

12 SUPPORT FOR ELECTRONIC DATA INTERCHANGE

Pierre Gibon[♦], Jean-François Clavier, Soizic Loison
LICHEN Informatique

EDI technology brings the exchange of trade information via electronic means. Under the PRODNET II project, the EDIFACT standard enables business applications to communicate directly, using to advantage its formal representation and non-ambiguous semantic codification of trade information. Its application to PRODNET provides direct interface between the application database and EDI for easy integration and extension.

INTRODUCTION

The new industrial landscape

For the processing of their operations, most industries and service organizations need to exchange a growing flow of information with external entities. Several factors contribute to this volume increase, such as the complexity of manufactured products, the reduction in production life cycles, the increase in legal or administrative constraints.

Among these contributive factors, one major aspect is the self-complexity reached by manufactured products. No or only a very few industries are able or willing to assume the manufacture of each component part of the products they provide to end-users. Enterprises need to establish reliable, efficient links with their suppliers.

Financial and economic constraints prompt manufacturers to minimize the production cycle and to reduce the costs of production administration. This side of company operations was the initial domain addressed by EDI technology.

The legal and administrative aspects of economic activity are also generative of information (observance of quality norms, security controls, etc.), given impetus by

[♦] LICHEN Informatique, 51-57 Bd de la republique – Bat. 8, 78400 Chatou (Paris), France,
tel.: +33-1-30710702, fax: +33-1-30710807, e-mail: illicom@calvanet.calvacom.fr

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35577-1_37](https://doi.org/10.1007/978-0-387-35577-1_37)

citizen awareness of the need to protect health and the environment and also by economic actors, whenever there is expected competitive advantage.

The main objective of EDI

The goal of EDI is to provide the automated exchange of documents between companies or between companies and administrations.

EDI is only part of the global effort made by companies to weave a network of competencies. EDI can be included in a set of recent technologies, where Concurrent Engineering is probably the main element. An effort was made under PRODNET to implement the convergence between STEP and EDI technologies, for technical and administrative information.

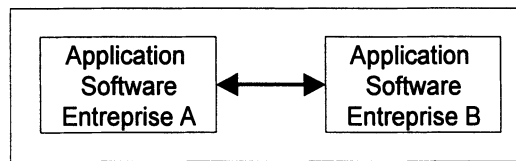


Figure 1 – EDI for direct connection between software applications

The automation of document exchange, as seen by EDI, is not related to the ability of computers to exchange data - this topic is covered by communication technologies - but to the direct exchange of information and its recording by software applications without the need for human intervention.

One main key to the automation of data exchange is the guarantee that data will be exchanged in an understandable format with no ambiguity. The role of standards like EDIFACT is to provide a convenient frame that overcomes this problem and gives set rules for:

- character sets,
- data structures, organization, length,
- semantic value of data.

and contains guidelines for its implementation.

Applicable character sets and data structure organization can be considered as standard features when defining norms. For these two features EDIFACT used already existing norms, but it was semantic codification that required the greatest effort on the part of those organizations which led the EDI projects and which probably brought significant added value to electronic data exchange.

BASIC REQUIREMENTS FOR EDI

The first requirement for Electronic Data Interchange is, evidently, the availability of computers. Nowadays most small and medium-sized companies have computerized their own administrative activities. Projects like PRODNET-II relate to the real needs these smaller entities.

A further requirement is the availability of communication links between computers. This initially refers to the existence of telephone equipment or networks able to support data transfer with sufficiently effective flow rates and of sufficient quality (we tend to forget that their general availability at a reasonable cost was only reached by the middle 1990's).

The last basic requirement is the availability of VANs -Value Added Networks - and the availability of communication stacks in order to connect to these VANs.

While the first two requirements can be easily understood by any reader, the importance of the third prerequisite perhaps requires some explanation.

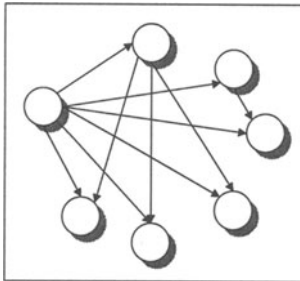


Figure 2 – Direct connections

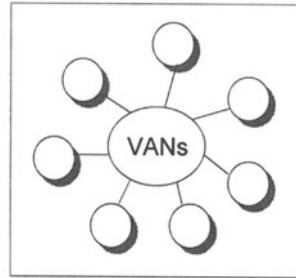


Figure 3 – Connections using VANS

Establishing a communication link between two computers is not a trivial task, and requires a fair amount of skill on either side that needs to be simultaneously available. It is easy to understand that the situation described in figure 2 can rapidly become unmanageable.

Initial response to this problem was given by some networks, such as PSTN (Public Switching Networks) based on X.25 norms. For these kinds of networks, there is no longer any “physical” link between two communicating partners. However the simultaneous availability of each party is still required for the transfer of data.

A second level of response brought real independence between companies, when data came to be exchanged through mail technologies, these being mostly X.400 mail services, Internet SMTP protocol and Odette-FTP, all of these networks providing intermediate storage for exchanged data.

When interconnecting applications, whereby a given application is able to directly incorporate data received from other computers, similar problems quickly arise. It is not difficult to interconnect two applications and adapt each one to the requirements of the distant application. But it is left to the reader's imagination to grasp the situation that arises with a global information system when applications with specific formats must be adapted to each other and when hundreds of links need to be established.

