competence redundancies;
- Attain better efficiency by reducing costs, eliminating heavy hardcopies and long distribution channels and creating new essential flexible and friendly-environment mechanisms that allow fast access to documents;
- Avoid the organization based on people by defining the *Organizational Entities and Job Position* attributes and instantiating the concept to cover all the Organization.

The following text illustrates the overall requirements specification, according to the requirements specified and defined in [2].

Ontology EMFA **Requirements Specification Document**	
Domain:	Military Organization (EMFA)
Date:	April, 4th 2007
Conceptualized by:	Carlos Páscoa
Implemented by:	Carlos Páscoa
Purpose:	Ontology about the Air Force Headquarters (HQ) organization to be used when information about hierarchy and thematic issues is required
Level of Formality:	Informal
Scope:	List of all the HQ elements, hierarchy definition and thematic issues handled by each one. List of Organizational Entities, list of Job Positions. List of thematic areas. At least information about: hierarchy and organizational *Job Position* (filled by staff officers) properties: function, rank, area of knowledge, Standard Language Proficiency requirements (French and English) and National and NATO security classification level.
Class:	Human Resources, Intelligence, Operations, Logistics and Planning
Source of knowledge:	Portuguese Air Force Manual 303-2 (2000).

Fig. 1. EMFA Ontology overall requirements specification

Knowledge Acquisition phase. The knowledge acquisition activity for the EMFA ontology was mainly performed by the author using its own experience as a staff officer and was done simultaneously with requirements specification. Within areas not so familiar to the author, knowledge acquisition was done by questioning other staff officers within the EMFA organization. Firstly, the author spent time on becoming acquainted with the document "*Air Staff Headquarters Organization*". An extensive period of discussion with other Staff officers was extremely important, since it allowed clarification of some very important concepts about the military organization.

The RFA 302-1 (B) [1] was the main knowledge source and defined the ontology extension and amount of detail. Other papers like the Enterprise Ontology [18] and an overview of enterprise architectures [20], among others, were used to clarify concepts. The JP 1.02 [21] was used for disambiguating between terms. Typically, the term with the biggest number of occurrences was selected. There was also an enormous discussion on the meaning of all the keywords included in defining the organization which led to the creation of the Air Force "*Glossary of Terms and Definitions*".

Conceptualization phase. During conceptualization the acquired knowledge was structured into modules, each corresponding to different domain areas. The main activities performed in the development of each module were (i) identification of concepts and their properties; (ii) classification of groups of concepts in classification trees; (iii) description of properties; (iv) identification of instances; (v) description of instances.

During this phase, the previously mentioned discussion and validation sessions were also used to: (1) identify the relations between classification trees; (2) debate the conceptualization as a concept or instance of ambivalent notions; (3) harmonize the identified properties and their definitions. Although METHONTOLOGY was used for the global ontology development, during conceptualization the concept development stages concept identification, relationship identification, properties identification, instance definition and validation were performed. Figure 2 presents the conceptual Model.

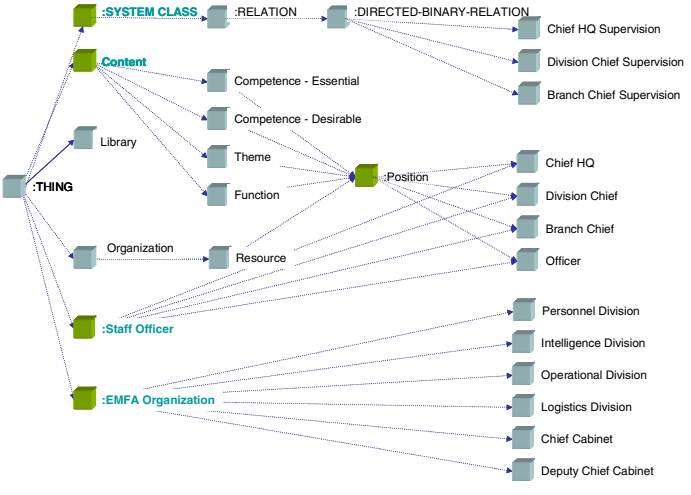


Fig. 2. EMFA Ontology concept

The knowledge structure found on the knowledge sources was divided into six main modules covering key concepts of the military organization domain. In Protegé the THING Class is divided into the following concepts:

- **EMFA Organization.** The *Organizational Units* existing in EMFA were grouped in "*HQ Organization*". The class is merely used to represent *Organizational Entities*. At this stage a decision was made towards labeling the several *Organizational Entities* relating them to the *Job Position* code as follows:
  - "EM00?", *Chief HQ* and *Deputy Chief HQ*;
  - "EM1??", *Personnel Division*;
  - "EM2??", *Intelligence Division*;
  - "EM3??", *Operations Division*;
  - "EM4??", *Logistics Division*.
- **Staff Officer.** Attached to the *Organizational Entity* and acting as a connector between the *Organizational Entity* and the *Job Position* is the *Staff Officer*. This class is actually the one that represents the hierarchy and also links the *Organization* class to the *Resources*, a super Class that is at *Staff Officer* level (due to the inexistence of material resources in the manual). Each *Staff Officer* has an attribute that defines his position in the hierarchy and who are the officers that he is responsible for. This is obtained by creating the "*responsible_for*" relation that allows for admin or hierarchic supervision. At this stage a decision was taken to label the several *Staff Officer* relating them to the *Job Position* code as follows:
  - "EM00?", *Chief HQ*;
  - "EM?00", *Division Chief*;
  - "EM??0", *Branch Chief*, being EM??0 starting at 1 for "Branch" order;
  - "EM??0", Officer inside each Branch, being "EM???" starting at 1 for the "Officer" order.
- ***Organization and :SYSTEM CLASS.*** The *Organization* class is tied to the ***:SYSTEM CLASS*** which is used to draw the hierarchy tree using the "responsible_for" and "supervision_type" relations. One of the Protegé add-ons, the Graph Widget allows for the creation and representation of hierarchy charts. A hierarchy tree was built representing the *Chief HQ*, *Division Chief* and *Branch Chief* classes.
- ***Library.*** The *Library* class is not used since, at the initial stage, it was decided not to proceed with the documentation issues.
- ***Resource.*** The definition of properties for each Class derived from the requirements and the information that one should obtain with reasoning. The *Job Position* class, a subset of *Resource*, represents the more complex one as it holds

most of the data to be queried. The *Job Position* concepts include *Function*, *Competence - essential* and *Competence - desirable*.

- ***Content.*** The *Content* class holds four important sub-classes. The four classes represent most of EMFA's domain ontology and its instantiations were one of the most challenging tasks during the whole ontology development process:
  - ***Competence - essential***, defines the mandatory skills that the *Job Position* occupant should possess;
  - ***Competence - desirable***, defines the optional skills that the *Job Position* occupant should possess;
  - ***Theme***, defines the areas of interest inside the EMFA;
  - ***Function***, defines as activities performed inside EMFA that are specific or common to each *Job Position*.

Figure 3 presents the EMFA ontology conceptual model showing the main concepts and the number of instances for each.

THING
- **SYSTEM-CLASS**
  - **RELATION**
    - **Directed-binary-relation**
      - Chief HQ Supervision Relation (9)
      - Division Chief Supervision Relation (1)
      - Branch Shief Supervision Relation (1)
  - **Staff Organization**
    - Chief HQ,
    - Deputy Chief HQ,
    - Personnel Division,
    - Intelligence Division,
    - Operations Division,
    - Logistics Division
  - **Staff Officer**
    - Chief HQ (2),
    - Division Chief (5),
    - Branch Chief (13),
    - Officer (43)
  - **Content**
    - Tema (35)
    - Competence - Desirable (49),
    - Function (274),
    - Competence - Essential (1)
    - Organization (1)
  - **Resource**
    - **Job Position**
      - Chief HQ (2),
      - Division Chief (5),
      - Branch Chief (13),
      - Officer (43)

Fig. 3. EMFA Ontology concept number of instances

A partial description of classes can be found in Table 2.

Table 2. EMFA's ontology specification

Name	Documentation	Value Type	Allowed Classes	Cardinality	Default	Domain
Employee_relationship	Relation type between two classes	Instance	Chief HQ & Division Chief supervision relations	Multiple	-	Organization
Organization	EMFA representation	Instance	Organization	Multiple		Library
Organization Issues		Instance		Multiple		Library
Organizational Entity	Organizational Entity being represented	String		At most 1		EMFA Organization & Job Position
National security classification	Required to access National documents needed to Job Position related activities	Symbol	Unclassified Restricted Confidential Secret	At least 1	Secret	Job Position
NATO security classification	Required to access NATO documents needed to Job Position related activities.	Symbol	Unclassified Restricted Confidential Secret	At least 1	Secret	Job Position
Code	Unique identification code	String		At most 1		Chief Supervision Relation, Staff Officer, Theme, Competences, Function, Organization & Resource
Initial date	Initial date	String		At most 1		Job Position
Description	Description	String		At most 1		Job Position
Function	Set of functions to be performed	Instance	Function	Multiple		Job Position
Competence - desirable	Desirable competences required	Instance	Competence - desirable	Multiple		Job Position
Competence - essential	Essential competences required	Instance	Competence - essential	Multiple		Job Position
Name	Office's name	String		At most 1		Job Position
Other Information	Additional information not available elsewhere	String		At most 1		Job Position
Rank	In the military hierarchy	String		At most 1		Job Position

Table 2. (*continued*)

Specialization	Derivates from the organization of careers and indicates the need of special skills deemed necessary to perform	String		Required At least 1 At most 1		Job Position
SLP English	Standard Language Proficiency for English. Four digits from "0" to "5".	String		At most 1	2222	Job Position
SLP French	Standard Language Proficiency for French. Four digits from "0" to "5".	String		At most 1	3333	Job Position
Phone	Phone of Job Position	String		At most 1		Job Position
Theme	Related themes	Instance	Tema	Multiple		Job Position
Quality	Supervision relation Quality (Good, Satisfactory, Bad)	Symbol	Tema	At most 1		Chief HQ Supervision Relation
Responsible_for	Responsibility relation between different levels as defined in "RELATION" and "Organization"	Instance	Job Position	Supervision type.	Multiple	Chief HQ, Division Chief, Branch Chief
supervision_type	Supervision type (administrative, hierarchic)	Symbol		At most 1		Chief HQ Supervision Relation
Text	Free text	String		At most 1		

Integration phase. There was little integration effort since most of the ontology was built from the enterprise ontology. During this phase, some ideas were taken from the *newspaper* sample ontology published by the Protegé Group. Relationships like the "*responsible_for*" or "*supervision_type*" were used as an example.

Similarly, as the Graph widget demo within the *newspaper* ontology uses the *Organization* Class to graphically represent the hierarchy tree, the same idea was ported into the EMFA ontology in order to represent its hierarchy tree, which was a essential tool to visualize the complete ontology.

The *Library* Class was also imported from the newspaper ontology allowing future work to expand and include actions that deal with material resources (example: producing a memo "M").

Implementation Phase. The Implementation phase was the hardest working phase since the ontology model has to be populated with information which was done using Protégé (Figure 5 – page 18 – shows an example).

Populating *Staff Officer* was easy because it was only necessary to instantiate the several classes according to the organization structure.

Populating classes, like *Competence - Desirable* and *Function* was an arduous task since repetitions of the same information were found repeatedly through out the insertion process. As an example, normalizing the several *Function* statements brought the number of instances down from 429 to 274. Sample statements included "*Participate on Working Groups*" and "*Integrate Working Groups*" which mean exactly the same.

The same happened to *Competence - Desirable* which was brought down from 76 to 49 instances.

Consistency between the different levels of the organization were also checked and implemented in the ontology. *Competence - Desirable* descriptions, for example, had to be created or modified to increase consistency.

At this phase some of the definitions had to be refined and completed with missing information.

Evaluation Phase. In the evaluation phase, compliance of the model to the requirements was tested by finding: i) if the ontology answers to a set of the questions defined by Fox et al. [19]; ii) by evaluating some competency questions and the results provided by questioning the ontology.

Taking into consideration that the EMFA manual and correspondent ontology does not define actions, material resources, roles[2] (associated with goals, processes, authority, skills, policies and resources) for the agents represented by the *Staff Officer*, on the questions defined by Fox et al [19] the following discussion can be applied:

Structure Competency:

- **What role does an agent play?** A Staff Officer has a set of well defined functions that performs under a set of themes. If not considering the full definition of role one can conclude that the ontology includes information about authority (who is responsible for whom), skill (in the form of essential and desirable qualifications) and resources (limited to other human resources that are below in the hierarchy).
- **Which division does the agent belong to?** The answer to this question is explicit in the ontology.
- **Who must the agent communicate with?** Although there is no explicit communication channels defined and since only EMFA is represented, one can assume that a particular agent can communicate with all the other agents, especially the ones that share the same *Themes* and the ones that are directly above or below in the chain of command.
- **What kind of information does the agent communicate?** The kind of information is not explicit but it can be derived from the *Themes* (operations, intelligence for example) and *Functions*.

[2] A role defines one or more prototypical job functions in an organization [19]. Each role is associated with: i) **Goals** (one or several goals that the role is intended to achieve); ii) **Processes** (activity networks that have been defined to achieve the goals); iii) **Authority** (adequate authority needed for the role to achieve its goals including the right of using resource, the right to perform activities, and the right to execute status changing actions; iv) **Skills** (required for the realization of the job functions); v) **Policies** (constraints on the performance of the role's processes unique to the organization role); vi) **Resource** (one or more resources may be allocated to a role for disposition under its authority).

- **Whom does the agent report to?** As the hierarchy is well defined, the ontology can answer this question.
- **Is a role a generalization of another role?** If a direct comparison is done between the functions of the *Job Positions* there is some room for deriving if a particular *Staff Officer* functions are generalizations of a "*Branch Chief*" functions.

Behavior Competency:

- **What are the goals of the organization?** There is no direct answer to the question as organization goals are not stored.
- **What are the goals of a role?** There is no direct answer to the question as role goals are not stored. However, Skills are included as each instance has a set of essential and desired competences.
- **What are the goals of person X?** At the Air Force level, goals are defined in an early document and derive from permanent and non permanent tasks. The EMFA structure includes the Activities each *Job Position* has to realize in order to attain yearly defined goals.
- **What activities are available for a role to achieve its goal?** The existing set of functions for each Job Position answers that question.
- **What non-human resources are available to achieve a goal?** Since there are no material resources included in the ontology, this question cannot be answered and represents an improvement area.

Authority, Empowerment and Commitment Competency:

- **What resources does the person have authority to assign**? Every direct subordinate defined in the hierarchy.
- **In order to perform a particular activity, whose permission is needed**? The EMFA structure grants persons to execute their assigned activities without further permission. However if coordination between several persons is needed, higher authority permission should be granted.
- **What activities may a person execute under his own authority**? All the activities included in his *Job Position*.

Evaluating the ontology by posing some general and some competency questions and the results provided by questioning the ontology were divided in two sections. The first section on generic questions and the second section on the competency questions defined previously. The generic questions and answers are presented below:

- **What is the agreed upon semantics for the organization?** As a result of brainstorming among the organization's departments the EMFA has an agreed semantics for describing the organization.
- **What is the representation of the organization?** There is one instance named "*EMFA HQ*" to represent the EMFA hierarchy tree. The graphical representation output (limited to the third level) is shown in Figure 4.

