Faculdade de Engenharia da Universidade do Porto

Mestrado Integrado em Engenharia Informática e Computação

Applying PLCS to EDMS/MTF at the European Organization for Nuclear Research

Master Thesis – MIEIC 2007/08

Mário Miguel Terra Pinheiro Fernandes Pereira

FEUP Supervisor: Prof. Jaime Villate

CERN Supervisor: Eva Sanchez-Corral Mena

April, 2008

"The rest of your life starts right now....so do your best, have fun and enjoy it!"

Abstract

The object of this document is to present a Master Thesis produced at CERN, the European Organization for Nuclear Research, in the context of the Mestrado Integrado em Engenharia Informática (MIEIC).

Today the need for sharing and integration of information is well known but successful implementations are not that common. Even with the use of Internet technologies, information sharing is to a large extent focused on product procurement and commerce. The sharing of information between partner companies over the product life cycle is still pretty limited. Often a major barrier is the difficult task of integrating PLM, ERP, and Product Support systems across partners in an ever-changing market.

The industry's answer to the need to ease data exchange took the form (1984) of a standard for data representation, STEP, *Standard for the Exchange of Product* model data which provides a neutral mechanism of product data exchange between product life-cycle systems.

STEP was limited to the design phase of the product life-cycle. An extension of the STEP standard covering the whole product life-cycle was developed in 1999, called Product Life Cycle Support, also known as Application Protocol 239.

The CERN Engineering Data Management System (EDMS) is the Organization's central technical document and equipment management system. This system is being used to manage all data related with the development of the Large Hadron Collider (LHC). A project with such dimensions and complexity involves hundreds of companies in tight collaboration and data exchange is constant. At CERN, the EDMS/MTF system serves as a repository where all data concerning this project is stored and managed.

However, for the LHC project, CERN imposed the methods and tools to be used in the collaboration with the external companies and collaborators. Data input in EDMS/MTF follows certain specifications and file formats, some of them different from the companies' internal business processes.

This document introduces the STEP and PLCS standards, providing a detailed view of their architecture, aim, scope and implementation guidelines. This work also contemplates the hypothesis of an implementation of the PLCS standard in EDMS/MTF, considering its advantages, issues and hints on the technology that could be used.

Resumo

O objectivo deste documento é apresentar a Dissertação de Mestrado produzida no CERN, Organização Europeia para a Pesquisa Nuclear, no context do Mestrado Integrado em Engenharia Informática (MIEIC).

Actualmente, a necessidade de partilha e integração de informação é bem conhecida, no entanto, implementações bem sucedidas não são muito comuns no mercado. Mesmo com a utilização de technologias relacionadas com a a Internet, soluções de partilha de informação estão normalmente relacionadas com *procurement* e *e-commerce.* A partilha de informação ao longo do "product life cycle", entre empresas num projecto de colaboração, ainda é muito limitado. A maior barreira é muitas vezes a dificuldade em integrar sistemas PLM, ERP e de "Product Support" de todos os *partners* num mercado que está em constante evolução.

A resposta da indústria à necessidade de facilitar a transferência the informação entre empresas tomou a forma de um *standard*. Em 1984, foi criado o STEP, *Standard for the Exchange of Product model data*, que fornece um mecanismo neutro de troca de informação de um produto entre sistema de gestão do ciclo de vida de produtos.

O standard STEP limita-se, no entanto, a cobrir a fase de *design* de um produto, pelo que havia necessidade de encontrar uma forma de representar um produto ao longo de todo o seu ciclo de vida, da concepção ao seu "desmantelamento". Para culmatar esta falha, foi desenvolvido, em 1999, um Application Protocol para o standard STEP, extendendo a sua funcionalidade ao longo de todo o ciclo de vida do produto.

O EDMS/MTF é o sistema desenvolvido pelo CERN para gestão de documentação técnica e de equipamento. O sistema gere toda a informação relacionada com o Large Hadron Collider (LHC). Um projecto com tais dimensões e complexidade envolve centenas de empresas em próxima colaboração, na qual a troca de informação é constante. No CERN, o sistema EDMS/MTF serve como repositório central onde toda a informação relativa ao projecto é armazenada e gerida.

No entanto, para o projecto LHC, o CERN impôs os métodos e ferramentas a usar na colaboração com empresas e outros colaboradores. A importação de informação segue certas especificações e formatos de ficheiros, muitas vezes diferentes dos processos de negócio internos das empresas envolvidas.

Este documento apresenta os *standards* STEP e PLCS, dando uma perspectiva detalhada da sua arquitecturre, objectivos, dominio de aplicação e directivas de implementação. Este trabalho também comtempla a hipotese de implementação do standard no sistema EDMS/MTF, considerando possíveis vantagens, problemas e sugestões quanto à tecnonologia a utilizar na sua implementação.