One method of achieving a stable, operative system consists of federating all the applications to be interconnected around a common information model.

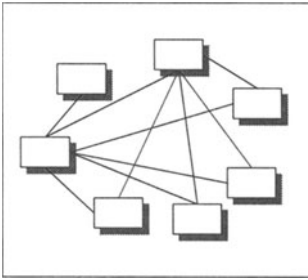


Figure 4 - Different data models between applications

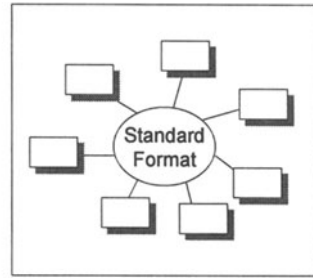


Figure 5 - The same data model for all applications

Many workgroups have focused their efforts (mainly in the 1980's) on developing common information models. EDIFACT is the outcome of some of these groups.

A BRIEF HISTORY OF EDIFACT

The EDI concept came into being at the end of the 1960's, when communication standards between computers became available. Many of the world's workgroups started offering models for data interchange. Most of these groups concentrated mainly on the exchange of trade information. Others, such as ODETTE, promoted the exchange of technical information. Up until the 1990's, EDIFACT technology was more exploratory than operational, and, in many cases, EDI projects were experimental ones. At the same time, however, some economical sectors felt an acute need for efficient tools in electronic trade and developed their own standards. Due to the complexity of the task in laying down standards, some working groups started working on the basis of a national or regional area, others specialized in dedicated industrial sectors, in order to provide these sectors with efficient tools in the shortest possible time. The result was an overwhelming number of standards.

A further consequence can still be seen in some companies which are equipped with EDI and various other software or systems, for they exchange data with several different sectors (for example automotive and distribution). Sometimes companies or governments lay down their own standard or systems.

Of all the standards that were developed, it can be considered that the ASC X12 (American), GTDI and ODETTE standards were the most successful encompassing important groups of economical actors.

The initiative of designing an electronic trade standard started in the United Kingdom in the year 1974. In April 1975, "an ad hoc meeting was held in Stockholm with the aim of setting out basic principles and of defining requirements for trade data interchange" [UNTDID- D100_d.htm]. A first result, close to the actual EDIFACT features was obtained in 1981: "Guidelines for Trade Data Interchange" (GTDI).

As national or sectoral standards were growing in many countries or industrial areas, new working groups were developed in order to reach a possible convergence between these standards. The UN-JEDI group was created in order to obtain

convergence between the most developed standards, the ASC X12 and the GTDI. In September 1986, the name of UN-EDIFACT was born. The first directories were published in 1990.

In industry, a few projects had been efficiently started by the early 1990's. These projects can be considered as pilot projects, and were, in many cases, supported and promoted by governments or economical groups. The full deployment of EDIFACT started in the middle of the 1990's, and since it was adopted by most of the leading industries in developed countries, it can now be considered that the "Grand Project" imagined by its pioneers in 1974 has reached its ambitious original goal.

THE UN-EDIFACT STANDARD

EDIFACT is the acronym of "Electronic Data Interchange For Administration, Commerce and Transport", UN being the acronym of the United Nations which is in charge of this standard.

If we consider EDIFACT within the world of norms and standards, it certainly holds a special position. Firstly, its realm of application is highly mobile. It is out of the question to hold back developments in commercial relationships because the supporting framework is too rigid. The EDIFACT standard is designed to support the constantly evolving area it covers and provides regular versions and releases. Unlike other norms or standards, the publication of a new version does not make former versions or releases obsolete which still remain available.

EDIFACT does not directly specify the format of data which parties can exchange by electronic means (it uses the recommendations given by existing norms for this purpose, mainly ISO9735). EDIFACT gives recommended structures for commonly used messages, specifying which information is mandatory and which is conditional, providing a frame to adapt these data structures to the real needs of the relevant trade, industry or administration.

The most important characteristic of EDIFACT is probably its semantic codification of information. In an order message, several dates are to be found. An important issue is to determine with certainty the function of these dates (date of the order, delivery date, payment date, etc...). Finding a solution to this problem is probably the most time-consuming aspect when designing a software application.

Definition of UN-EDIFACT

UN-EDIFACT: United Nations rules for Electronic Data Interchange For Administration, Commerce and Transport. These rules comprise a set of internationally agreed standards, directories and guidelines for the electronic interchange of structured data and, in particular, the related documents for trading in goods and services, between independent computerized information systems. [*Le Commerce par ordinateurs* - B. Stoven p224]

Recommended within the framework of the United Nations, the rules are approved and published by the UN/ECE in the United Nations Trade Data Interchange Directory (UNTDID) and are maintained under agreed procedures.

UNTDID includes:

- the EDIFACT syntax rules (mainly based on ISO 9735),
- Message design guidelines,
- Syntax Implementation guidelines,
- the EDIFACT Data Elements Directory,
- the EDIFACT Code List ,
- the EDIFACT Composite Data Elements Directory,
- the EDIFACT Standard Segments Directory,
- the EDIFACT United Nations Standard Messages Directory,
- Uniform Rules of Conduct for the Interchange of Trade Data by Teletransmission,
- Explanatory material, as appropriate.