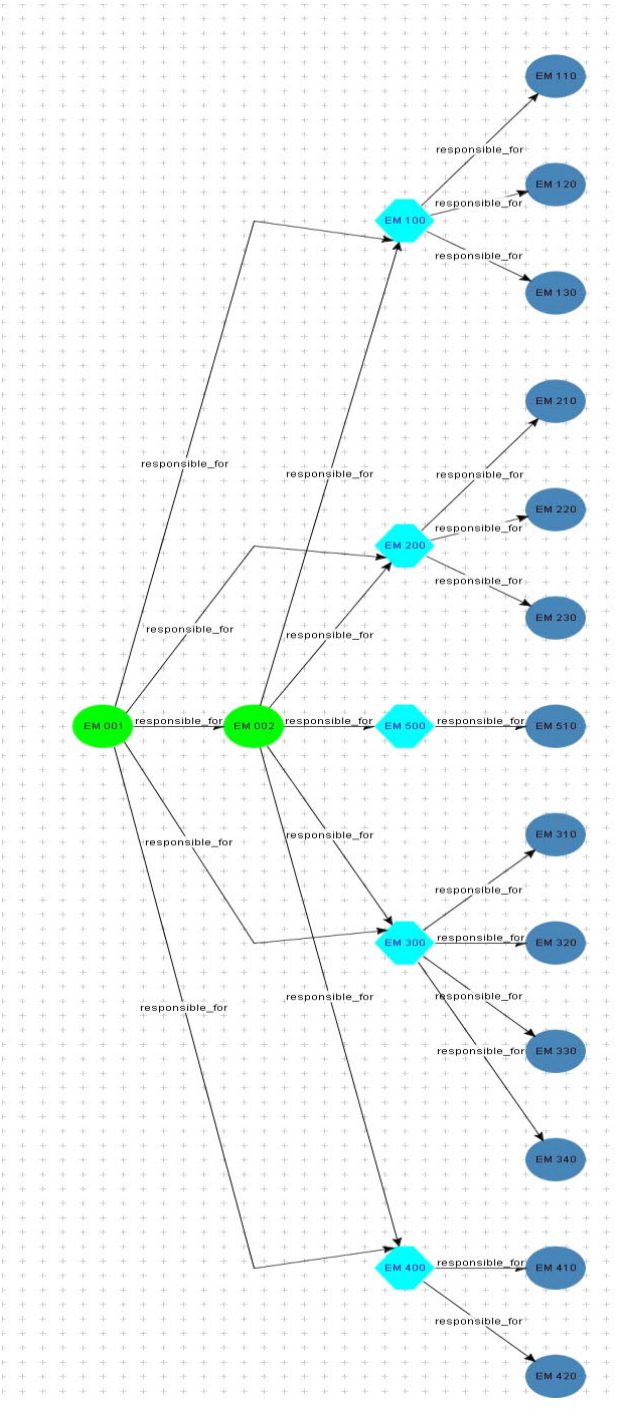


Fig. 4. EMFA hierarchy up to the third level

- **Who works for whom?** The EMFA's hierarchy is embedded in the ontology and is not shown on this paper due to space constraint.
- **What are the essential or desirable requirements needed to occupy a particular *Job Position*?** These elements are present in the ontology and can be queried for any Job Position.
- **What is the relevant information that one has to know about each *Job Position* (like, telephone number, job code, hierarchal dependency)?** These elements are included in the ontology and can be queried for any *Job Position*.
- **What are the competences needed to occupy a *Job Position*?** These elements are included in the ontology and can be queried for any *Job Position*.
- **What are the functions (activities) of each *Job Position*?** These elements are included in the ontology and can be queried for any *Job Position*.
- **What are the theme that each *Job Position* deals with?** These elements are included in the ontology and can be queried for any *Job Position*.
- **Who deals with a specific subject?** These elements are included in the ontology and can be queried resulting on a definite number of *Job Positions*.

The competency questions represent a more complex stage of ontology answering because questions include "AND" or "OR" clauses. Some answers like, knowing the telephone number for a particular *Job Position* implies a query on the *Job Position* and then a double click to obtain the detail. Sample queries that returned the right answers are presented below:

- ***Job Positions*** (Figure 5)**:** What is the telephone number for *Job Position* P? Whose *Functions* is *Job Position* P responsible for?

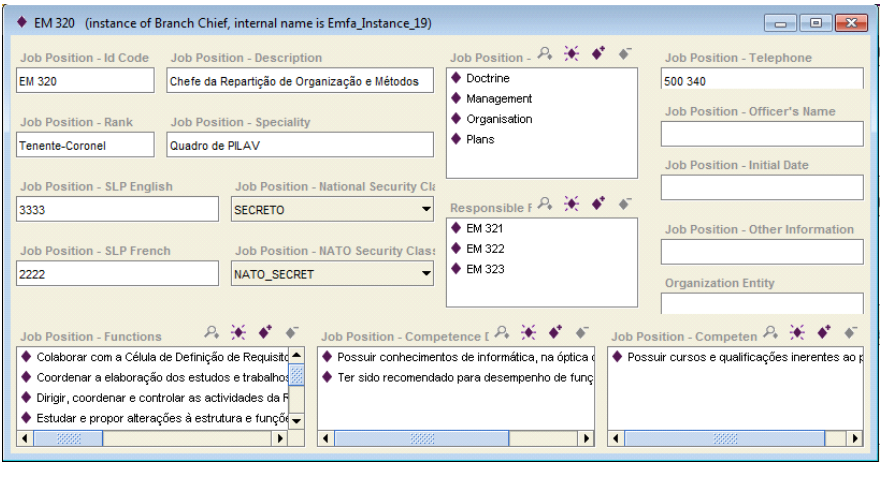


Fig. 5. Example of an answer to the *Job Position* competency questions

- **Organization:** Who works on the *Organizational Entity* O? The query to the question showing every *Job Position* that works for "EM 21" ("External Relations Branch") returned four items which was the right answer.

- **Themes:** Whose *Job Positions* handle *Theme* T? – Competency Question IV; Whose *Job Positions* handle *Theme* T but not *Theme* A? – Competency Question V.
- **Competences:** What are the *Job Position*s that need *Competence* C? The query was made to find out whose *Job Positions* needed "*Having training and experience in Intelligence areas*". – Competency Question VI
- **Library:** Who is the boss of *Job Position* P? The answer to that question can be seen querying first for the EM??? Code and than see what is the boss (example: querying for *Job Position* "EM 213" turns out that the boss is *Job Position* "EM 210") – Competency Question VII.

Upon completion of the validation phase, the existing ontology was tested for clarity, consistency and responsiveness by some of the co-workers which made comments and helped to improve the final product. As a result of such tests, new competency questions were created and semantics was improved. The resulting ontology shows that it is possible to actually have an EMFA knowledge base that it is organized and provides answers to most organizational (general) and competence questions.

Discussions that were carried prior and during the ontology development concluded that the document dealing with defining and describing the organization had to be altered to include and define actions (like producing a Memo or coordinating a particular subject between entities), material resources (in the way that if the *Job Position* is an abstract entity, rather than a person, computer types and software needs, for example, can be assigned to *Job Position*), roles (including Processes, Authority, Skills, Policies and Resource) for the *Job Positions* and *Staff Officer*.

4 Current Situation

After the initial work was accomplished, in 2007, there was a decision to modify the Information Systems (IS) in the Air Force and to wide the Ontology concept to all the Organization as the Air Force was moving towards the design of a new concept for its IS systems.

In 2008, the system was updated to allow for statistics development based on the number of functions attached to each *Job Position*. The result drew some interesting conclusions in what related to *Job Position* distribution and workload.

An EMFA ontology prototype was designed in a relational database system at the Air Force Academy in 2009, serving as the basis for the building of a comprehensive requirement specification document which was completed early in 2010. This was done side by side with the new IS analysis done by the IT Directorate and has been completed early in 2010.

In fact, today, the EMFA possesses an ontology, in two formats, that allows for competency questions formulation and answer. This has not been extended to the remaining entities since it is expected to be part of the new IS.

Unfortunately, due to financial constraints, the new IS will not be put into production until, at least, 2014. In the mean time the concept was being put into practice by the Organization officers at the Air Force Headquarters.

At the present time, all the Air Force Organizational Entities and related *Job Position*s (where around 8300 person work) have been identified and all put on paper. An effort was made to produce the hardcopy of all the macro structure organization documents which will be signed by the Air Force Chief-of-Staff in April 2011. The second level organization documents will be ready to be signed by the end of September 2011. All this documents are available in the Organization Intranet as a mean to improve organizational self-awareness.

A three year hardcopy revision process has been approved until all the work is dematerialized into the new IS.

The concept application allowed increasing coherence and consistency in the organizational field and caused that everyone in the Air Force has a well defined *Job Position* knowing what the required *competences* are and which *functions* they need to perform.

As to the problems encountered before the ontology development (see page 6), the current situation is the following (for better comprehension the situation development is graded "*Improved*", "*Partially Improved*" and "*Not Improved*"):

- Inexistent agreed semantics. There is a formal agreed terminology accepted through out the Organization. *Improved*.
- Low consistency. The consistency and coherence was greatly improved by the use of matrices all across the Organization. *Improved*.
- Regulations on paper. Although it was not possible (except for the EMFA) to extend the ontology to the remaining Organization, the concept was applied manually. An original hardcopy exists for each document and the distribution is done through the Intranet in Electronic Format. *Partially Improved*.
- Inefficiency. The existence of a single point of entry for originals in the Intranet allows for obtaining the latest version. However, if a change reflects on several documents, each will have to be updated. *Partially Improved*.
- Slow access to documents. Access to regulations now fast through the Intranet. However searching for a *Job Position* implies knowing the document where it is defined. *Partially Improved*.
- Metrics inexistence. Inability to know the working hours associated to *Job Position*s is still continuing (except for EMFA). *Not Improved*.
- Organization based on people. The concept has change and is now based on organizational entities and *Job Position*. However, inexistence of a software tool is still causing inability to count persons. *Not Improved*.
- Difficult and expensive upgrades. Once the document distribution is done, in electronic format, through the Intranet, considerable savings in paper and toners were accomplished. *Partially Improved*.

5 Future Work

Future work necessarily includes dematerialization with the correspondent insertion in a software tool part of the new generation IS.

As actions, material resources and roles are not present today in the EMFA ontology, future work recommends including these items in order to make it more complete and consistent.

Standardization of competences and functions is also a matter that needs further work. Aligning the organizational entities competences is also of great importance because it permits eliminating redundancies and allowing for separation of functions between the strategic, tactic and operational organization levels and also for building the necessary metrics.

Grouping competences in core competences (like individual competences, team competences and operating competences for instances) and grouping those in profiles is also an area of interest that will allow coaching and grooming of persons from a earlier stage in the organization passing careful planned *Job Position*s routings.

Core competences and profiles, once defined, would also allow to developing a graph of *Job Position*s at different levels of the organization and estimate the cost of education for a particular person, in a determined area of interest.

On the scientific field it would be interesting to make an approach of this work with the principles defined in the DEMO [31]. DEMO is a methodology for the (re)design and (re)engineering for organizations [32] that has its theoretical origins in enterprise ontology and is perceived as a model for describing and understanding the enterprise construction and operation, being fully independent of the way the enterprise is implemented.

According to [31] the enterprise ontology should be coherent, comprehensive, consistent and concise and it should only abstract its essence. A number of reasons can be mentioned to justify this approach to DEMO [33].

- None of the enterprise modeling techniques purpose a reduction of complexity as high as the one purposed by DEMO (over 90%). This topic is directly related with the concise and essence focus qualities of DEMO;
- Most of these modeling techniques are not based in a strong well-formed theory. DEMO methodology is based on a rigorous theory: the psy-theory which combines the knowledge from ontological works, language/action perspective, logic and systems theories. This stands for coherence and consistence of the DEMO models;
- DEMO clearly defines three notions that are considered relevant in governing the enterprise dynamics (competence, authority and responsibility). Most of these notions are absent or not clear defined in others enterprise modeling techniques;
- DEMO has been widely accepted in both scientific research and practical appliance;
- DEMO practical application has been successful validated in several enterprises[3]. An extensive ten year study executed with 28 projects concluded that DEMO is a good method for the fast (re)design of organizations [34].
- Demo has been used as a base for formalizing enterprise architecture and governance and for formalizing the splitting and allying of enterprises [35].

6 Conclusion

In order to add flexibility to the organization and to avoid losses caused by failure of competences a question was formulated: **what are the views translated by**

[3] DEMO Pratical Case Studies and Publications (DEMO, 2010)

competence questions that the organization should have that would permit to replace a resource, even with a less degree of competence?

To answer the question the Portuguese Air Force initiated a project that used ontologies to set up the necessary views that would enable answering to the question.

Being "an explicit specification of a conceptualization" [5] Ontologies represent a shared understanding of some domain of interest which may be used as a unifying framework to identify important underlying concepts, define them, assign terms to them and identify their important relationships.

The spectrum of Ontologies varies from catalog identification (informal) to a logical theory (formal) [11].

Although no standard approved methodology exists, the process of developing an Ontology, Fernandez at al [2] proposed a structured methodology for building Ontologies called METHONTOLOGY that comprehends several phases and encompasses the following steps: purpose identification, domain knowledge capturing, specification, conceptualization, formalization, integration, implementation and evaluating.

Related to enterprise organization ontologies, the Enterprise Ontology [18] is a collection of terms and definitions relevant to business enterprises modeling the enterprise in an organization wide-view manner that can be used as a basis for decision making acting as a communication medium between different people and systems.

There are several languages available for ontology representation. Protegé, from the Stanford University Group, is one of the available that allows the representation in OWL and other formal languages.

Using Protégé, not only it was required to lay down the basic information for the ontology construction, but also to start the steep curve of learning the software tool in a manner that could achieve the objective.

The *Estado-Maior da Força Aérea* organization exists to support the Air Force Commander delivering staff studies, policy, guidance and requirements definition while conducting thorough assessments of the Air Force daily activities and tasks. The enterprise ontology domain seemed the best reference for developing the ontology here presented.

Following METHONTOLOGY as a baseline (since not all the steps were implemented), which proved very adequate, and the document RFA 303-2 *Headquarters Organization* [1] as the primary source of information, a discussion was held among the staff personnel to generate common understanding and to identify concepts and groups of concepts while building classification trees. The knowledge model was formalized in Protégé, which was also used, together with some of the *plug-ins*, to automatically generate the ontology code.

The resulting EMFA ontology was formalized in Protégé and comprehends six main modules covering the present and future key concepts of the military organization domain – *EMFA Organization*, *Resource*, *Staff Officer* and *Content*, which comprehends four auxiliary modules (*Theme*, *Competence - essential*, *Competence - desirable* and *Function*), *Library* and *Organization*. All modules were cross-validated in several meetings and informal competency questions were used in order to check the usefulness of the ontology.

Within the development, several shortfalls were identified, starting with lack of common semantics. This required some corrections, namely the introduction of the concept of "abstract" *Job Position* which helped to save resources in configuring the position for a new person and the introduction of themes which can help in finding out who is dealing with a particular subject. Another improvement was the refinement of existing information with the elimination of redundancies and verification of consistency within the several levels of the organization.

As actions, material resources and roles are not present today in the EMFA manual, future work recommends including these items in order to make it more complete and consistent.

However, the resulting ontology does answer a set of general and competency questions which improved considerably EMFA's situation awareness. The ability to make questions and get answers about an oriented set of questions, using the EMFA ontology developed in Protegé, has proven to be a very valuable tool.

Unfortunately, installing Protegé in each computer is not a feasible solution. In order to proceed with the ontology development at the Air Force level a creation of a relational database, using the same concepts outlined in this document and the "future work" ideas was proposed and accepted by the Air Force Chief of Staff. The project is called "*Online Organization Ontology*", it encompasses all Air Force departments.

The concept was further instrumented by the Air Force Academy that developed software in a relational database management system. Due to lack of financing to implement the concept in the new IS, the work has been undertaken manually.

Now, document distribution is done, in electronic format, through the Intranet, which represents considerable savings in paper and toners and also eased document consultation.

Besides discussing the competency questions validation, section "Current situation" discusses the outcome of the ontology on a set of general questions.

The answer to the initial formulated question is on the competency questions. Each type enables a view to the organization that allows replacing a competence, required by an activity on a business process, in almost real time.

Future work includes investigating new ways of grouping competences and inserting the concept of profile. Reformulating the study in the view of DEMO methodology is also a perspective for future developments on this issue.