Acknowledgments

I would like to thank the following people for helping me in my work:

To my supervisor at FEUP, Jaime E. Villate, for introducing CERN.

And particularly to my supervisor at CERN, Eva Sanchez-Corral Mena, for the patience, dedication and support during this stay at CERN.

Additional and very much appreciated help during the work and stay at CERN:

David Widegren, TS/ASE section leader

Teresa Batista Neto, TS/ASE/EPS

Stefan Milenkovic, TS/ASE/CSE

Marcin Sobieszek, TS/ASE/EPS

Aleksandra Wardzinska, TS/ASE/CSE

Table of Contents

Abstract 3
Resumo 4
Acknowledgments 5
1. Introduction 13
1.1. CERN 13
1.2. The LHC 13
1.3. The Detectors 15
1.4. Other activities at CERN 17
1.5. Applying PLCS to EDMS/MTF 18
1.6. Disposition of the report 19
2. EDMS at CERN 20
3. EDMS Fundamentals 24
3.1. Projects, Items and Documents 24
3.2. Project Management 25
3.3. Item Management 25
3.4. Document Management 26
3.5. Document Workflow and Release Procedures 26
4. MTF Fundamentals 27
5. PLM, PDM & ERP 28
5.1. PLM FUNDAMENTALS 28
5.2. PDM 28
5.2.1. Engineering Data 29
5.2.2. Bill of Materials 30
5.2.3. Product Structures 31
5.2.3.1. Product Breakdown Structure 31
5.2.3.2. Assembly Breakdown Structure 33

5.3. Enterprise Resource Planning, ERP 34

6. Extended Enterprise Concept 35

6.1. Information sharing in the extended enterprise 36

6.2. Information sharing through product's life 38

6.3. The Information Standards....................39

6.4. Common information source 39

6.5. Point-to-point information transfer 40

6.6. Information broker....................41

6.7. Information manager....................41

7. STEP 43

7.1. Introduction....................43

7.2. Scope and Domain of Application 43

7.3. Goal....................45

7.4. Application of the standard....................46

7.5. Current Status and future perspectives....................47

8. PLCS 48

8.1. Introduction....................48

8.2. Scope 49

8.3. Goal....................50

8.4. PLCS Concepts 50

8.5. PLCS Components 51

8.6. PLCS Implementation....................54

8.6.1. Principles....................54

8.6.2. DEX....................54

8.6.3. DEX Architecture....................55

8.6.4. Capability 57

8.6.5. Template 57

8.6.6. Reference Data 58
8.6.7. Business DEX, Templates and specific Reference Data 58
8.7. Example - Aviation Maintenance DEX 59
8.8. Business Model 61
8.9. Data Exchange 62
8.10. Implementation Methods & Tools 65
9. Business Benefits of PLCS 66
10. Share-a-Space, an example of Extended Enterprise concept 68
10.1. Principles and implementation approach 68
10.2. Architecture 69
11. Case Study 71
11.1. The role of CERN in the supply chain 72
11.1.1. Product and information management 73
11.1.2. Challenges 75
11.2. EDMS/MTF Architecture 76
11.3. EDMS/MTF Workflow 81
11.3.1. Design 81
11.3.2. Manufacturing, Installation & Commissioning 82
11.3.3. Operation & Maintenance 83
11.4. EDMS/MTF Information Flow 84
11.4.1. Planning, Specification & Requirement Management 85
11.4.2. Design 85
11.4.3. Manufacturing 85
11.4.4. Installation 86
11.4.5. Maintenance 86
11.5. EDMS/MTF, Collaborative workspace 88
11.6. Applying PLCS to EDMS/MTF 89

11.6.1. The Ideal..92
11.6.2. The Compromise Solution...93
11.7. Data Importing with PLCS 99
11.8. Product as Individual 101
11.8.1. EDMS/MTF Product Breakdown 101
11.8.2. EDMS/MTF Versioning 104
11.9. Product Breakdown in PLCS 107
11.9.1. Representation of a product as planned..108
11.9.2. Referencing a Part in a Breakdown..109
11.9.3. Representation of a Product as Realized 110
11.9.4. Relationship of a Product as Planned to a Version of the Design.......................111
12. Conclusions...112
References 115
Appendix I...117
Appendix II..119

Table of Figures

Figure 1: LHC layout 14

Figure 2: Accelerators at CERN 15

Figure 3: The interaction of various particles with the different layers of a detector 16