EDIFACT bases its normative rules on the following norms [UNTDID - D422_D.97A]:

- ISO 31/0-1981 General principles concerning quantities, units and symbols
- ISO 646-1983 Information processing - ISO 7-bit coded character set for information interchange
- ISO 2382/1-1984 Data processing - Vocabulary - Part 01: Fundamental terms
- ISO 2382/4-1987 Data processing - Vocabulary - Section 04: Organization of data
- ISO 6523-1984 Data interchange - Structures for the identification of organizations
- ISO 6947/2-1983 Information processing - Coded character sets for text communication
- ISO 7372-1986 Trade Data Elements Directory, (UNTDDED)
- ISO 7498-1984 Open Systems Interconnection - Basic Reference Model
- ISO 8859-1987 Information processing - 8-bit single byte coded graphic character sets 1)

Message Design and Subsets

An EDIFACT message is a sequence of segments. A segment is a set of logically grouped information. Within a message, the presence of each segment can be mandatory or optional.

Segments can be assembled to form a “segment group”. A segment group may be made up of several segments and segment groups.

UNSMs (United Nations Standard Messages) are designed to be “universal”, so that they comply with most sectorial or national possibilities. Messages are not directly usable in their full form, but must be adapted by users: two companies wishing to use an EDIFACT message to exchange some commercial document make a “projection” of a standard message and design a simpler message corresponding to their needs. These reduced messages are called SUBSETS.

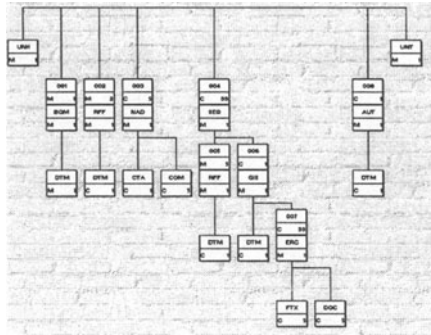


Fig.6 – A message composed of segments and of groups of segments

Segments or segment groups are, under the UNSM definition, qualified either as M (mandatory) or as C (conditional i.e. optional). Some industries may re-qualify some conditional element as “required”.

There are two kinds of segments:

- services segments,
- application data segments.

Service segments are used to identify messages (types of messages, version of EDIFACT syntax used ...).

A segment is identified by a mnemonic made up of three letters (example: DTM segment: date indication). The DTM segment is defined once in the EDIFACT but used several times in most of the messages. Standard segments are the basic material used to build the messages. They are referenced in the EDSD, United Nations Standard Data Segment Directory.

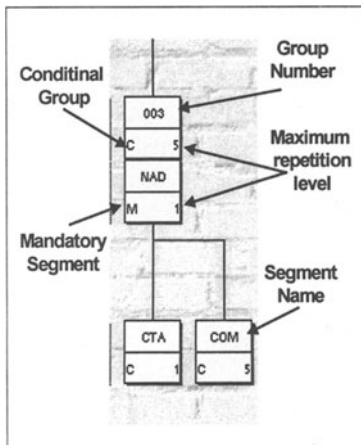


Figure 7 – Segments and segment groups

Segments and Data Elements

Segment definition.

A predefined and identified set of functionally related data element values which are identified by their sequential positions within the set. A segment starts with a segment tag and ends with a segment terminator. It can be a service segment or a user data segment. [UNTDID / D300_D.97A]:

Less than 150 segments are defined by the UN-EDIFACT, which are sufficient to describe most commercial information elements.

MEA	MEASUREMENTS		
	Function: To specify physical measurements, including dimension tolerances, weights and counts.		
010	6311	MEASUREMENT PURPOSE QUALIFIER	M an..3
020	C502	MEASUREMENT DETAILS	C
	6313	Property measured, coded	C an..3
	6321	Measurement significance, coded	C an..3
	6155	Measurement attribute identification	C an..17
	6154	Measurement attribute	C an..70
030	C174	VALUE/RANGE	C
	6411	Measure unit qualifier	M an..3
	6314	Measurement value	C an..18
	6162	Range minimum	C n..18
	6152	Range maximum	C n..18
	6432	Significant digits	C n..2
040	7383	SURFACE/LAYER INDICATOR, CODED	C an..3

Figure 8 – Example of a segment definition – segment MEA

Data Elements

As can be observed in the segment description above, a segment is made up of a collection of more elemental data, called Data Elements, which Elements can be grouped into intermediate structures named Composite Data Elements.

Data Element:	a unit of data for which the identification, description and value representation have been specified. [ISO 9735]
Composite data element:	a data element containing two or more component data elements. [ISO 9735]

Figure 9 – Data Element and Composite data element definition

Data elements and composite data elements form a “dictionary”. Their identifications follow explicit rules. The definition of the elements, and their representation formats, are specified by ISO-7372, UN-EDED (EDIFACT Data Elements Directory) and UN_EDCD (EDIFACT Data Composite Directory).

The information within an element of data, is either explicit (text or value), or coded. Various code lists are available and can be accessed in the EDCL (EDIFACT Data Code List) or refer to other norms or standards (the coding of EDIFACT 3207 element, Country Codes refers to norm ISO 3166).

Codification of data elements:

Data elements relate either to codified information, or to plain text information. The names of the data elements follow the rule:

- odd data element numbers have a codified meaning,
- even number data elements relate to information supplied by the application.

In the DTM example (Fig. 6), the format of the date inside the 2380 is specified by the 2379 date/time format qualifier. 70 code formats are specified in the 97A release. In our example, the value 102 given for 2379 specifies that the supplied date is under the format CCYYMMDD.