References

1. Portuguese Air Force Headquarters. Organização do Estado-Maior da Força Aérea RFA 303-2, Portuguese Air Force, Lisboa, Portugal (2000)
2. Fernandez, M., Gomez-Perez, A., Juristo, N.: METHONTOLOGY: from Ontological Art towards Ontological Engineering. In: Proceedings of the AAAI 1997 Spring Symposium Series on Ontological Engineering, Stanford, USA, pp. 33–40 (March 1997)
3. Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., Swartout, W.R.: Enabling Technology for Knowledge Sharing. AI Magazine 12(3), 36–56 (1991)
4. Swartout, W.R., Neches, R., Patil, R.: Knowledge Sharing: Prospects and Challenges. In: Fuchi, K., Yokoi, T. (eds.) Knowledge Building and Knowledge Sharing, pp. 102–109. IOS Press, Amsterdam (1994)

5. Gruber, T.: Towards Principles for the Design of Ontologies for Knowledge Sharing. International Journal of Human Computer Studies 43(5/6), 907–928 (1995)
6. Pinto, M.J.P.: Ontologies: How can they be Built. Knowledge and Information Systems 6(4), 441–464 (2004)
7. Merriam-Webster, "Ontology" word In Merriam-Webster Dictionary (2008), http://www.m-w.com/dictionary/ontology (accessed in 2008-02-01)
8. Uschold, M., Gruninger, M.: Ontologies: principles, methods and applications. Journal of Knowledge Engineering Review 11(2), 93–155 (1996)
9. Guarino, N.: Understanding, Building, and Using Ontologies. In: Proceedings of the 10th Knowledge Acquisition for Knowledge-Based Systems Workshop, Alberta, Canada (1996)
10. Benjamins, V.R., Gomez-Perez, A.: Overview of Knowledge, Sharing and Reuse Components: Ontologies and Problem-Solving Methods. In: Proceedings of the IJCAI 1999 Workshop on Ontologies and Problem-Solving Methods (KRR5), Stockholm, Sweden, pp. 1.1–1.15 (1999)
11. McGuinness, D.: Ontologies Come of Age. In: Fensel, D., Hendler, J., Lieberman, H., Wahlster, W. (eds.) Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press, Cambridge (2003)
12. Pinto, H.S., Martins, J.P.: Some issues on ontology integration. In: Proceedings of the Workshop on Ontologies and Problem Solving Methods during IJCAI 1999, Stockholm, Sweden (1999)
13. Uschold, M., King, M.: Towards a Methodology for Building Ontologies. In: Proc. of IJCAI 1995 Workshop on Basic Ontological Issues in Knowledge Sharing (1995)
14. Lopez, M.F., Gomez-Perez, A.: Building a chemical ontology using Methontology and the ontology design environment. IEEE Intelligent Systems 14(1), 37–46 (1999)
15. Studer, R., Benjamins, V.R., Fensel, D.: Knowledge engineering: principles and methods. IEEE Transactions on Data and Knowledge Engineering 25(1-2), 161–197 (1998)
16. Guarino, N., Welty, C.A.: An overview of OntoClean. In: Staab, S., Studer, R. (eds.) Handbook on Ontologies in Inf. Sys., pp. 151–172. Springer, Heidelberg (2004)
17. Guha, R.V., Lenat, D.B.: CYC: A Mid-Term Report. AI Magazine 11(3), 32–59 (1990)
18. Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology (1995), http://citeseer.ist.psu.edu/uschold95enterprise.html
19. Fox, M.S., Barbuceanu, M., Gruninger, M., Lin, J.: An Organization Ontology for Enterprise Modeling, Enterprise Integration Laboratory - Department of Mechanical and Industrial Engineering, University of Toronto (1997)
20. Zacarias, M.: Enterprise Architectures: An Overview. INESC, Lisboa (2007)
21. Department of Defense. Dictionary of Military and Associated Terms JP 1.02, US DOD, Washington (2007)
22. Guha, R.V., Lenat, D.B.: CYC: A Mid-Term Report. AI Magazine 11(3), 32–59 (1990)
23. Horrocks, I., Fensel, D., Broekstra, J., Decker, S., Erdmann, M., Goble1, C., van Harmelen, F., Klein, M., Staab, S., Studer, R., Motta, E.: The Ontology Inference Layer OIL (2000)
24. Harp, B., Aberg, P., Benjamin, D., Neches, R., Szekely, P.: DRAMA: An Application of a Logistics Shell. In: Proceedings of the Annual Conference on Artificial Intelligence Applications for Military Logistics, pp. 146–151. American Defense Preparedness Association, Williamsburg (1991)
25. Brickley, D., Guha, R. V.: RDF Vocabulary Description Language 1.0: RDF Schema, 2004, at http://www.w3.org/TR/rdf-schema/ accessed in 2010-06-06.

26. Luke, S., Heflin, J.: SHOE 1.01 Proposed Specification (2000), http://www.cs.umd.edu/projects/plus/SHOE/spec.html (accesses in 2010-06-06)
27. Patel-Schneider, P.F., Hayes, P., Horrocks, L.: OWL Web Ontology Language; Semantics and Abstract Syntax (2004), http://www.w3.org/TR/owl-semantics/ (accessed in 2010-06-06)
28. Gómez-Pérez, A., Fernández-López, M., Corcho, O.: Ontological Engineering. Springer, Heidelberg (2003)
29. Gomez-Perez, A., Corcho, Ó.: Ontology Languages for the Semantic Web. IEEE Intelligent Systems 17(1), 54–60 (2002)
30. Su, X., Ilebrekke, L.: A Comparative Study of Ontology Languages and Tools. In: 14th International Conference on Advanced Information Systems Engineering, pp. 761–765. Springer, Heidelberg (2002)
31. Dietz, J.L.G.: Enterprise Ontology: Theory and Methodology. Springer, Delft (2006)
32. Aveiro, D.: GOD theory for organizational engineering: continuously modeling the (re)Generation, Operationalization and Discontinuation of the Enterprise, Doctoral dissertation, Department of Computer Science and Engineering, Technical University of Lisboa, Instituto Superior Técnico, Lisboa (2010)
33. Henriques, M.: Enterprise Governance and DEMO. Master Thesis, Department of Computer Science and Engineering, Technical University of Lisboa, Instituto Superior Técnico, Lisboa (2010)
34. Mulder, J.: Rapid Enterprise Design. Tech. rept. PhD thesis, Delft University of Technology (2006)
35. Op't Land, M., Proper, E., Waage, M.: Enterprise Architecture: Creating Value by Informed Governance. Springer, Heidelberg (2009)
36. Matos, M.G.: Organizational Engineering: An Overview of Current Perspectives, Master Thesis, Department of Computer Science and Engineering, Technical University of Lisboa, Instituto Superior Técnico, Lisboa (2007)
37. Mintzberg, H.: Organizations Design: Fashion or Fit. Harvard Business Review (January–February 1981)
38. Abreu, M., Tribolet, J.: Considerations about measuring soft factors in organizations. Paper, Department of Computer Science and Engineering, Technical University of Lisboa, Instituto Superior Técnico, Lisboa (2008)
39. Business Rules Group: The Business Motivation Model: Business Governance in a Volatile World, Verion 1.3 (2007), http://www.businessrulesgroup.org/bmm.shtml (consulted October 2009)
40. Tribolet, J.: Sistemas de Informação Organizacionais. In: Amaral, L. (ed.) Organizações, Pessoas, Processos e Conhecimento: Da Reificação do Ser Humano como Componente do Conhecimento à "Consciência de Si" Organizacional, Edições Sílabo (November 2005)

Exploring Normalized Systems Potential for Dutch MoD's Agility

(A Proof of Concept on Flexibility, Time-to-Market, Productivity and Quality)

Martin Op 't Land[1,2,3], Marien R. Krouwel[1], Edward van Dipten[4], and Jan Verelst[5]

[1] Capgemini Netherlands, P.O. Box 2575, 3500 GN Utrecht, The Netherlands
{Martin.OptLand, Marien.Krouwel}@capgemini.com
[2] Delft University of Technology, Delft, The Netherlands
[3] Technical University of Lisbon, Lisbon, Portugal
[4] Command and Control Support Centre, Defense Material Organization, Ministry of Defense, Bernardkazerne, Amersfoort, The Netherlands
EG.v.Dipten.01@mindef.nl
[5] Department of Management Information Systems, University of Antwerp, Prinsstraat 13, B-2000 Antwerp, Belgium
Jan.Verelst@ua.ac.be

Abstract. Both the Command and Control Support Centre (C2SC) of the Dutch Ministry of Defense, and Capgemini are constantly exploring better ways of building and maintaining information systems which effectively support strategy and operations of an enterprise. In a Proof of Concept conducted in March and April 2011, together with University of Antwerp, they evaluated the possibilities of the Normalized Systems approach for building information systems. In 50 man days not only a system was built that would normally be built in 100-150 man days, also a major technology change had taken place (in week 2 Cocoon and EJB2 were replaced by Struts and EJB3) and changes in the (data) models were instantly processed, even in the last week of development. With these results, Normalized Systems is considered a key enabler in rapid enterprise transformation. Moreover, because Normalized Systems makes it possible to have a continuous link from enterprise (process) modeling to software development and a short feedback loop from system development back to enterprise modeling, it is also considered a key enabler for agile enterprise engineering. This will also enable shortening the typical cycles of enterprise governance.

Keywords: DEMO, Normalized Systems, Agile Enterprise Engineering, Ministry of Defense, case study, SCRUM.

1 Situation

In today's dynamic environment, competitive success depends on being able to quickly respond to changing market, economic and regulatory conditions. Because every enterprise currently is supported by IT, enterprise IT has to be

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 110–121, 2011.

able to quickly respond and change directions to support critical strategic initiatives and requirements. The increased required flexibility as well as the ever increasing complexity of both enterprises and supporting IT make that software development and maintenance budgets have to grow every year, as was already stated by Lehman in 1980 [1]: "As an evolving program is continuously changed, its complexity, reflecting deteriorating structure, increases unless work is done to maintain or reduce it." Enterprises that decrease — or (even) keep constant — the IT budget will be faced with less satisfactory IT, decreased support of organizational changes, decreased business-IT alignment and decreased *situational awareness*. There is a definite need for a software delivery approach with which high quality IT solutions can be developed and continuously changed quickly.

As an answer to this need and the challenges, University of Antwerp developed the Normalized Systems (NS) theory [2] on *evolvable complexity*. The ambition of the NS concepts and its associated tools, called *expanders*, is to create transparency and controllability in the development and maintenance of software and business processes, ultimately leading to productivity improvements, comparable with the effects of industrialization in other sectors. The theory prescribes a fine-grained modular structure such that one type of anticipated change only impacts one module in the information system. This results in manageable and evolvable complexity; for common maintenance actions it is formally proven that the impact on the system is the same when it is performed now, two years ago, or two years from now when the system has grown and become more complex. Over the last years, the Normalized Systems approach and tools have been used in 12 real-life projects in Belgium, creating a similar number of information systems of which four are mission-critical. University of Antwerp now wanted to share their knowledge and to receive feedback on NS and its supporting tools in a robust way. Such a controlled NS adoption would include applying intake and audit sessions for each real-life project.

As a global leader in consulting, technology, and outsourcing, Capgemini is always interested to collaborate on innovations with potential to create business value; a reason to actively participate in, e.g., the international CIAO! Research Network[1]. When University of Antwerp end of 2009 first explained the concepts of NS to the CIAO! network, Capgemini decided to start exploring the potential of Normalized Systems [3], in combination with the (earlier) proven [4,5] business agility delivered by using the Design & Engineering Methodology for Organizations (DEMO[2]) [6]. In fall 2010, Capgemini division Custom Software Development (CSD) recognized that NS could increase productivity, flexibility, quality, and maintainability, and could also decrease the time-to-market of IT solutions, which fits in their ambition to make projects highly industrialized and predictable. Capgemini decided to adopt NS, to educate a team in NS, and to start seeking for an opportunity to see hands-on (1) to what extent NS already offers the expected improvement, and (2) what still can be improved on the NS

[1] http://www.ciaonetwork.org/
[2] http://www.demo.nl/

approach and tools. A dialogue between Capgemini and the Dutch Ministry of Defense (MoD) was started.

The Command and Control Support Centre (C2SC) of the Dutch MoD builds mission critical systems for land based operations. In doing so, it acts as supplier for other parties within the Ministry of Defense. Often, COTS and MOTS solutions are integrated with self developed software solutions. In that way, they are also demand party for external vendors. Since every operation differs in terms of people involved and their information and communication needs, C2SC must be able to adapt their existing applications and services or create new ones at high speed, both for current and future operations. In order to do so, C2SC is always looking for new ways to build information systems which (1) offer the freedom to be adapted to changing needs, (2) can be realized quickly, offering a short time to market, and (3) provide high quality, as bugs should be prevented and solved before the operation of the system and not during the operation.

2 Tasks

Both C2SC and Capgemini were curious to test the claims of NS in their own practical situation. Therefore a Proof of Concept (PoC) was conducted, a collaboration between C2SC and Capgemini, supported by University of Antwerp, in which the added value of Normalized Systems in terms of flexibility, time-to-market, productivity, and quality was evaluated, and experience in building a Normalized System was gained, finding indicators for improvement on the Normalized Systems approach and tools. In order to perform the evaluation, an application was build to support the domain of Information Management.

This domain of information management, earlier modeled by C2SC in an enterprise ontology according to DEMO (Figs. 1 and 2) operates as follows. For a successful execution of military operations it is of utmost importance that information products (reports, order, plans) are available valid and on time to planners, decision makers and executors. Easiest is if everyone involved in a military operation has access to all information. However, because of confidentiality but also because of limited available means for information exchange, for every single deployment the information exchange requirements must be determined:

- who offers which information (products)? – InformationProductOffer (IPO);
- who needs which information (products) from whom? – InformationExchangeRequirement(IER);
- which mean (medium) is required to transfer the information?
- which limitations in confidentiality apply for an information exchange?

The application should demonstrate NS capabilities in four main areas:

A1. CRUDS (Create, Read, Update, Delete, Search) screens;
A2. custom screens;
A3. workflow;
A4. export/visualization.

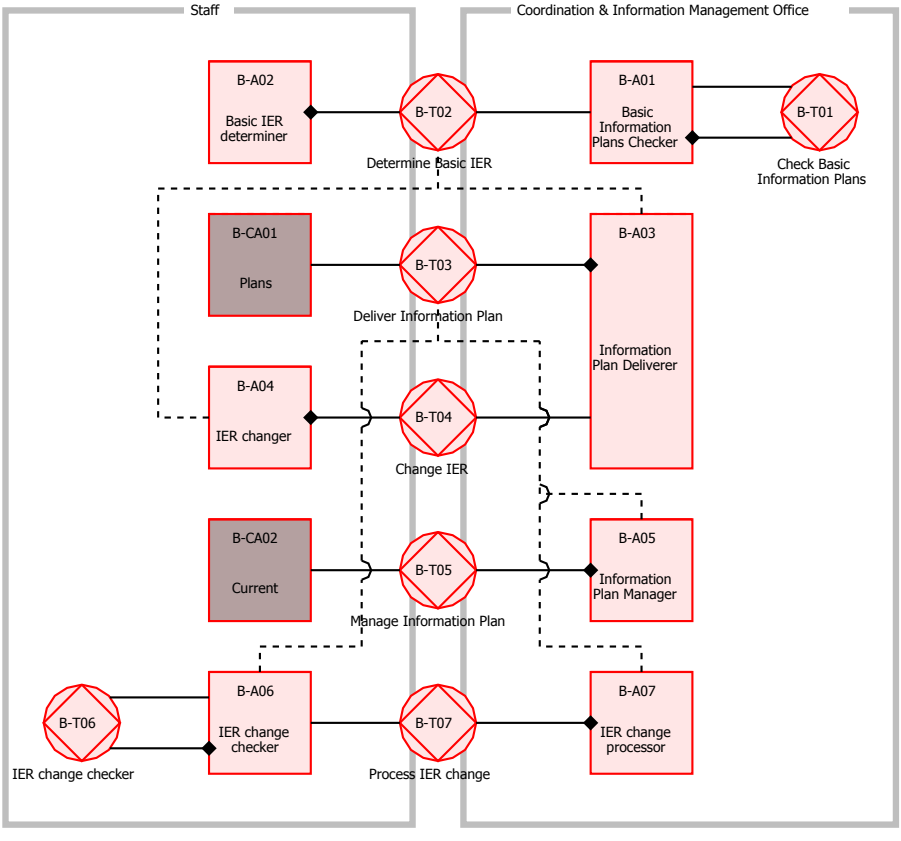


Fig. 1. Information Management MoD: DEMO Construction Model (input version)

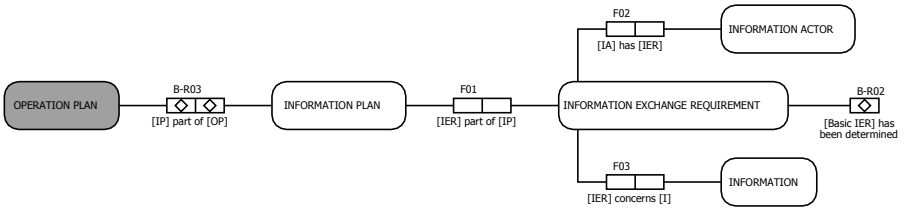


Fig. 2. Information Management MoD: DEMO Fact Model (input version)

Table 1. Planned activities and budget (man days)

Activity	C2SC	Capgemini	UA
Business Modeling	2	1	
Requirements specification	4	9	
Architecture and Design	2	6	
Development	0	8	
Testing	2	7	
Deployment	2	2	
Project Management		7	
Report NS benefits and concerns	2	6	
List of requirements for improvements		2	
Intake / audit / technical support			2
Normalized System training			2
Total	**14**	**48**	**4**

Next to an application, a report about the findings on using the Normalized Systems approach and tools should be delivered. Finally, at the end of the project, University of Antwerp would audit the created application.

Table 1 shows the planned activities and budget in terms of man days expressed in RUP terminology. System development of the project was scheduled from March 16thuntil April 13th, 2011. Deadline for the final report of the findings was scheduled for May 1st (Fig. 3). The developers from Capgemini that worked on this project were only recently trained in the Normalized Systems theory and had no experience with the supporting toolset yet.

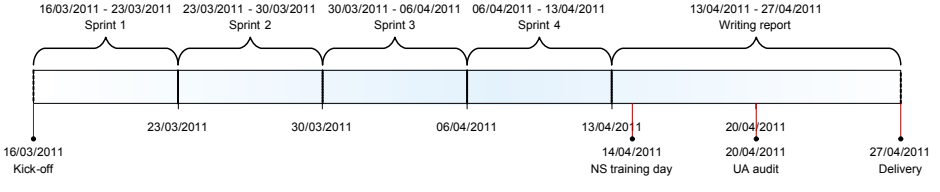


Fig. 3. Timeline

3 Approach

The development part of the PoC was executed with an Agile/SCRUM[3] approach with four *sprint*s of one week (Fig. 4). Every sprint started with a planning session in which the goals, in terms of user stories, for the sprint were defined. At the end of every sprint the realized application was reviewed by subject matter experts of C2SC.

[3] http://www.scrumalliance.org/

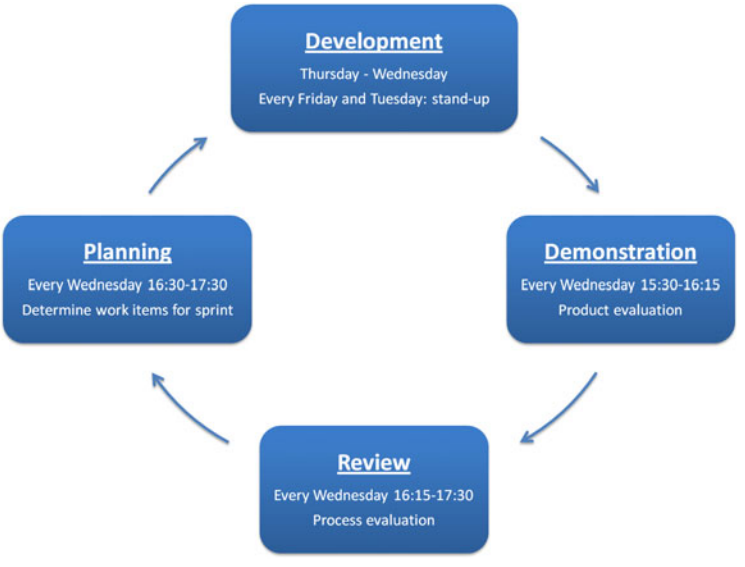


Fig. 4. Sprint activities

Input for the first sprint comprised a DEMO Construction Model (an organization implementation model of a chain of actors, delivering each other services) (Fig. 1) and a DEMO Fact Model (a semantic model of facts created by actors; used as a non-normalized data model) (Fig. 2), complemented by examples of the current IT implementation (Excel sheets). Already on the first day it was discovered the models were incomplete and partly inconsistent with the current operations of Information Management. As soon as the models were clarified, the data model was implemented instantly by the Normalized Systems tools, providing a first application with mainly CRUDS screens (A1) and components for user and profile management. In the second week, custom screens (A2), screens that were not provided by the Normalized Systems tool set, were realized. Because of the clear interfaces the toolset provides, it was easy to build these screens. The NS workflow (A3) mechanism was tested by realizing a basic workflow with which an actor was able to view a list of IERs and accept or reject them. Setting up the workflow turned out to be straightforward as well. An export functionality (A4) was realized to export a set of IERs to XML. Because of the clear interfaces provided by the tool set, it was easy to integrate external packages for the mapping to XML.

In every demonstration session it was discovered the application did not meet the real needs of C2SC yet; often the models had to be adapted and re-implemented in the following sprint. However, changes in the data model were implemented as easily as the first time, without impacting the rest of the application. The impact on custom screens brought about by the changes in the data model proved to be small and traceable. Even changes in the models, made in the last week (see Fig. 6 for the final — normalized — data model), did not

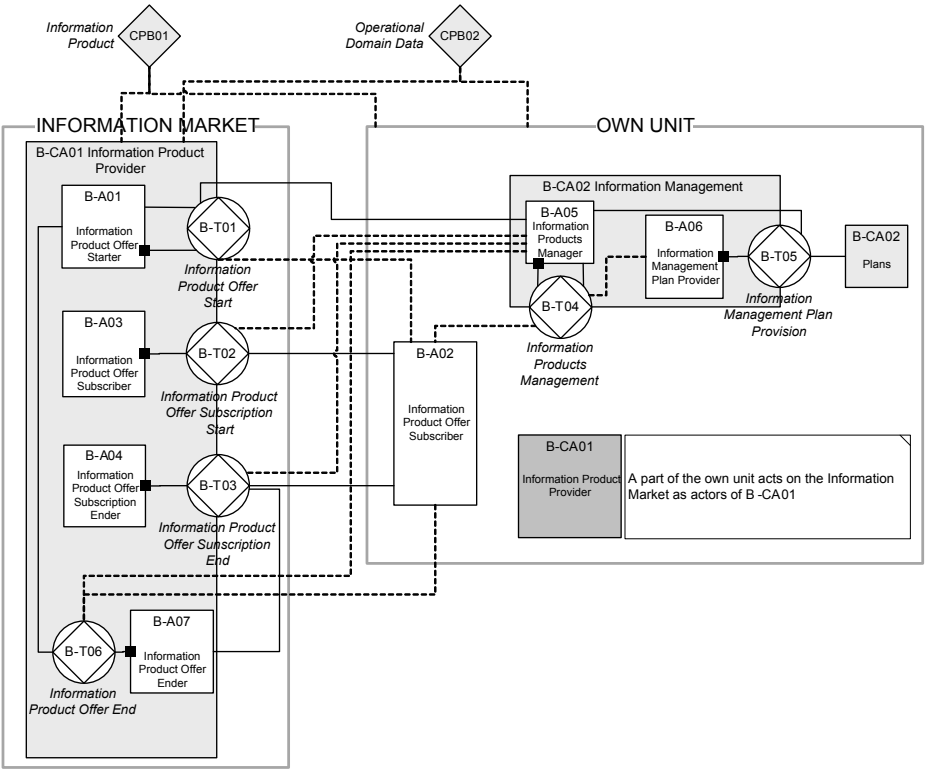


Fig. 5. Information Management MoD: DEMO Construction Model (final version)

impact the realized application with its custom screens, workflow, and export functionality.

Another field explored during this PoC was the migration of the application to another technology stack: within one day the application was migrated from Cocoon/EJB2 to Struts/EJB3, something that usually would not be done during a project, or would have cost at least a man month.

During the project, University of Antwerp supported the Capgemini team of developers in using the tool set for three Fridays, for them also a way to receive feedback on the tool set. At the end of the project UA performed an audit on the realized application, double checking the application is built according to the Normalized Systems principles. They confirmed the application was indeed built according to the Normalized Systems principles, the customizations are well maintainable, and the application thus is able to evolve. Also, UA provided a training on Normalized Systems for some C2SC decision makers in order for them to understand the results of the audit and PoC, and also to enable them starting to visualize the organizational and personnel impact of a — at that moment: hypothetical — Normalized Systems adoption.

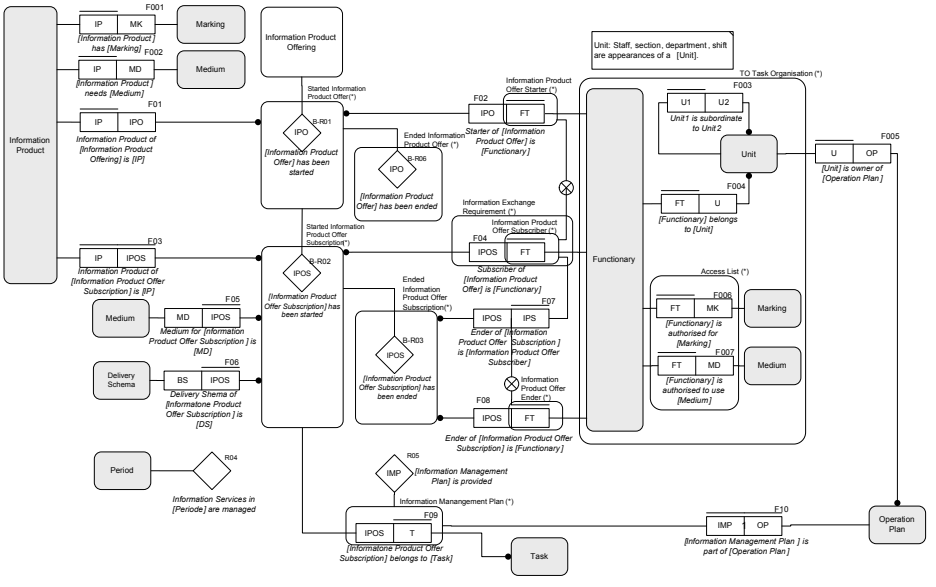


Fig. 6. Information Management MoD: DEMO Fact Model (final version)

In order to measure the productivity gain, both C2SC and Capgemini Accelerated Delivery Center (ADC) performed a NESMA[4] Function Point[5] Analysis (FPA); both parties agreed upon approximately 260 function points. Based on their FPA and their own best practices, both parties provided a time estimation (Table 2).

Table 2. Realized activities by Capgemini (man days)

Activity		**NS (planned)**	**NS (realized)**	**Capgemini (estimated)**	**C2SC (estimated)**
Business Modeling		1	1		
Requirements specification		9	2.5	7	0
Architecture and Design		6	3	34.5	42
Construction	Expanding	4	2.5	44.5	43.5
	Customizations	4	12		
Testing	Unit	2	1	5	5
	Other	5	1	27.5	10.5
Deployment		2	0	3	1
Project Management		7	17.5	16.5	20
Report + List		8	5		
Learning the tool		0	13.5		
Total		**48**	**59**		

[4] http://www.nesma.nl/section/home
[5] http://en.wikipedia.org/wiki/Function_point

4 Results

Although the application was not actually deployed in the organization, its possibilities for the organization can be valued; the current implementation with Excel sheets is definitely less scalable than the created application. Also, the Excel sheets are difficult to maintain and to trace back to the owner. Next, while the Excel sheets only supported the gathering of data, the realized application also provided support for the processes.

The flexibility of the realized application has proven to be good, both for functional changes, such as last-minute changes in the data model, and technical changes, such as a stack change from Cocoon/EJB2 to Struts/EJB3; the impact of changes is small and predictable, implying that maintainability of the application is also ensured. Greater flexibility and maintainability has a positive effect on the time-to-market. Moreover, the quality of the software satisfies the characteristics of evolvable software as described by the Normalized Systems theory.

The productivity gain is hard to measure since the approach — RUP, SCRUM, etc. — highly influences the time needed for an activity. However, there are activities that can be compared, as indicated by a grey shade in Table 2: As the Normalized Systems has its architecture defined in the tool set, only for custom components, not provided by the NS tool set, time has to be spent on Architecture and Design. And since the NS tool set provides standard functionality for, i.e., persistency, remote access, input validation, logging, and user management, only for custom components time has to be spent on Construction. Also, since the provided architecture and standard functionality is already tested thoroughly, only custom components need to be unit tested. On the other hand, less time was spent on Requirements Specification — instead, every requirement was almost instantly implemented. Also, because the project was a PoC, no real functional tests were performed — only during the demonstration sessions. In total, the productivity gain for an entire project, from business modeling until deployment including project management, is estimated at a factor of 2 to 2.5. With these results, Normalized Systems is considered a key enabler in *Rapid Enterprise Transformation*; an interesting next step after Mulder's DEMO-based *Rapid Enterprise Design* [5])!

Short feedback loops are a main feature of iterative (software development) approaches such as SCRUM. However, feedback loops are only useful when feedback can be processed within a reasonable amount of time. Because the Normalized Systems approach uses expanders for code generation, changes in requirements originating from feedback can be processed almost instantly. The short iterations between business modeling and implementation, proved to provide better insight in the organization (see Fig. 5 for the final process model), and enable almost simultaneously (re)engineering the organization and its supporting IT. Some changes cannot always be processed instantly but require more work. However, as with maintainability and flexibility, with the Normalized Systems approach it is very clear what the impact of a change is. Normalized Systems therefore is considered a key enabler in agile system development.

5 Reflection

At the start of the project, it was thought the provided DEMO models represented the organization's needs and the requirements of the information system to be built. However, as soon as implementation started, it was clear the models lacked information. Samples of the current implementation of the enterprise (Excel sheets) were helpful in clarifying the DEMO models and identifying additional needs. However, because the Normalized Systems approach enables quick implementation and demonstration, enabling the organization to provide feedback, it was found the DEMO models were not valid and needed to be adapted. Does this mean DEMO models do not suffice in creating information systems? Or does it mean DEMO models can only be validated by means of implementation, where with Normalized Systems these models can be implemented and thus validated within a significantly shorter period?

A significant part of the budget (13.5 man days out of 59) was spent on getting to know the toolset or solving issues with the toolset, and also the reason for the project to run over budget (11 man days over budget). A major issue in development was the lack of testing possibilities, of both the expander and generated code. A list of recommended improvements on the tool set is composed and is currently being processed by UA and Capgemini in productizing the NS tool set. To be fit for large-scale use in MoD, the NS expanders should also support geographical positioning and editing of data, tabular data entry, and possibilities for generating mobile platform specific code. It is estimated these features can be implemented within six months.

Also, the sprint length of one week was found to be rather short, making it difficult to reach target when activities take some longer than planned and the overhead of collective project sessions and project management activities too large. It is expected that with a team with more experience with the NS tool set, more mature expanders, and longer sprints — of say two weeks — the productivity improvement is even higher. However, the willingness of all people involved to communicate at high frequency and with short response time, was also rated as one of the success factors of this project and also had a major positive influence on the productivity gain. Without commitment of all parties involved, the productivity gain will be much lower.

Based on this evaluation of Normalized Systems, we think in future NS has to be incorporated in existing enterprise transformation methods. We will now illustrate the potential impact of NS in four areas of enterprise transformation.