2379	Date/time/period format qualifier	[B]
Desc: Specification of the representation of a date, a date and time or of a period.		
Repr: an..3		
2	DDMMYY	
Calendar date: D = Day; M = Month; Y = Year.		
3	MMDDYY	
Calendar date: M = Month; D = Day; Y = Year.		
101	YYMMDD	
Calendar date: Y = Year; M = Month; D = Day.		
102	CCYYMMDD	
Calendar date: C = Century ; Y = Year ; M = Month ; D = Day.		
103	YYWWD	
Calendar week day: Y = Year ; W = Week ; D = Day Week number 01 is always first week of January Day number 1 is always Monday		
....		
....		
814	Working days	
Number of working days.		

Figure 10 – [UNCL_D97A] Code list value (partial)

Most codified data elements have their entries in the UNCL file, published by the UNTDID. When an international codification exists (ISO or other bodies), the UN-EDIFACT refers to this codification.

Interchanges

The Interchange is a structure allowing message encapsulation and grouping, and the referencing of transactions. When transmitted, EDIFACT messages are encapsulated in an Interchange. An interchange may contain one or more EDI messages.

A sub-structure, the section, may also be used in order to group part of messages within an interchange, called functional groups and managed with the UNG and UNE segments. This sub-structure is mainly used to ensure compatibility with other standards such as ASC X12. The EANCOM, in its guidelines does not recommend their use.

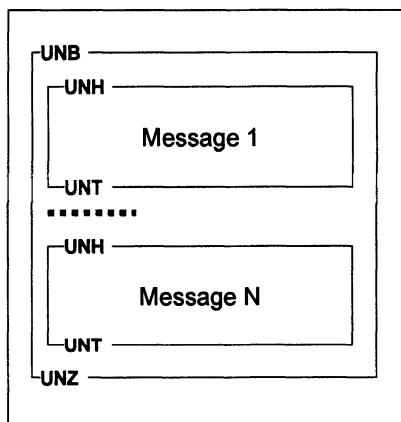


Figure 11 – Interchange envelope

Interchange:
 communication between partners in the form of a structured set of messages and service segments starting with an interchange control header and ending with an interchange control trailer. [ISO 9735]

Figure 12 – Interchange definition

CONTRACT ENVIRONMENT

Each time a new information vehicle is used for trade and commercial relations, the issue is immediately raised of the contractual and legal value of this new means.

The importance of this issue is subject to the consideration that must be given to the trade environment in which EDI can be used. The natural area of EDI application is used by companies who have strong, permanent trade links, for instance automotive manufacturers and their suppliers, where several orders or delivery instructions can be exchanged per day. In this business sector, common trade instructions have long been exchanged without any official and legal considerations. The contractual side is negotiated, not for a single order, but for a global trade exchange protocol over long periods. Generally both parties have identified potential problems and have already agreed to solve them in the most efficient way.

The UN-EDIFACT gives an example of a standard interchange agreement in its directories.

IMPLEMENTATION PROCESS

Why use EDI?

A number of good reasons are needed to warrant the implementation of EDI technology in a company or an administration. All these reasons could lead to cost-effective gains for the company.

The following are among the major aspects to be examined:

- profits to be obtained with the automated administration of trade documents,
- potential quality gains (loss of documents, error-free administration),
- time constraints in the processing of trade documents.

For large companies, the first aspect may suffice to decide on implementing this technology. With EDI, large companies can divide the individual cost of producing trade documents by 5 or more, and often obtain a return on investment in less than 6 months (the situation may be different for small and medium-sized companies). One obstacle to profitability is the general necessity of maintaining both EDI and non-automated circuits. Larger companies often have to find other arguments to convince their suppliers or customers to use EDI.

Improvement in quality is an important motivation to use EDI, both in large and small industries. The level of human error in the processing of information is difficult to eliminate. The cost of EDI investment can soon be recovered through a

reduction in costs incurred through nonconforming quality, even in small enterprises. Sometimes, quality considerations are not assessed from an economical viewpoint, but solely in relation to the image the company wishes to convey.

Many supplier companies are involved in just-in time production cycles. For these companies, EDI is just a necessary tool.

Some other features can be put forward to promote EDI technology. One may be of interest to small and medium-sized companies in that with the establishment of EDI links customers can concurrently install global supply plans over long periods enabling the company to plan ahead.

Sometimes, the success of EDI can be attributed to quite different reasons. In Spain, for example, it has been used by third parties to certify that invoices had been transmitted to the customer. Companies have seen this service as a good means to reduce delays in invoice payments. The result is that 60% of EDI traffic in Spain is related to invoices [AECOM – 1996], although invoice messages represent less than 10% of EDI traffic in France and Germany.

Implementation methodology

It is possible that specification of the data to be exchanged, and schedules for exchanges represent the most complex part of an EDI project. As small and medium-sized companies are rarely the initiators of EDI projects, most of these tasks are already completed when they set up EDI links, either by their major customers or suppliers, or by financial bodies.

Smaller companies may encounter some major difficulties when they decide to equip themselves with EDI technology. The first is related to their own computing environment and its capacity for the automatic incorporation of trade information from its partners. A second difficulty is the ability to manage the related telecommunication infrastructure. A last point is the necessary adaptation of the people working in the company, and the difficulty in obtaining staff trust in this new way of working. Two opposite attitudes must be avoided: outright rejection of automation, or an inconsiderate level of confidence leading to a lack of supervision and control.