First, using Normalized Systems itself requires an enterprise transformation as it impacts the skills required from the people in the organizations: As the tool set already provides standard functionality, less programmers will be required for realizing a single application. Instead, business modelers will need analysis skills as well be able to use the NS tool set in creating a basic application. Some of the programmers can then focus on the application-specific interfaces and connections that are not provided by the NS tool set (custom components) while other programmers focus on maintaining or extending the NS tool set. Since the customizations in this PoC appeared to be small and can be

well-defined, this perfectly fits in the trend of right sourcing. Also, less application testers will be needed; instead of focusing on 'did we build the things right', the focus can be more on 'did we build the right things', as can be determined by modelers and subject matter experts.

Second, Normalized Systems can ease the splitting and allying of organizations. Since it is easy to identify the impact of business changes on IT that is built according to the NS theory, the impact of outsourcing parts of the organization on the IT becomes well-predictable. This may change insight in which parts are or are not eligible for outsourcing. Moreover, as the Normalized Systems theory ensures easy change of the IT system, including IT splits, the process of outsourcing will be sped up, significantly improving the agility of organizations in splitting, but also in allying with new partners [4, p.102]. Also, the Normalized Systems theory might provide insight in how to engineer the organization in such a way that organization splitting can be done much faster.

Third, the typical cycle of enterprise governance can be shortened by using Normalized Systems. Although the typical cycles of enterprise governance are already shortening, it is often the IT that is believed to cause the most delays in these projects because of its complexity. Normalized Systems is considered to enable shortening these projects, thus shortening the typical cycles of enterprise governance.

Finally, Normalized Systems highlights the importance of data foundations and the need for structural renewal of these data foundations of many organizations. As organizations change, their information needs in terms of information products change as well. However, if the underlying data services are well-designed, the data services do not have to change. Let's illustrate this with an example of Rijkswaterstaat (RWS), the Dutch Agency which is a/o accountable for Road and Traffic Management. At RWS, until 2005 two roles a/o existed, viz. 'road quality observer' and 'dead animal remover', each implemented by different people. The road quality observer was supported by an asset observance list and geographical data, and the dead animal remover was supported by a dead animal list and geographical data. The three information products were each supported by its own data service. From 2006, the two roles were combined into one functionary type 'road inspector', fulfilling both roles, now supported by three data services. Because the data services were well separated, instead of combining the information products, the three data services could be combined into one information product, offering live data about road inspections and dead animals as well as a map. This shows that well-designed data services support evolvable information needs and enable evolvable information products. In this PoC, NS has shown its strength in stable data-driven applications. We think organizations should consider rebuilding their data foundations with NS, as it offers interfaces through which the data services can be combined in a maintainable way, being able to follow the changing information needs of the organization.

Acknowledgments. We want to acknowledge Colonel Jean-Paul L. Duckers, Commander C2SC, for his sponsorship for this Proof of Concept Normalized

Systems, and for generously making available the materials for this project. A thanks for cosponsorship goes to Professor Herwig Mannaert, Vice-Dean Faculty of Applied Economics at University of Antwerp, and to Nico Brandt, Director Custom Software Development at Capgemini. Finally we want to thank all Ministry of Defense and Capgemini experts and team members involved, for their wonderful cooperation and valuable input.

References

1. Lehman, M.: On understanding laws, evolution, and conservation in the large-program life cycle. Journal of Systems and Software 1, 213–221 (1980)
2. Mannaert, H., Verelst, J.: Normalized systems: re-creating information technology based on laws for software evolvability. Koppa, Kermt, Belgium (2009)
3. Krouwel, M., Op't Land, M.: Combining DEMO and Normalized Systems for developing agile enterprise information systems. In: Albani, A., Dietz, J.L.G., Verelst, J. (eds.) EEWC 2011. Lecture Notes in Business Information Processing, vol. 79, pp. 31–45. Springer, Heidelberg (2011)
4. Op't Land, M.: Applying architecture and ontology to the splitting and allying of enterprises. PhD thesis, Delft University of Technology (2008)
5. Mulder, J.B.F.: Rapid Enterprise Design. PhD thesis, Delft University of Technology (2006)
6. Dietz, J.L.G.: Enterprise Ontology – Theory and methodology. Springer, Heidelberg (2006)

Defining Requirements for an Entrepreneurship Marketplace: Business and IT Alignment in Practice

Lucien Etzlinger[1], Pierre Castori[1], Gil Regev[2], and Alain Wegmann[2]

[1] e-novate, Geneva, Switzerland
{lucien.etzlinger,pierre.castori}@e-novate.ch
[2] Ecole Polytechnique Fédérale de Lausanne (EPFL),
School of Computer and Communication Sciences
CH-1015 Lausanne, Switzerland
{gil.regev,alain.wegmann}@epfl.ch

Abstract. e-novate is an IT consulting company based in Geneva, Switzerland. They recently made a radical change to an IT product they were developing. The change compelled e-novate to define a new business model for the product and to align it with the existing IT architecture. Through e-novate's episodic relationship with a research team, they learned about the SEAM Enterprise Architecture method. Based on a set of research papers downloaded from the web, e-novate's models were created as prescribed by SEAM for defining early requirements, they were validated by their stakeholders and the required changes were implemented. In this paper we present the project, the reasons for selecting SEAM, the models that were created, the difficulties in creating and applying them, and the lessons learned for both practice and research.

Keywords: Requirements practice, business models, enterprise architecture, business and IT alignment, entrepreneurship platforms.

1 Introduction

e-novate IT Consulting is a small local IT consulting company located in Geneva, Switzerland. e-novate is specialized in Customer Relationship Management (CRM) system implementation and sales process enhancements. e-novate's team consists of experienced consultants who have demonstrated a strong ability for leading major projects for a multitude of businesses in different countries, e.g. Finance, trading banking, food and beverages.

In February 2010 e-novate undertook the development of a completely new service, called TradeYourMind (TYM), a platform for supporting entrepreneurship. Defining the requirements for the platform proved problematic because e-novate first needed to define the business model to implement. The business model includes aspects such as, what should the service provide, with which partners, to which clients, and how the service would become profitable. These aspects must be clarified in order to define meaningful requirements and IT architecture. The traditional requirements methods e-novate was accustomed to using (e.g. Use Cases, Business Process

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 122–133, 2011.

Modeling) do not have the concepts necessary to define such a business model and to align the requirements with it. e-novate was therefore looking for a business and IT alignment method that would enable them to systematically move from the business model to the IT.

e-novate had an episodic relationship with the research team that created the SEAM Enterprise Architecture method [1]. SEAM contained the modeling concepts that e-novate were looking for, e.g. market segment analysis, value analysis, alignment with operations and IT. e-novate was able to successfully use SEAM for defining the business model and requirements for TYM by creating the models described in the research papers downloaded from the research team's website. In this paper we show why and how e-novate used SEAM. We describe the advantages as well as the problems that e-novate faced.

In Sections 2 we describe the business situation surrounding the TYM project. In Section 3 we explain the selection process of the modeling method. In Section 4 we explain the SEAM models that were used for specifying TYM. In Section 5 we reflect on the advantages and problems e-novate faced using SEAM.

2 The TradeYourMind Project

At the beginning of 2009, e-novate came to the conclusion that it spent too much time and money in pre-sales activities, such as the Request For Proposal (RFP), often imposed by customers. The way customers select suppliers for a project through an RFP is opaque to suppliers such as e-novate. The process does not allow a supplier to exchange key information in a proactive and fair way (e.g. How many companies are bidding on the project? What is the real budget and timeframe? What are the hidden considerations of decision-makers to launch the project? What is the strategy behind the project?). Discovering this information is expensive and time consuming for suppliers but is essential for providing value proposals to customers. As a way to address these problems, the idea emerged to develop a web application that will help collect key sales information early in the process.

e-novate's initial goal was to create a web platform that enables customer companies to post new ideas of projects, help them to transform their ideas into concrete projects and finally link them with the best possible suppliers, thus reducing the pre-sales efforts for all parties. The underlying model was the trading of commodities (e.g. buy raw material, transform, deliver and sell it as a finished product). e-novate called this web platform TradeYourMind.com with the slogan, "When ideas become projects". The underlying idea behind this platform was that the earlier suppliers are involved in the tender process, the better their proposal will be.

In October 2009, e-novate demonstrated a first prototype to several customers. The customers did not appreciate the idea of offering more transparency in their purchasing process. They were either afraid of reducing competition between suppliers or they already had tools to manage their RFP process.

During a special event held in Geneva in February 2010, e-novate met a representative of a non-profit organization that provides financing facilities to startups in the Geneva region. The representative asked e-novate whether TradeYourMind.com could be used for making entrepreneurship in Geneva more open and connected.

Breaking away from the previous business idea (i.e. a platform to support the RFP process), e-novate embarked on the development of a platform for helping entrepreneurs to connect and collaborate with incubators and venture capitalists. This new business idea forced e-novate to define requirements for a new platform for supporting entrepreneurship. e-novate then faced the problem of building an application for a brand-new business. A new business model had to be specified completely because it did not exist yet. The requirements for the platform naturally depended on this business model.

The use of SEAM enabled e-novate to define a business model for a marketplace of ideas that brings together entrepreneurs, business partners (e.g. investors, service providers such as insurances, banking services, legal advisory, telco, office equipment, hardware) and incubators (organizations supporting entrepreneurship). In this marketplace, entrepreneurs offer business ideas for sale and receive help in sharpening their ideas. Business partners can "buy" projects that they want to contribute to. Incubators can virtually host entrepreneurs and link them with business partners.

The development of TradeYourMind.com was insured by e-novate's own funds. The project lasted for 11 months. In March 2010, they began modeling the initial business ideas with the Business Model Canvas [2]. e-novate then switched to SEAM for defining the business models. They developed the service and behavior models in April and May 2010. The data model was defined in June 2010. The application was developed from September 2010 until January 2011. The first customer signed up at the end of 2010.

The overall time spent on the business model and the requirements was 190 days. Within e-novate the effort involved two people with the roles of business analyst and developer. It also involved potential customers, an incubator and two entrepreneurs.

3 Requirements Methodology Selection

e-novate initially used a waterfall process based on PRINCE2 to manage the project. In the feasibility study phase, they evaluated three technologies for modeling and implementing the system:

1. Aqua Logic Business Process Management from BEA System
2. CRM Salesforce from Salesforce.com
3. The Intalio Open Source platform

These are platforms that include a modeling front-end, as well as a programming environment. After testing these platforms and making additional surveys, e-novate opted for Salesforce.com for its flexibility, extremely fast development process, and easy integration of the TYM kind of applications with Salesforce's CRM functions.

To complete the Initial Phase, e-novate began to specify the requirements for their future web portal. After a few weeks, they realized that they first needed a clear view of the business model they wanted to implement. Something was missing in the approach. How could they know who are the real users? Who will pay for the services? What part of the services will be free of charges? TradeYourMind.com was an entirely new business with no existing references and everything had to be built from scratch. e-novate had to imagine directly the «TO BE» model without going through

the «AS IS» model that usually enables a better understanding of the business environment in which the company operates.

To overcome the lack of a business model, e-novate first used the Business Model Canvas [2]. The use of the Business Model Canvas clarified some of the business issues. However, the Business Model Canvas does not have tools for aligning the business model with the underlying IT platform. e-novate needed models for defining the requirements based on the business model and deriving the IT architecture.

Partly through chance and partly through its business network e-novate was exposed to SEAM [1]. A chance encounter with one of the SEAM authors in a business conference provided the spark and it so happened that e-novate's key developer personally knows the SEAM authors. Despite this relationship e-novate had only two short informal introductions to SEAM. Through these introductions e-novate realized that they could use SEAM for the alignment of their system with their business. e-novate downloaded four research papers describing SEAM [3, 4, 5, 6]. The papers described SEAM for Business [3], SEAM goal-belief models [6], SEAM for Enterprise Architecture [4] and SEAM for Software Architecture [5].

Based on these four papers, e-novate began to build its own SEAM models. They faced difficulties apprehending the global concept behind SEAM and understanding the links between models. All through this project they made no attempt to contact the SEAM authors because they did not think that their project presented challenges that were interesting enough for the researchers. They built all the models in complete independence. Only after TradeYourMind.com was operational did they contact the SEAM authors to show them the product and the SEAM models. e-novate and the SEAM authors are now working together to improve the TYM models and therefore TYM itself. However, the models in this paper are the originals that were created independently of the SEAM authors.

SEAM for Business provided e-novate with the modeling tools to generate several business models. With these models e-novate was able to see the possible scenarios and thus choose the most promising one. The business model selected by e-novate was a multi-sided platform pattern [2]. A multi-sided platform is characterized by several interdependent customer groups who are all needed for delivering value to each one of them. In the TYM case, these customers are the entrepreneurs, the partners and the incubators. Their existence and relationships are necessary for TYM to succeed.

SEAM for Enterprise Architecture was used to fill the necessary information to set up the design and development stages with a clear understanding, for all contributors, of the business configuration and the expectations of the adopters' value networks in terms of service value. It was key for e-novate that people in charge of the application development really understand to whom and how the service is to be delivered, thus ensuring that the envisioned web platform will be aligned with the defined business strategy.

4 Results of SEAM Applied to TYM

The SEAM research papers describe the succession of models to be created but do not explicitly define a SEAM modeling process. Figure 1 shows e-novate's understanding

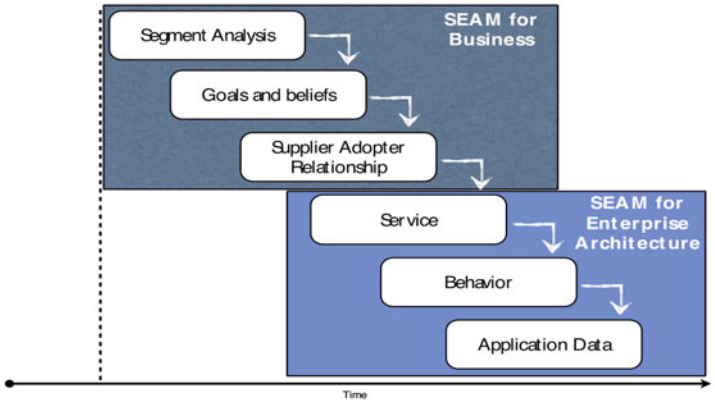


Fig. 1. SEAM Development Process as inferred by e-novate

of this process. The steps in Figure 1 give the succession of models that e-novate built during the project. These models are shown in Figures 2 to 7. Note that these are the original models created by e-novate before the SEAM authors reviewed them. They are not as rigorously aligned as they would have been in a research paper.

The segment analysis model (Figure 2) was used to specify the different components of the market. It confirmed that the entrepreneurship market is a multi-sided market involving 3 main actors who need to be addressed in parallel.

Figure 2 shows a SEAM segment analysis model for the service provided by TYM to its main adopters, i.e., incubators. It names the segment as IT for entrepreneurship.

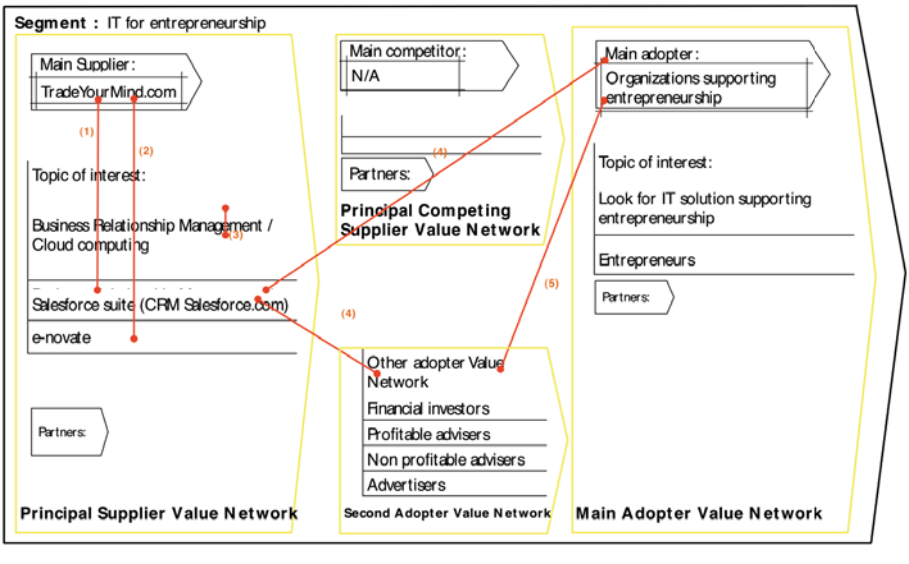


Fig. 2. TYM Segment Analysis Model

It shows that in e-novate's view there are no competitors for this service. It identifies Salesforce as the enabling technology for this service. It shows a number of secondary adopters (e.g. investors, advisors and advertisers).

The SEAM Goal and Belief model (Figure 3) was used to better understand the stakeholders identified in the segment analysis model. The Goal and Belief model shows who these stakeholders are. The model shows the problems the TYM stakeholders (incubators, entrepreneurs, investors) face without the TYM service, and their expected gains from the TYM service. For example, the model shows that e-novate think that incubators need a tool to manage an increasing volume of entrepreneurs. This participates in the justification for the development of TYM. Note that the model, as originally done by e-novate is not really aligned with the model in Figure 2, For example, the entrepreneur in Figure 3 is named applicant in Figure 3.

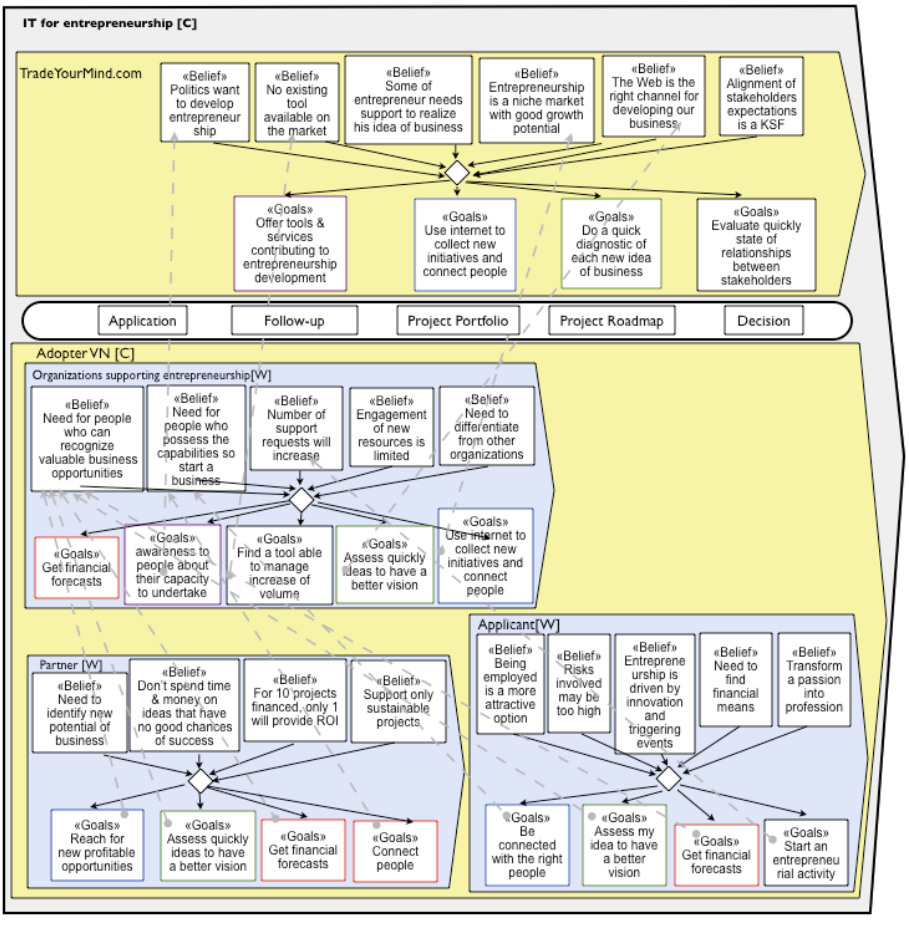


Fig. 3. TYM Goal-Belief Model

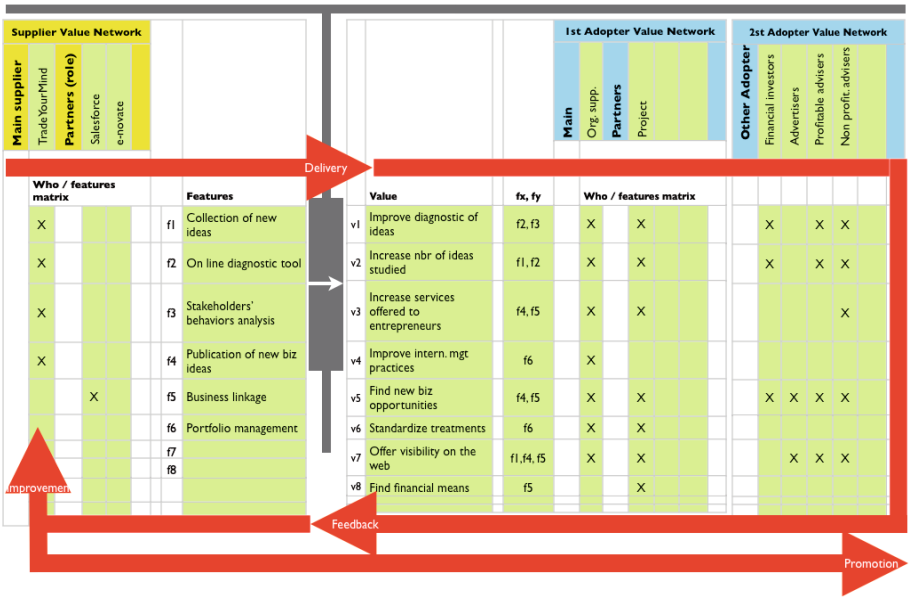


Fig. 4. TYM Supplier-Adopter Relationship Model

e-novate reviewed this model with customer segment representatives in order to validate e-novate's assumptions about stakeholders' expectations. This is also a very important step in defining the services provided to customers.

The goals defined in the Goal and Belief model were used as input for the Supplier-Adopter Relationship (SAR) model, (Figure 4). The SAR maps customer value to service features and the features to service components. It enabled e-novate to identify macro functionalities to be delivered through the customer portal, in order to provide added value to its customer segments. For example, incubators will see value in a tool for improving their evaluation of ideas (called diagnosis in the model) and the number of ideas they can evaluate.

The SEAM behavior models (shown in Figures 5 and 6) describe how TYM would deliver services to customers and what kind of information the stakeholders would exchange. e-novate produced a macro representation of the two essential components of the TYM customer portal, «Sell Ideas» (Figure 5) and «Buy Projects» (Figure 6). These models show how the business process as viewed by TYM and its stakeholders (e.g. entrepreneur, incubator, partner).

During the design of the sell and buy processes, e-novate collected bulk information (e.g. adhesion questionnaire, reports, sample of projects) from potential adopters to enrich their understanding of the information that needs to be managed by the portal.

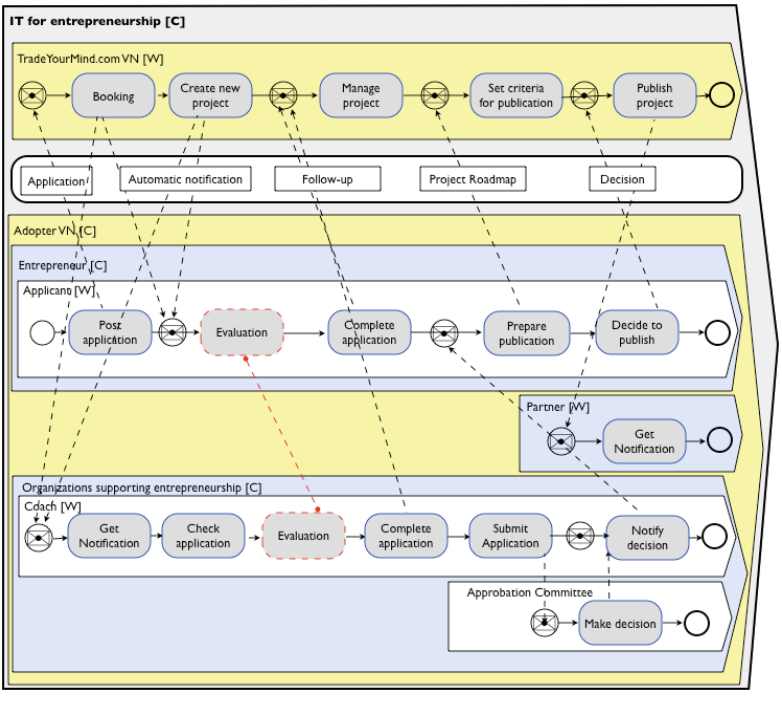


Fig. 5. TYM Sell Ideas Model

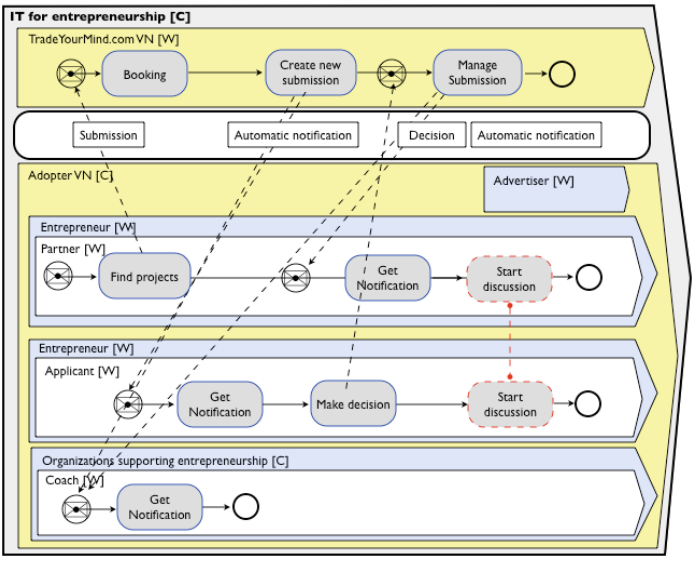


Fig. 6. TYM Buy Projects Model

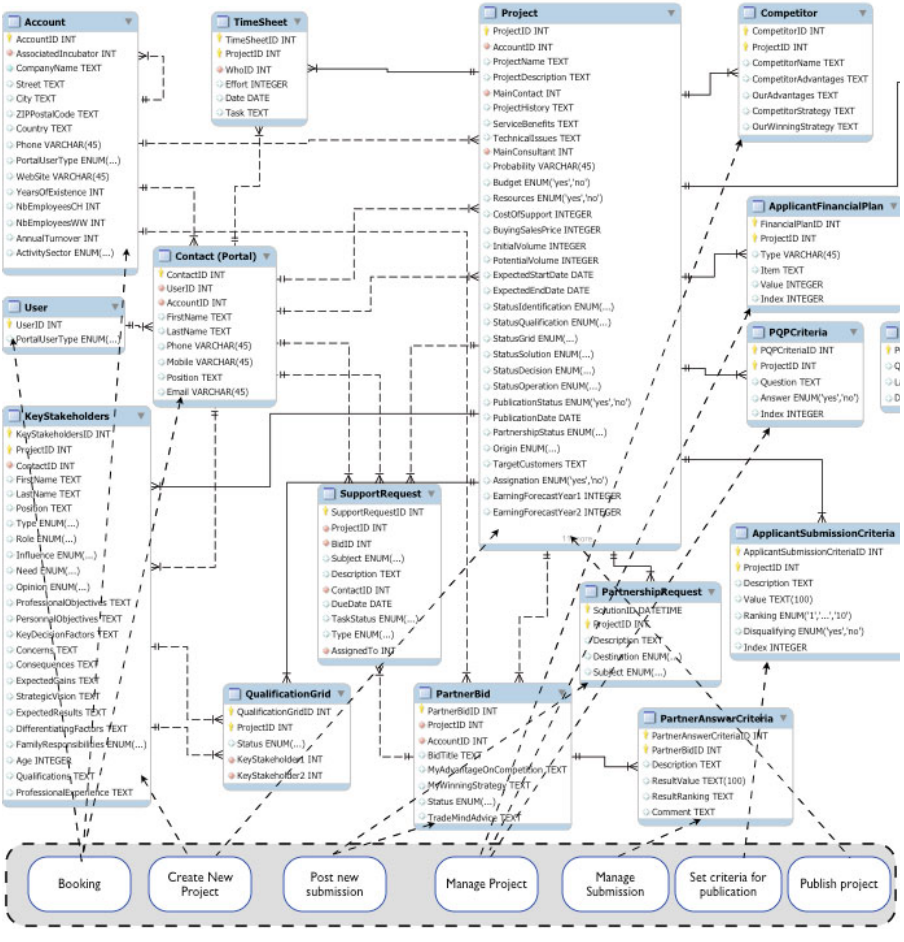


Fig. 7. TYM Partial Data Model

With the macro processes identified in the behavior models, e-novate created the application data model (a partial view is presented in Figure 7). This establishes the link between the macro business process and data that need to be managed by the IT systems. For example, we see the relationship between the Project table and the Create New Project activity. The data model is a key element that supports the web based application development. e-novate did multiple iterations until they finalized it. e-novate were able to specify, at the same time, the data security model and access rights because they now had a clear understanding of the portal roles and could rely on Salesforce's highly integrated security functions.

5 Discussion

e-novate were looking for a modeling method that would enable them to define the TYM business model and then systematically move toward its implemented. e-novate

chose SEAM because it offered these features but also because they knew its authors and could obtain help from them if they needed it. As is customary in business settings, they did not do a comprehensive search for the state of the art in business and IT alignment methods. The relationship with the authors of a method or with other experts who can help in its use should not be under-estimated when analyzing the choice of a method.

Without the support of the SEAM authors, it took e-novate much time to understand and interpret the research papers. The lack of correlation between the papers illustrating the different parts of SEAM did not offer the necessary global view of its integration. e-novate were able to create and maintain the SEAM models despite the lack of a specialized modeling tool. However, The lack a tool was penalizing for the maintenance of the models. e-novate had to build the models using Keynote or PowerPoint, which are not efficient tools for this purpose. To keep the models aligned, e-novate had to manually manage the relationships between the models. When they changed a model, they had to review all the others for possible changes. They would really appreciate having an application allowing them to quickly and dynamically manage SEAM models.

The models shown in this paper are the original models, before the authors of SEAM reviewed them. These models are not totally aligned from an academic point of view. For example, they do not use the same names for the same concepts across all models. However, the models were sufficiently aligned for e-novates purposes. We believe that the models were valuable for e-novate despite them being non completely aligned because of the co-construction of the models by complete development team. e-novate involved the development team from the strategic goals definition to the business modeling process. As a result, they all had a clear vision of the business environment, which ensured a good alignment of the business strategy with the envisioned IT system. More precisely, the SEAM models enabled e-novate to have, before they began the development phase, a clear idea of the following aspects of the system:

- Portal roles and the process specific to each role (including triggering events and notifications)
- Data elements accessed by each role.
- Security constraints for each role and data.
- Interfaces between the IT system (platform) and its environment.

The switch from design to development therefore resulted naturally. A key added value of SEAM is that developers can easily trace the requirements and know why a development step has to be taken. The ‘why’ is important because development quality is improved when developers are able to justify their decisions.

Some aspects of SEAM (e.g. Goal and Belief models) were difficult to understand and apply. However, they proved to be of much value for understanding customer concerns. e-novate invested a substantial amount of time and energy to build all the models proposed in the research papers even though some of the models were not well understood.

e-novate had the opportunity to review and validate some of the SEAM models (Segment Analysis, Goal and Belief and Behavior models describing Sell Ideas and Buy Projects processes) with their customer representative, who is an experienced

coach for entrepreneurs. e-novate spent two sessions of 3 hours reviewing and improving the models. Reading and interpreting SEAM models were not difficult for the customer. e-novate also discovered that SEAM helped them to not dive directly into the process details; the high-level models proposed by SEAM are more focused on strategic issues than on operational details. The Segment Analysis model enabled the customer to quickly understand what kind of business model is behind the web platform and opened the discussion about it. e-novate collected some key information about business issues.