As in most new areas, the product offer is not yet truly stabilized, and choosing an adapted product may be difficult

Cost evaluation

EDI implies several levels of skill:

- modification of software applications used by the company,
- understanding of EDI technology,
- internal network architecture,
- telecommunications stacks and network connectivity.

The cost of EDI installation includes not only the price of licenses and computer hardware, but also software consultancy fees for the integration of all these new technologies. This last point can be estimated as the main cost source.

PRODNET objectives

One of the main objectives of the PRODNET project is to enable smaller companies to access a set of technologies (EDI, STEP, WorkFlow, Data Bases, communication) that are ready-integrated, and to provide simple interfaces for applications, in order to cut integration time considerably shorter and to eliminate any risks of software incompatibility.

PRODNET FUNCTIONAL REQUIREMENTS

Basic requirements

Below is a schematic representation of an EDI subsystem:

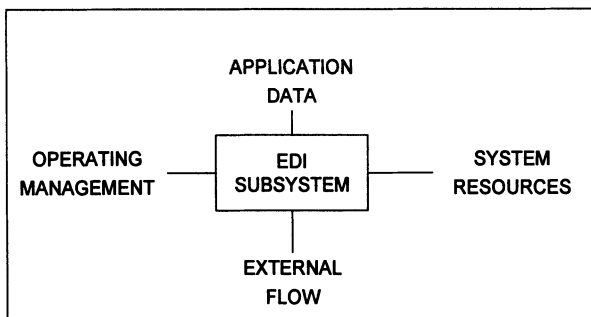


Figure 13 – EDI subsystem functional relation

The main function of an EDI system is to translate application data into an EDI format, and, conversely, to parse received EDI messages in the application format.

Building EDI messages

As previously described, EDIFACT rules are based on the Interchange concept. EDI messages are not randomly sent to their recipient, but enveloped in a structure called “interchange”. An interchange may group several EDI messages relative to a same recipient.

The basic functionality attached to the translation process can be summarized as two main functions:

- EDI message building,
- Interchange encapsulation of built EDI messages.

Parsing EDI messages

Parsing EDI messages allows the EDI subsystem to process EDI interchanges received from the network and to deliver the content to the application in the appropriate format.

Three sub-functions operate in this process:

- syntax analysis,
- checking the originator of the interchange and its rights,
- parsing the messages and delivering them to the application.

Syntax analysis verifies that the received messages conform to general EDI syntax rules and checks the message received against the subset format negotiated with the originator.

GENERAL SYSTEM REQUIREMENTS

A number of system requirements can be identified for the integration of EDI functionality.

Directory services

EDI needs to store partner-dependent configuration parameters. These parameters can be stored in a partner database or some equivalent structure containing EDI related information. As PRODNET commercially targets smaller companies, a configuration interface, with a reduced set of parameters was designed, in order to avoid unnecessary end-user to face parameter constraints, following EANCOM recommendations. The EDI subsystem is able to manage the Virtual Enterprise environment; a partner can be member of several VEs. As two VEs may define different EDI formats for the same type of message, it is possible to address the same partner, in different VE environments, with different message formats.

Alarm/event reporting

Logging functions are critical in the trade information exchange process, and are important criteria for choice of product. Within the PRODNET environment, the role of the EDI subsystem is to clearly identify the possible difficulties that could be encountered. A PRODNET Co-operation Layer (PCL) centralizes all events related to the circulation of documents and the services supplied by the subsystems.

Time management

The LCM module handles this function, conventionally supplied with EDI systems. PRODNET architecture brings the advantage of having time control over global document flow.

Tracking

A common need in EDI systems is to allow operators or applications to browse back events within the system, for example to give confirmation to a partner that an EDI message has or has not been sent.

Standard system management facilities

Some features are common to any computer system management such as purges, backups, and system crash recovery procedures. Some interactions with work packages may be identified, for example configuring parameters for the preservation of information, or signaling to a given application the necessity to rebuild database indexes after a crash...

EDI IMPLEMENTATION IN PRODNET II

Architecture of the EDI subsystem

The EDI subsystem is a subset of **Visual Commerce®**, the most recent EDI product marketed by LICHEN Informatique, which is the outcome of PRODNET II research combined with private investments made by the company with its own funds. The software package for the implementation of the EDI subsystem can be divided into three items of software:

- The Mapping process (EDI OLE SERVER),
- The interface with the LCM. (PCL-EDI AP),
- The Mapping configuration tools (EDI Subset Editor).

The EDI OLE server can be considered as the kernel of the EDI Subsystem. This kernel is independent from any host system in terms of functionality. Its function is to build EDI messages and interchanges from application data, and to provide application data from received interchanges. The EDI kernel is designed to provide EDI services in a multi-session and multi-transaction environment, being able to manage the full features of EDIFACT as well as other popular standards. These features, that were not mandatory in the PRODNET-II project, are partly necessary for some industry environments. The internal architecture of the EDI kernel is mainly based on a multi-line organization, with one main line managing the global activity of the EDI kernel, other lines being started for each transaction or session, depending on various criteria such as the type of services required or the profile configured for an application.

The EDI OLE Server presents an interface to applications, with session and transaction management facilities.

The PCL-EDI AP implements the Local Management Co-ordination interface designed for PRODNET. The main function of the PCL-EDI AP is firstly to conduct macro-EDI services supplied to the LCM, EDI building and EDI parsing services and secondly to interface the Distributed Information Management System (DIMS) in order to store and retrieve information inside PCL data organization. The PCL-EDI AP contains also on-line configuration and supervision tools.

The other EDI components, which are optionally supplied to the end user, are EDI configuring tools (message design, subset editor).

With the EDI Subset Editor it is possible, in a graphical environment, to design subsets of messages, to produce relevant documentation and to specify mapping between EDI messages and application data.

The architecture of the EDI subsystem can be schematized as follows:

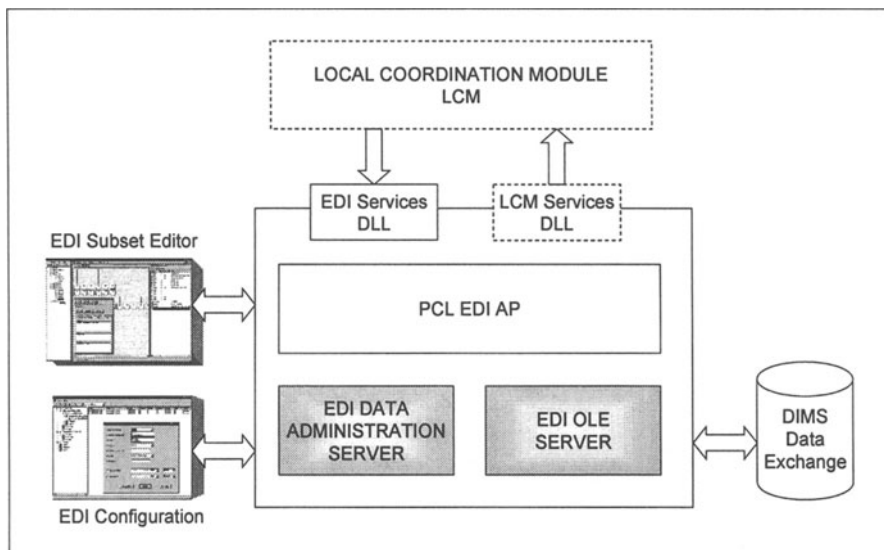


Figure 14 – PRODNET II – EDI Subsystem architecture

The PCL-EDI Application Process communicates within the PCL through the LCM interface and, if necessary, uses a number of services available in the PCL such as Database services.

The PCL-EDI-AP uses the services of the EDI OLE server, via the COM technology provided by Windows NT, for the completion of EDI tasks. The choice of COM technology (Component Object Model) was based upon various criteria, in particular, the potential upgrading of the server to DCOM or ORB technologies. This architecture enables the EDI package to be split into two layers:

- the EDI service layer that meets EDIFACT norm requirements,
- the application layer that provides the PCL environment.

In principle, the EDI layer is independent from the environment in which it operates if we except the need to access an appropriately designed database.

The role of the Application process is to customize EDI processing services to meet the requirements of the hosting system.

To design mapping, an off-line tool, **EDI Subset Editor** is used to configure the pairing of application databases with EDI messages. This tool is used when initially configuring the EDI module, but it can also be used during the process for the dynamic reconfiguration of pairing rules between “in-house” data and EDI messages.

Supporting technology

Several technology keywords are attached to the design of the architecture:

- Database oriented software,
- 32 bits technology,
- Object oriented programming,
- Application interfaces based on COM technology,
- Graphical User Interface.

These main technical features (if we exclude COM technology) were practically a pre-requisite for the PRODNET project, whose natural goal was to demonstrate the concepts of virtual enterprise in an up-to date technological environment.

Database technology

Database environment, is the natural solution for company management of data and applications: all databases use a common language to define data and operations on these data, such as the normalized SQL92. In today's highly competitive environment, we have even reached a situation in which most suppliers of popular databases also offer tools for the simultaneous management of their own databases and those databases designed by other providers. Some operating systems also provide common interfaces able to work with any data pounds, interfaces provided with the OS or by database suppliers or third parties. The most known of these database interfaces is ODBC, DAO being the emergent successor. Needless to say, not all database providers give the same level of services for data management. One of the major features provided by databases is the maintained consistency of data: the transaction concept, which preserves the atomicity of data operations. Access rights can be different for a number of users working on the same set of information. A more extended description of database characteristics can be found in the chapter on DIMS. The product chosen for PRODNET II offers an interesting number of available functional operations, and the strength of the EDI subsystem is mainly based on these features.

A few years ago, it would have been foolhardy to attempt to run EDI software on a database, but improvements in computer performance has now made this a possibility. This was one of the challenges facing the project. As hardware suppliers continued to increase performance levels, the questions that could have been raised as to this technical choice no longer apply.

32 bits technology

In order to maintain the potential growth of the EDI subsystem, all software items are written in 32 bits. This is in line with state-of-the-art software development tools, and is a pre-requisite to allow software to evolve together with potential advances made by hardware platforms and related operating systems.

Object oriented programming

Throughout the project, the software was re-designed so that it could be programmed following an object oriented programming technique. This programming method is fairly well suited to EDI techniques if we consider that EDI messages are built from a number of determined elements, in a hierarchical organization. A strong motivation for making this technical investment was the advantage we saw in the promises of COM technology that we decided to adopt.