The Goal and Belief model was the most interesting model to review with the customer representative. It helped e-novate to identify their customer's main business concerns and thereby to improve their value proposition. It was also the most difficult to understand and create because it was not clearly documented by the SEAM authors. To better understand these models, it was necessary to meet with the authors.

e-novate's goal using the behavior models was not to describe in detail the processes rather to identify the basic processes that needed to be implemented. To go deeper in the process analysis (e.g. process enhancement, re-engineering), it is possible to use a common Business Process Management approach with a Process Modeler tool such ALBPM, or Intalio, to complement SEAM. These products provide the necessary information that can be useful later to the detailed work of the development team.

6 Conclusions

In this paper we presented business and IT alignment of an on-going project in a consulting company in Geneva called e-novate. We explained how and why e-novate chose the SEAM method, its advantages and disadvantages.

Using SEAM for TradeYourMind.com was considered by e-novate as a success because they clearly demonstrated that they were able to design and build TradeYourMind.com from the stage of a single idea into a concrete customer portal where people make e-business transactions and exchanging information on the web.

e-novate used SEAM over the whole hierarchy from the business model down to the data model. This enabled them to move smoothly from the business modeling phase to the development phase.

e-novate first developed all the SEAM models and built the platform with no help from the authors of SEAM. e-novate presented the models and the platform to the authors of SEAM only after the first version of TYM was deployed. This prompted the refactoring of the SEAM models, which will result in changes to the platform itself. The models could have been better aligned and TYM would have been better suited for its mission had e-novate contacted the SEAM authors during the project. E-novate and the SEAM authors are now working together to improve the models and TYM itself.

References

1. Wegmann, A.: On the Systemic Enterprise Architecture Methodology (SEAM). In: International Conference on Enterprise Information Systems (ICEIS), Angers, France (2003)
2. Osterwalder, A., Pigneur, Y.: Business Model Generation. Self published (2009)

3. Wegmann, A., Regev, G., Rychkova, I., Julia, P., Perroud, O.: Early Requirements and Business-IT Alignment with SEAM for Business. In: 15th IEEE International Requirements Engineering Conference (RE 2007). IEEE Press, Los Alamitos (2007)
4. Wegmann, A., Regev, G., Rychkova, I., Lê, L.-S., De La Cruz, J.D., Julia, P.: Business-IT Alignment with SEAM for Enterprise Architecture. In: 11th IEEE International EDOC Conference. IEEE Press, Los Alamitos (2007)
5. Rychkova, I., Regev, G., Lê, L.-S., Wegmann, A.: From Business to IT with SEAM: J2EE Pet Store Example. In: 11th IEEE International EDOC Conference. IEEE Press, Los Alamitos (2007)
6. Regev, G., Wegmann, A.: Defining Early IT System Requirements with Regulation Principles: the Lightswitch Approach. In: 12th IEEE International Requirements Engineering Conference (RE 2007). IEEE Press, Los Alamitos (2004)

Enterprise Transformation: The Serasa Experian Case*

Pedro Sousa[1], Ricardo Gabriel[2], Gustavo Tadao[2], Rosana Carvalho[3], Pedro Miguel Sousa[3], and André Sampaio[1]

[1] Link Consulting, Avenida Duque de Ávila, nº23 1000-138 Lisboa, Portugal
[2] Serasa Experian, Alameda dos Quinimuras, 187, CEP 04068-000 São Paulo, SP, Brasil
[3] Aitec Brasil, Rua Luigi Galvani, 200, CEP 112 04575-020 São Paulo, SP, Brasil
pedro.sousa@link.pt, ricardo.gabriel@br.experian.com, gustavo.tadao@br.experian.com, rosana.carvalho@aitecbrasil.com.br, pedro.r.sousa@aitecbrasil.com.br, andre.sampaio@link.pt

Abstract. This paper presents the development of an Enterprise Architecture project at Serasa Experian. With a strong transformation plan Serasa needs tools to monitor the progress of the strategic roadmap. With this project Serasa achieved an integrated management of architectural views, projects results, and the roadmap progress. It is also brings IT Architects, Project Managers and IT Governance together, identifying concepts and a common language between them, thus facilitating the aggregation of information from these groups into a consolidated vision. Project completions are the drivers for updating the architectural views and progress of the roadmap. A knowledge base keeps the information gathered from external information sources and supports the automatic generation of architectural views, which can be browsed back and forward in time.

Keywords: Enterprise Transformation, Enterprise Architecture Case Study, Temporal Architectural Views.

1 The Situation

The Serasa Experian[1] (hereafter designated by Serasa) is the biggest credit bureau in the world outside the USA, holding the largest data bases in America Latina about consumers, companies and economics groups. With over 40 years of experience in the market, the company is involved in most credit decisions taken in the country, corresponding to 4 million queries each day, done by over 400 thousand clients.

Serasa makes available integrated solutions that cover client needs regarding credit data: Market Prospecting, Customer Management, Retention and Profitability, Procurement and Concession of Credit Portfolio Management, Credit Management and Billing Fraud and Validation.

* The statements and opinions expressed in this paper are those of the authors and not an official position of Serasa Experian.

[1] www.serasaexperian.com.br

F. Harmsen, K. Grahlmann, and E. Proper (Eds.): PRET 2011, LNBIP 89, pp. 134–145, 2011.

Serasa has a strategic plan for the entire IT that is embodied in roadmaps for the key business and technologic areas of the company. These roadmaps set the targets to be achieved for each area over the next five years. These targets are defined in terms of a maturity level measured on a 5 level scale (non-existent, initial, defined, managed, and optimized).

At the end of each year, each area submits a plan with the initiatives proposed for the coming year. If approved, these initiatives become actual projects. Then, one year after, each area evaluates the progress achieved and updates the roadmap accordingly. Projects are transformation elements of the organization, and so project achievements drive the progress of roadmaps. Roadmaps are manually maintained in numerous Excel Sheets; therefore updating the roadmaps according to projects' results is a complex and time consuming process.

Project planning requires knowledge of the actual state of the organization, nonetheless, such knowledge is scattered across numerous fonts, and there is no simple mechanism to consolidate disparate information into an integrated view of the actual state of the organization. Consequently, the project planning should include a survey of the As-Is of the project related areas. However, this is insufficient. Since there are multiple parallel projects the identification of the project impact cannot be established taking only into account the project related As-Is, it also ought to consider the foreseen future (To-Be) accomplished by parallel projects (on-going and planned) up to the day the project is set to production [6]. Therefore, the project planning should also include a survey of on-going and planned projects that may affect the project in question. Without taking the previous considerations into account the probability of finding dependencies between projects solely during the execution is indeed high, which can lead to an increase in projects costs and execution time, thus jeopardizing the roadmap's success.

In early 2011, Serasa has initiated an Enterprise Architecture project, perceived as an instrument to control and monitor the transformation of the organization achieved in each project. This project was led by the Corporate Architecture team[2], which is comprised in the Department of Information Technology. The IT areas involved in this project were: IT Governance and Systems Development, Security and Infrastructure, and Corporate Architecture.

2 The Project

Our task was to design, plan and implement an instrument that enables the integrated management of architectural views, projects results, and the roadmap progress. More specifically, this instrument should allow Serasa to:

- Have an integrated and up-to-date view of the As-Is of business and IT architecture, by consolidation of information from different sources.
- Have a view of the To-Be state of the business and IT architecture based on the foreseen results of on-going and planned projects.

[2] See Serasa "*Case empresa: Arquitetura Corporativa, uma gestão integrada do Roadmap de TI*" presented in the conference: www.congresso-ae.com.br/programacao.php.

- Have the ability to identify the progress of each roadmap updated according to the projects' progression.

The following figure presents a generic view of the instrument. Basically, it imports textual information from various information sources, generates architectural views and keeps roadmap progress up-to-date. The instrument is built upon two tools: IBM's Rational System Architect and Link Consulting's Enterprise Architecture Management System[3] (EAMS). System Architect serves primarily as the knowledge-base. Information integration and architectural views generation is achieved via EAMS.

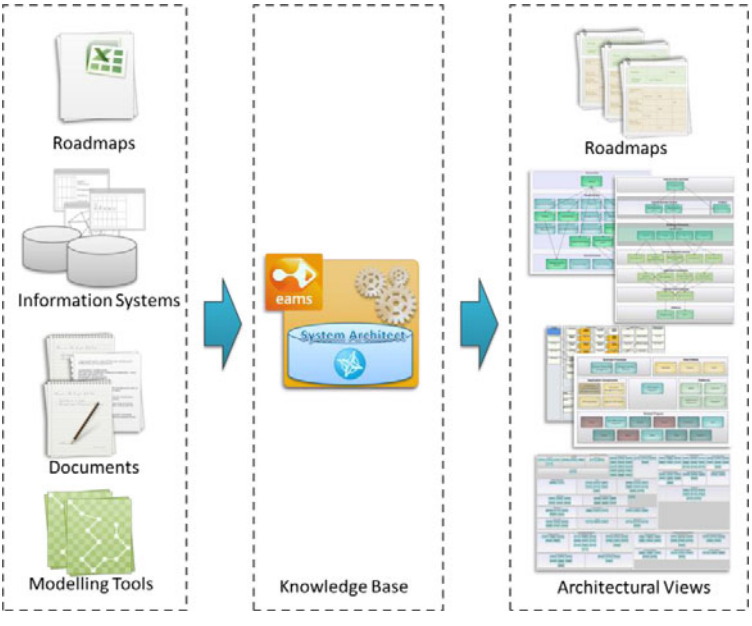


Fig. 1. Overview of the instrument to implement

2.1 Information Integration

Regarding the subject of Information Integration, our task was to import the information into the knowledge base, via integration with other information sources scattered throughout the organisation or by introducing the information directly into the knowledge base.

Whenever the organization already has the processes to maintain and update a particular source of information, one should provide the automatic import mechanism to integrate it with the knowledge base. Otherwise, one must decide between creating a new information source or entering the information directly into the knowledge base, and introducing its update process to the organization.

[3] www.link.pt/eams

Information sources can be highly structured such as information systems, less structured such as modelling tools, intranet pages, or even documents such as Microsoft Office (Word, Excel, PowerPoint and Visio), among others. Information integration must be made from structured sources. Therefore integrations with unstructured sources require prior structuring of the same. The roadmaps' Excel Sheets are an example where the structure is done by creating new Sheets filled with lookups and other Excel functions, creating tabular records ready to be imported.

2.2 The Knowledge Base

Regarding the subject of the knowledge base, our task was to prepare a repository that supports the evolutionary vision of architecture and its connection with projects and roadmaps. The knowledge base keeps the information that describes the organization as textual information and enables the automatic generation of architectural views [5].

A necessary characteristic of the knowledge base is that its entities have a life cycle that is associated with the project concept. We consider projects as the elements that induce the organization changes, thus being responsible for applying the changes to the organization's artefacts and relationships [6]. Such entails that every planned change results from a project. By planned change we state that the changes in the organization occur as result of activities with a planned start and end date. Unplanned changes are considered a priori in our approach, hence implying that a project has produced those changes in some past date.

Projects have a start date, an end date, and two lists with references to other artifacts. The alive list references the artifacts that will go into production (*alive*) with the project completion, and the dead list reference the artifacts that will be decommissioned (*dead*) after project completion. Using the projects start date and an end date, we tag each artifact within the knowledge base with tree time-stamps:

- *Gestation*, when it is being developed within a given project.
- *Alive*, when they are put into use within the organization as a result of a project.
- *Dead*, when they are no longer used, also as a result of some project.

These time-stamps are necessary to allow the generation of architecture views related to any time in the past, in the present or in the future.

2.3 Architectural Views and Roadmap

Here, our task is to define and implement the graphical outputs/models that represent the roadmap progress and the representation of Serasa's enterprise architecture. A key aspect is that these outputs must to be generated automatically from knowledge base contents.

As a general rule, we envisage the use of ArchiMate Viewpoints [1][3] for architectural views, but we also require a more generic approach to generate architectural views of any type of system, in the general sense.

Our approach, provided by default in EAMS, considers four types of viewpoints that can be used to analyse any set of artifacts in the knowledge base. We call them *Organic*, *Context*, *Structure* and *Integration* [6]. These views describe the elements

and dependencies of a system defined by a set of elements in a graph of a larger system [5]. For example in Fig. 2 we present the system *S01* as an ellipse and in figures 3-7 we present the matching generic views.

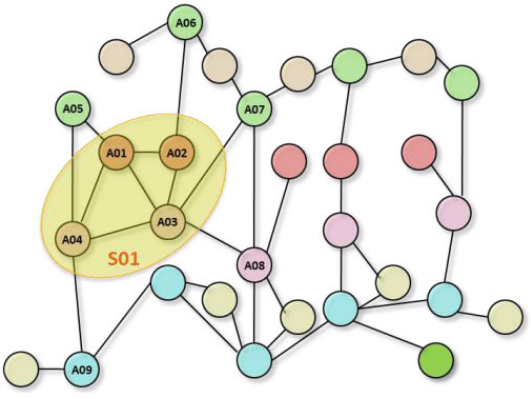


Fig. 2. System boundary of S01

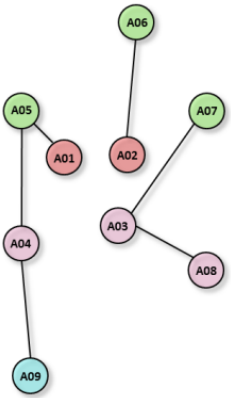


Fig. 3. Integration view of S01

The *Integration* system identifies the set of influence bonds among the members of the composition and the elements in the environment, and also the elements covered by these bounds. The *Organic* is focused in a hierarchical decomposition that identifies the artifacts in the composition of a system. The *Context* encompasses the environment of a system based on the relationship bounds established with its composition, and the *Structure* covers the artifacts in the composition and the set of structural bounds among the artifacts.

Fig. 4. Organic view of S01

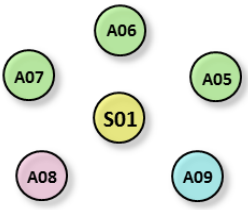


Fig. 5. Context view of S01

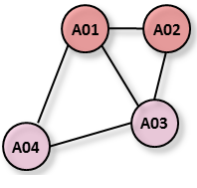


Fig. 6. Structure view of S01

In this general perspective, roadmaps correspond to a set of initiatives that relate to the artifacts to be created in each of them. Regarding the example in Fig. 2, the S01 roadmap would be complete when the initiatives A01-A04 created the artefacts A05-A09.

3 The Approach

The approach we used was designed to mitigate the main project risks, which were the following:

- Too much time spent on the definition of the knowledge base meta-model.
- Inadequacy of the meta-model's level of detail and the matching information sources. The complexity of the knowledge base meta-model should be suitable to the immediate needs and the existing information sources.
- Time spent adjusting unstructured information and manually guarantying the information quality.
- People can lose confidence in the project if they do not see short-term results.

Thus, we adopted an incremental approach by dividing the problem into several stages. The project began with an analysis and planning stage from which resulted in eight further stages.

3.1 Planning Stage

The analysis and planning stage includes the following steps:

1. Collect relevant issues from the different areas, and assess its priority.

   In this step we identified and prioritized 82 issues after workshops with each area. Some of these issues were high level statements, such as "*What are the costs of planned maintenance in the coming years?*" or detailed ones, such as "*What were the architectural artefacts produced in each project?"*.
2. Define knowledge base meta-model to hold information required to answer the identified issues. This is a quite straight forward exercise for the detailed issues. For high-level issues, one needs to unfold them into a set of detailed issues, and identify the information related to each one. For the example above, a set of more detailed questions could be: "*What is estimate regarding the resources required for the operation of each system, as disk?, mips?, bandwidth?,..?.* ".
3. Identify the best sources of information for each entity in the knowledge base meta-model.

   In a few workshops with each of the areas, 16 sources of information were selected. In parallel with this step, we did the installation and general configuration of the tools used, in particular the meta-model configuration.
4. Plan the whole project.

   The questions were clustered into eight groups, according to the related entities of the meta-model, thus resulting in eight further stages, presented next.