COM based interfaces

With COM technology it is possible to publish an interface that the client application may import and use. The EDI OLE server implements the Automation COM protocol. This Automation enables a client, in this case called the Automation controller, to access the services of the Automation server as if they were an object in its own application, without any consideration of communication constraints or development, these being taken in charge by the operating system. This technology is widely used by Windows applications.

Graphical User Interface

The EDIFACT standard sets forth rules for the graphical representation of messages. The EDI subsystem included in the PRODNET kernel conforms to recommendations. Some additional features are also included when they can help the designer of the subset. The EDI Subset Editor can be considered as a software development tool that provides a frame for fast mapping design with due respect to standard structured programming rules and facilities for the management of the development and production environment. The graphic workshop is not an integrated part of the PRODNET Co-operation Layer, but can be bought separately from the editor or its licensees.

PCL-EDI-AP main functions:

The PCL-EDI Application process carries out the following main functions:

- Building EDI messages and interchanges,
- Message identification,
- Parsing interchange,
- Self-management within the PRODNET-II environment.

Building EDI messages and interchanges, and parsing interchanges can be considered as workflow operations. Identifying messages is a service the EDI process supplies to the LCM.

Within self-management functions, functions such as process initialization can be included (connection to the LCM), activity scheduling, error signaling and all types of processing appropriate to PCL global management.

Building EDI messages and interchanges

The *EDI_BuildMessage* service converts a technical or commercial document (i.e. an order) into an EDI format (EDI message embedded in an EDI interchange).

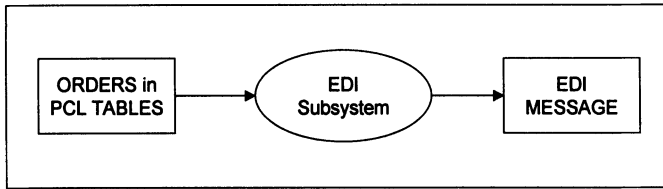


Figure 15 – Build EDI message

When the LCM gives indication to the PCL-EDI-AP that commercial or technical-related information must be sent, the EDI_BuildMessage function executes the following macro-operations:

- checks the validity of the incoming information,
- builds the message,
- builds the interchange,
- signals to the LCM the availability of an EDI message to be sent.

The building process depends on the profile configured for the pair represented by the application and the partner. From one application data pound, several message types can be built according to the interchange agreement negotiated between the parties. The EDI subsystem enables one system to support several VE or non-VE environments simultaneously.

Message Identification Service

The *EDI_MessageIdentification* service analyses the submitted document in order to determine whether or not the document is a valid EDI message.

This service can be provided directly to the process (PCI) that needs to know if the information is an EDI message. The Message Identification service is a synchronous service. Its result indicates whether or not the “document” submitted was a message identified as an EDIFACT interchange.

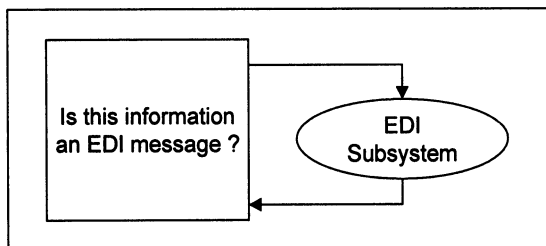


Figure 16 – EDI Message Identification service

In the Message Identification Process, a first syntactic control is conducted on segments UNA, UNB and UNZ of the interchange envelope.

Parsing Message Service

The *EDI_ParseMessage* service converts an item of EDI information into a technical or commercial document in PCL format.

As the EDIFACT norm manages the facility for receiving several messages in a single interchange – which is widely used in industry, one EDI Interchange received may lead to the generation of several commercial or technical documents.

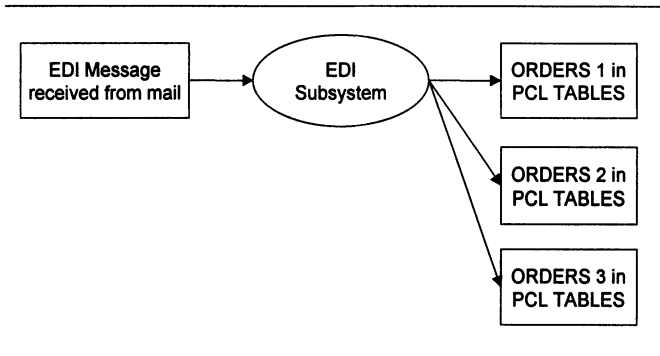


Figure 17 –Parsing several messages in an interchange

This is a key point for the design of the EDI subsystem: EDI cannot be considered simply as a service, but also as an activity, according to Work-Flow terminology. The Parsing Message function analyses the “document” and extracts all the EDI messages contained in the “document” to be parsed. The parsed information, if no fatal error is detected, is stored in the DIMS database. As a result of this operation several “documents” may be submitted to the LCM.

FUNCTIONAL INTERFACE DEFINITION AND REQUIREMENTS

This chapter describes for each of the services, the *Build Message service* and the *Parse Message Service*, the necessary input parameters to process the services and the resultant parameters needed for implementation of the following activity.

Building Message Flow

The process for building EDI messages can be divided into two stages:

- a first stage of selection of the EDI subset and mapping program to be applied, according to the parameters transmitted by the Work-Flow,
- processing of the data in the DIMS database according to the rules determined during the first stage.

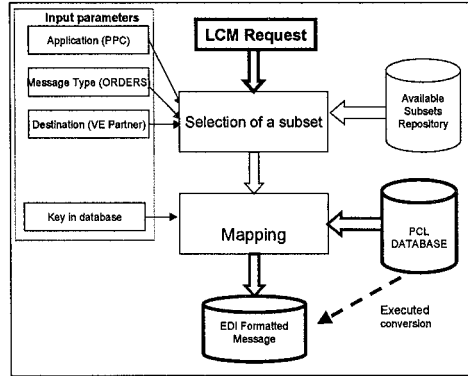


Figure 18 – Building message flow

When an application (PPC) has to send commercial messages (ORDERS), the format of the EDI messages may vary between the various partners, for instance between a partner A in VE 1 and a partner B in VE 2, and two distinct subsets and mapping rules must be applied.

Parsing Message Flow

When an EDI message is received in a system, and if this message is valid, this message contains the information that permits:

- Identification of the sender of the EDI message, the type of document and the process to be applied.
- Processing of the data in the DIMS database according to the rules determined during the first stage.

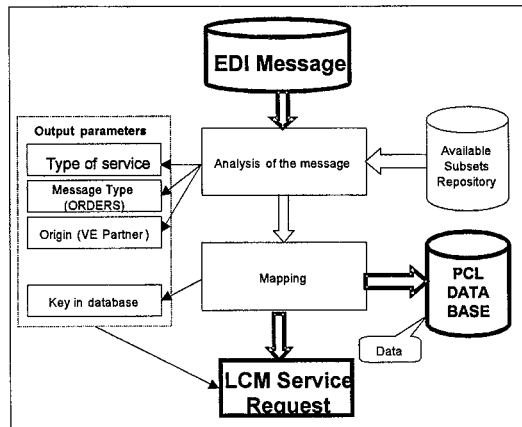


Figure 19 – Parsing message flow

Validation of a message received from a partner consists of a number of controls: checking the structure of the interchange envelope, control of sender and recipient

identities, search in the configuration base for the association configured for the partner and the local application.

PROSPECTS

It can be considered that EDI technology has just reached the status of a commonly used tool, led chiefly by the major industries and services. EDI technology is now undergoing further concentration with an accelerated tendency to merge the existing standard into the UN-EDIFACT standard. One of the most important events in this area was the decision taken by ODETTE to change over the entire automotive industry to EDIFACT by the start of the year 2000. The PRODNET project presents some exploratory features, by providing STEP files for administration data exchange using EDIFACT CONDRA messages.

In 1997 the EFI (Electronic Form Interchange) concept was developed mainly based on HTML technology and WEB services, positioned as the competing technology that could supersede EDIFACT. Initial implementations of EFI technology confirmed that EFI was not an alternative to EDI, but a complementary tool, able to provide Electronic Commerce facilities to looser commercial links. More recent terminology has partitioned the commercial environment in two classes: B-to-B (Business to Business) and B-to-C (Business to Consumer). It can roughly be summed up, in the state of current technologies, that EDIFACT is still the privileged domain for B-to-B, while EFI is more dedicated to B-to-C or occasional trading. In most industrialized countries, EDI B-to-B covers about 40% of trade information with an estimated growth of 30% a year. B-to-C only really got off the ground over the last two years, and experts expect it will show explosive growth with the spread of the Internet.

On many occasions alternative solutions to EDI, in its EDIFACT form, have been put forward. Some offered interesting features, such as the replacement of the ISO-9735 representation by a model based on the more efficient ASN.1 representation. In the face of the slow spread of EDI, mainly in small and medium-sized companies, the CEN/ISSS Electronic Commerce Workshop European XML/EDI Pilot Project was set up, based on emergent XML technology, which provides more potentiality than HTML. This latter approach offers very attractive features, not only through its own technology, but endeavoring at the same time to integrate the results already obtained with EDI.

Another aspect calling EDI into question in its current EDIFACT form concerns the use of Value Added Networks instead of the Internet, and the costs this involves. The main reasons for these costs are attributable to the ability of VANs to enter into commitments for contractual service quality. Now some Internet providers offer quality levels that are suitable for trading requirements, but at extra cost. While free EDI may be difficult to attain, using standard tools for communication can be considered as an important advantage for smaller companies.

Acknowledgements

This work was funded in part by the European Commission, Esprit programme. The authors also thank the valuable contributions from their partners of the PRODNET II consortium: CSIN (P), HERTEN (BR), MIRALAGO (P), University of Amsterdam (NL), Universidade Federal de Santa Catarina (BR), Universidade Nova de Lisboa (P), Lichen Informatique (F), ProSTEP (Germany), Uninova (P), ESTEC (P).

REFERENCES

1. UN/ECE –United Nations Trade Data Interchange Directory (UNTDID) 1997/1998/1999
 - PART 1 INTRODUCTION
 - PART 2 UNIFORM RULES OF CONDUCT FOR INTERCHANGE OF TRADE DATA BY TELETRANSMISSION (UNCID)
 - PART 3 TERMINOLOGY
 - PART 4 UNITED NATIONS RULES FOR ELECTRONIC DATA INTERCHANGE FOR ADMINISTRATION, COMMERCE AND TRANSPORT
2. Bernard STOVEN: *Le Commerce par Ordinateur*.- SIMPROFRANCE/SIPROCOM 1990
3. AECOC – CERTIFICACION TECNICA AECOM/EANCOM – 1996
4. GENCOD – EANCOM – Silène V3 – EANCOM 1997 – May 1998