3.2 Further Stages

After the analysis and planning stage, one started the remaining stages, in sequence, each with having the following steps:

1. Analyse the corresponding information sources and find a way to extract the necessary information.

   The key factor is the degree of structure in each information source. For highly structured sources, the extraction is a quite straightforward task. For unstructured sources, one must first propose a structure and re-structure the data.

The structuring of information sources is based on eliminating the information in natural language and record all the information in specific tables whose cells only have predefined values. For example, the document describing the architecture of a *system A*, instead of having a natural language sentence such as "*The system A reads the information from the repository B through the middleware C*", could instead have a table named "*accessed repositories"* with the fields *Repository source* and *Middleware technology*, in which the values are filled based on pick lists with the allowed values for repositories and technologies, that in this example would be *B* and *C*.

But the structuring of an information source also involves the clarification of the semantics of each used artifact type. In the example above, one must define the semantics of the *Repository* and *Middleware* and ensure its existence as an entity in the meta-model. Thus it also becomes clear what dependencies may exist between artifacts. Continuing the previous example, it was necessary for the meta-model to hold a relationship between the entities *Repository* and *System*, and that this relationship may be qualified by the *Middleware technology*.

2. Implement a mechanism to feed the knowledge base with information residing in information sources.

   We prefer to import via editable files (like CSV or XML) because it allows a more controlled environment. One can even edit and correct wrong information before importation. The contents of each source of information were exported to conforming to a CSV format, so that responsible areas could validate the content using Microsoft Excel. The detected errors can be corrected later in origin.

   The importation mechanism must handle many situations that arise from imports, for example, repeated artifacts, artifacts with extra fields, default values, and so on. The importation mechanism must also address the rules to set the timestamps. For example, whenever a new artifact is imported the *alive* timestamp is set to the completion of associated project. If the project does not have a completion date, or if the artifact is not associated with any project, then the *alive* timestamp is set to the loading date.

   In addition, EAMS allows the definition and management of jobs and batches to handle multiple and sequential file importation thus enabling the load process to occur in well-defined moments.

3. Define and implement appropriate views to respond to relevant issues.

   As mentioned, we favour the use of well-established viewpoints, either the ones provided by ArchiMate, or the ones supported by default by EAMS. We found that in fact most answers can be found with these viewpoints.

   Another important aspect is the definition of the navigation path between representations according to the stakeholders' concerns [2], allowing different stakeholders to navigate to different graphical models after interacting with the same artifact in the same architectural view.

From the eight stages planned we are completing the first two. The time and effort depends mostly on the structuring of information sources. With clean and well-structured sources one can expect to take between 2 to 4 weeks for each stage.

4 The Results

The immediate result of this project is to provide to the various areas of IT a consolidated and updated view of the architecture of Serasa. Consolidated - because it results from the information provided from different areas. Updated - because it results from information sources maintained by the different areas with clear processes in an automated manner.

Since each group contributes with information to the other areas and receives information from others, the instrument entails an important transformation in the organization, as it homogenized languages and tools. In this regard, the knowledge base meta-model is an important asset because it identifies the concepts common to the various areas. The meta-model is illustrated in Fig. 7.

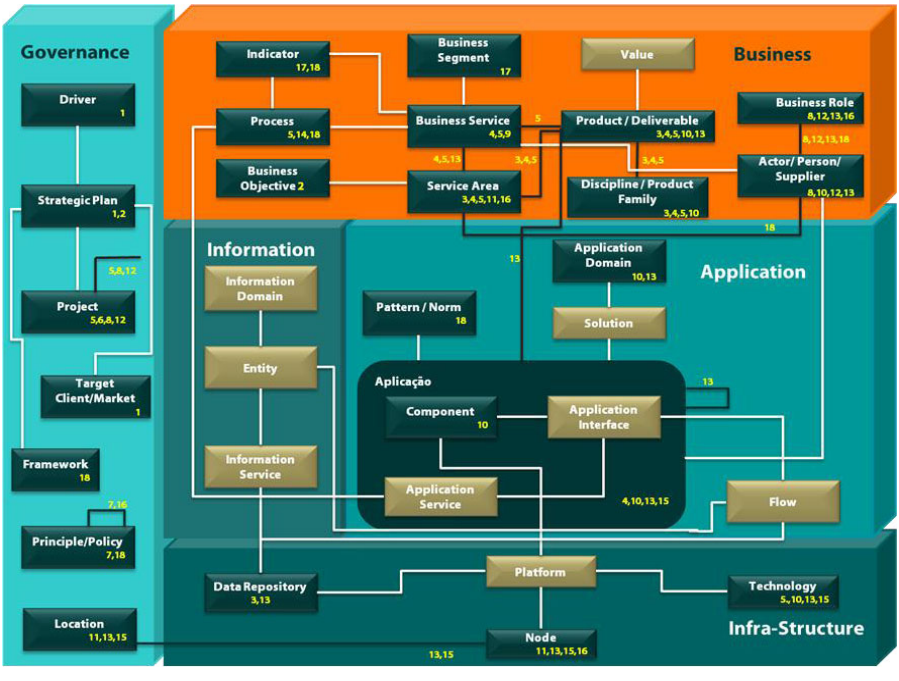


Fig. 7. Identified meta-model

The meta-model was derived from the 82 most relevant questions presented by the key areas. To answer these questions we have reached a model with 32 entities, allowing all but two of the issues to be answered. Both the entities and the relationships between them were somewhat influenced both by TOGAF [4] and the ArchiMate [3].

To populate these entities we have identified 16 sources of information, 9 of which are collections of Microsoft Office documents (PowerPoint, Excel and Word), and 7 are systems and tools (ChangePoint, SharePoint, BizAgi, Visio, and three other

specific systems). After thorough analysis of potential information sources, we found that most of the information needed to populate each entity in the meta-model is in more than one source, and in different formats. The numbers depicted in Fig. 7 represent the information sources associated with each entity.

An entity without a number means that we found no information source within the company holding the necessary information. We have 9 of such cases. In these cases it is necessary to identify the processes of Serasa where this information is produced and where it should be recorded. This allows the definition of the artifact's description that will be created in the future. The information survey regarding the artefacts that already exist implies a human endeavour that should be raised incrementally, as needed by on-going projects.

We now present examples of the architectural views generated, but with fictions values in order to hide Serasa actual IT. In addition to the more common approach of static visualization the solution allows interaction with the representations. Stakeholders may interact with produced blueprints by selecting and inquiring information about artifacts and navigating between blueprints.

The blueprint in Fig. 8 presents how an application "*Customer Order Management Application*" is organized regarding its *composition* (components distributed by application layer) and identifies both the structural bounds between the artifacts and the environment and the artifacts in the environment (platforms where the components are executed).

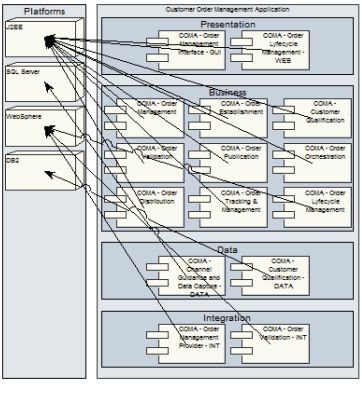


Fig. 8. Application Structure blueprint

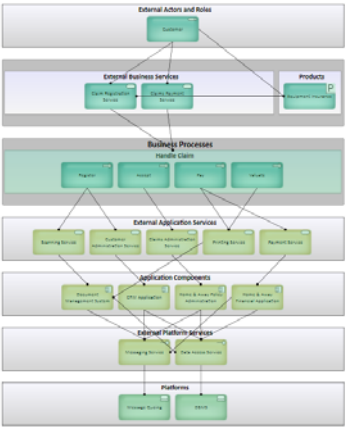


Fig. 9. Layered blueprint

Another example of an architectural view is represented in Fig. 9 this blueprint is an example of an instantiation of an ArchiMate viewpoint, the Layers Viewpoint [3].

Regarding the topic of information integration, Fig. 10 is an example of information automatically extracted from normalized documents that was provided to generate the blueprint presented in Fig. 8.

	A	B	C	D	E	F	G
1	Type	Name	Gestation Date	Start Date	End Date	Layer	Platform
2	Component	COMA - Channel Guidance and Data Capture - DATA	20-09-2010	15-12-2010		Data	SQL Server
3	Component	COMA - Customer Qualification	20-09-2010	15-12-2010		Business	J2EE
4	Component	COMA - Customer Qualification - DATA	20-09-2010	15-12-2010		Data	DB2
5	Component	COMA - Order Distribution	20-09-2010	15-12-2010		Business	J2EE
6	Component	COMA - Order Establishment	20-09-2010	29-11-2010		Business	J2EE
7	Component	COMA - Order Lyfecycle Management	20-09-2010	27-01-2011		Business	WebSphere
8	Component	COMA - Order Lyfecycle Management - WEB	20-09-2010	27-01-2011		Presentation	J2EE
9	Component	COMA - Order Management	20-09-2010	29-11-2010		Business	J2EE
10	Component	COMA - Order Management Interface - GUI	20-09-2010	29-11-2010		Presentation	J2EE
11	Component	COMA - Order Management Provider - INT	20-09-2010	29-11-2010		Integration	WebSphere
12	Component	COMA - Order Orchestration	20-09-2010	15-12-2010		Business	J2EE
13	Component	COMA - Order Publication	20-09-2010	15-12-2010		Business	J2EE
14	Component	COMA - Order Tracking & Management	20-09-2010	27-01-2011		Business	J2EE
15	Component	COMA - Order Validation	20-09-2010	15-12-2010		Business	J2EE
16	Component	COMA - Order Validation - INT WebSphere	20-09-2010	15-12-2010		Integration	WebSphere

Fig. 10. Partial view of information obtained from normalized project documents

An important result of the project is the time navigation. All architectural views have a time slider associated, in which are marked the moments in time in which there were projects that produced[4] a change in that architectural view. When you move the handle along the slider, and cross a mark, the name of the project that led to changes appears on the left and the content of the architectural view changes in accordance with those changes. An example of this is presented in the top time slider depicted in Fig. 11. If one splits the handle in two, as depicted in the bottom of the same figure, then we specify a time window and the slider presents the name of all projects that have an impact on that architectural view on that time interval, and the contents of the architectural view changes accordingly.

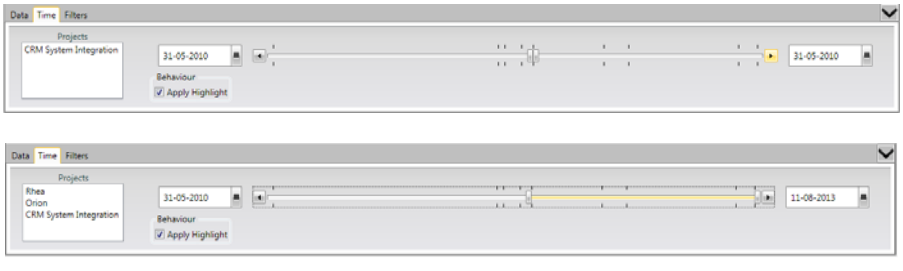


Fig. 11. The time slider specifying a point in time (top) or a time window (bottom)

In figures 12 and 13 we present a Context blueprint of the *Campaign Management Application*. This application is positioned in the centre and the artifacts in its surroundings are those having dependencies with it. In the figure, going clock-wise from top-left corner, the related artifacts types are: Processes, Informational Entities, IT Platforms, Provided Applications, Consumed Applications and Projects. The contents change from left to right-hand side based on the position of time slider, stating, for example, that additional processes the application will support on December 2011.

The blueprints for the roadmaps are not yet completed, but follow a similar philosophy. The current status of each roadmap will be displayed by the colours of

[4] In fact the verb could be in the past, present or future, depending if the handle is positioned in the past, present or future.

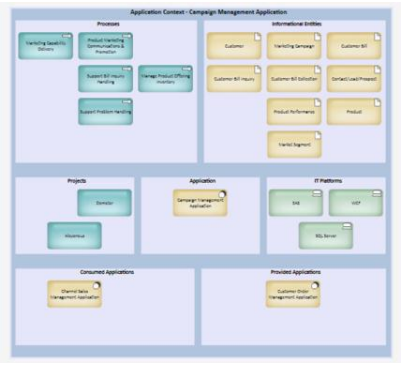


Fig. 12. Visualization for 1-06-2011

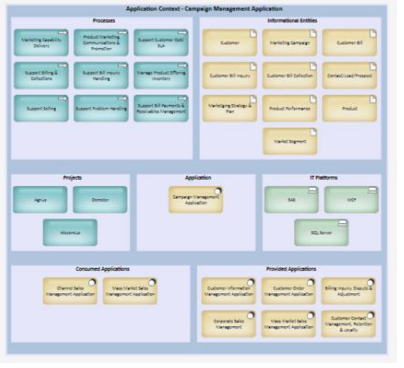


Fig. 13. Visualization for 01-12-2011

each project and, by moving the slider to a given point in time one can see the state of the roadmap at that time (actual state if in the past and foreseen state if in the future).

5 Reflection

The challenge of creating a representation of the enterprise architecture of a changing organization is not small. Making alive/updated representation of that architecture is an even greater challenge. Current practice consists on having modelling tools with which the architects manually design the architectural models of the organizations. Such is valid only for the creative phase, when one is conceiving and designing the proposed future. But the price of using this approach, the models draw by hand, is high, because when they change, someone will have to re-draw them again by hand. And this has been a well-known problem of all organizations.

Thus, in the case of Serasa, we use the opposite approach, and consider the architectural views are generated automatically because it is the only way we can keep them updated .Not wanting to argue that the previous statement is an absolute truth, we claim however that the automatic generation of architectural views is an approach much more suited to the current maturity of organizations, where "architectural chaos" is in place and where the representations are in fact the very limited.

Still, gathering all the required information needed to generate the blueprints is a huge challenge because it forces the organization to have reliable information sources, even if they are scattered and unstructured. In the case of Serasa, it was clear that about 12 of the relevant questions remain unanswered due to lack of information. Serasa will have to create these information sources and adopt practices that keep them reliable. The main difference is that now there are 12 good reasons to justify such effort. The effort associated with extracting information from existing sources of information may not be small, especially for documents and unstructured sources. But once again it is clear that the inherent value.

But the existence of the relevant issues is not only a motivation to do a corporate architectural project. It is also a necessary condition! In fact, when it comes to the

design of the knowledge base meta-model, without a number of questions which are able to set a clear purpose and scope of the project, discussion of the metamodel is a process that tends to have no end, because it is no longer clear what is and what is not important to consider regarding the meta-model.

This confirms the role of enterprise architecture as a tool for organizational change. In fact, the artifacts to be included in the architecture are needed to respond to important issues by those who are responsible for the transformation of the organization. In this approach we are pursuing, we have not found many publications indeed, it may be because one may consider this "cartography" dimension of architecture of a less noble part of the overall theme of Enterprise Architecture, it is our opinion that it is certainly not less important.

Of course there are other important dimensions of enterprise architecture as an instrument of transformation, as for example the architectural principles and policies, but this is a whole new world.

References

[1] Lankhorst, M., et al.: Enterprise Architecture at Work: Modeling, Communication and Analysis. Springer, Germany (2005)
[2] Leal, R.: Navigation model between Architectural Views. DEI, IST-UTL, Portugal (2010)
[3] Open Group: ArchiMate 1.0 Specification: Technical Standard. Van Haren Publishing, The Netherlands (2009)
[4] Open Group: The Open Group Architectural Framework (TOGAF) - Version 9 Enterprise Edition. Van Haren Publishing, The Netherlands (2009)
[5] Sampaio, A.: An Approach for Creating and Managing Enterprise Blueprints. DEI, IST-UTL, Portugal (2010)
[6] Sousa, P., Lima, J., Sampaio, A., Pereira, C.: An Approach for Creating and Managing Enterprise Blueprints: A case for IT Blueprints. In: Albani, A., Barjis, J., Dietz, J.L.G. (eds.) CIAO! 2009. Lecture Notes in Business Information Processing, vol. 34, pp. 70–84. Springer, Heidelberg (2009)

Author Index