Figure 4: Equipment traceability through its lifecycle 20

Figure 6: D7i screen shot - object form 23

Figure 7: Relationship between projects, items and documents 24

Figure 8: Project management 25

Figure 9: Document management 26

Figure 10: Example of a release procedure 27

Figure 14: Product Breakdown Structure 32

Figure 12: Relationship between PBS and WBS 33

Figure 13: Assembly Breakdown Structure 33

Figure 14: Traditional supply chains 36

Figure 15: Extended Enterprise information 37

Figure 16: The Engineering Supply Chain through the product's life-cycle 38

Figure 17: The Point-to-Point information management 40

Figure 18: Consolidated information for Extended Enterprise 42

Figure 22: Central concepts in PLCS 50

Figure 23: The role of a STEP AP 51

Figure 24: Schematic representation of PLCS 53

Figure 25: DEXs, Capabilities, Templates, and Reference Data 55

Figure 26: DEX architecture extended with business specific constructs 56

Figure 27: Activity model for "Provide Support" 60

Figure 28: Activity model for "Collect data and provide feedback" 60

Figure 29: Typical business process using the Aviation Maintenance DEX 62

Figure 30: Example exchange points 63

Figure 31: Overview of PLCS constructs used to represent maintenance feedback information 64

Figure 32: An EXPRESS-G representation of the information model for "*assigning_activity*" ... 65

Figure 33: Share-a-Space architecture 70

Figure 34: CERN - the supplier of its suppliers 72

Figure 35: Example of the Supply Chain for the cryodipoles 74

Figure 36: EDMS/MTF main modules 76

Figure 37: The "best of breed" 77

Figure 38: Simple view of EDM/MTF architecture 78

Figure 39: EAB architecture 79

Figure 40: Information Flow 87

Figure 41: Data exchange process 90

Figure 42: EDMS/MTF with AP239 database compliance 92

Figure 43: PLCS plug-in 93

Figure 44: New EDMS/MTF generic architecture 94

Figure 45: Data access using SDAI 96

Figure 46: Deriving an implementation model 98

Figure 47: Proposed EDMS/MTF PLCS plug-in architecture 99

Figure 48: Importing data into EDMS/MTF using PLCS 100

Figure 49: Imported data conversion to PLCS 100

Figure 50: As-designed structure of LHC baseline 102

Figure 51: Assembly Breakdown Structure in EDMS/MTF 103

Figure 52: Data Organization in EDMS/MTF 104

Figure 53: Connection between Bill of Material and Manufactured Equipment 104

Figure 54: Versioning in designs 105

Figure 55: Connection between versions in Bill of Material and Manufactured equipment 105

Figure 56: New version in design 106

Figure 57: Connection between new versions in Bill of Material and Manufactured equipment 106

1. Introduction

1.1. CERN

The European Organization for Nuclear Research, commonly known as CERN[1] is the largest particle physics laboratory in the world. Situated just outside Geneva on the border between France and Switzerland, CERN is considered to be the Mecca of all physicists. Founded in 1954 under the name "Conseil Europeen pour la Recherche Nucleaire", from which the abbreviation was taken, it is one of the first European joint ventures. It was formed as a collaboration of 12 European countries and currently the membership has grown to 20 member states and 8 observer states or organizations.

Its main purpose is high-energy physics research which involves cutting edge technology creating particle accelerators and detectors, powerful magnets, vacuum systems and a computing infrastructure to support all its activities. With an annual budget of over 850 million francs (~500 million Euros) is one most important research centers in the world.

CERN employs about 3000 people although there are approximately 6500 scientists from all over the world collaborating on its projects, representing 500 universities and more than 80 countries. Actually half of the world's particle physicists do their research at CERN.

The intention behind the creation of CERN was to study the structure of matter, where it comes from, how it behaves, what forces keep it together and how do the different components of matter interact between themselves. This also the aim of a branch of physics called particle physics.

To answer these questions and others CERN has built some of the most amazing instruments know to Man. Particle accelerators and detectors, machines designed to explore a little further beyond what is known in physics.

1.2. The LHC

In 1995 the CERN council approved the construction of the LHC (Large Hadron Collider) with a budget of 2.5 billion Swiss Francs. It is new particle accelerator with 27km of diameter at a depth of 100 meters underground, becoming the largest and most advanced scientific instrument in the world. It is built in an existing tunnel of the LEP or Large Electron Positron Collider which was closed down in November 2000.

The collider tunnel contains two pipes enclosed within superconducting magnets cooled by liquid helium, each pipe containing a proton beam. The two beams travel in opposite directions around the ring. Additional magnets are used to direct the beams to four intersection points where interactions between them will take place.

The LHC will probe deeper into matter than ever before. Due to switch on in 2008, it will ultimately collide beams of protons at energy of 14 TeV, a unit of energy used in particle physics. Beams of lead nuclei will be also accelerated, smashing together with collision energy of 1150 TeV.

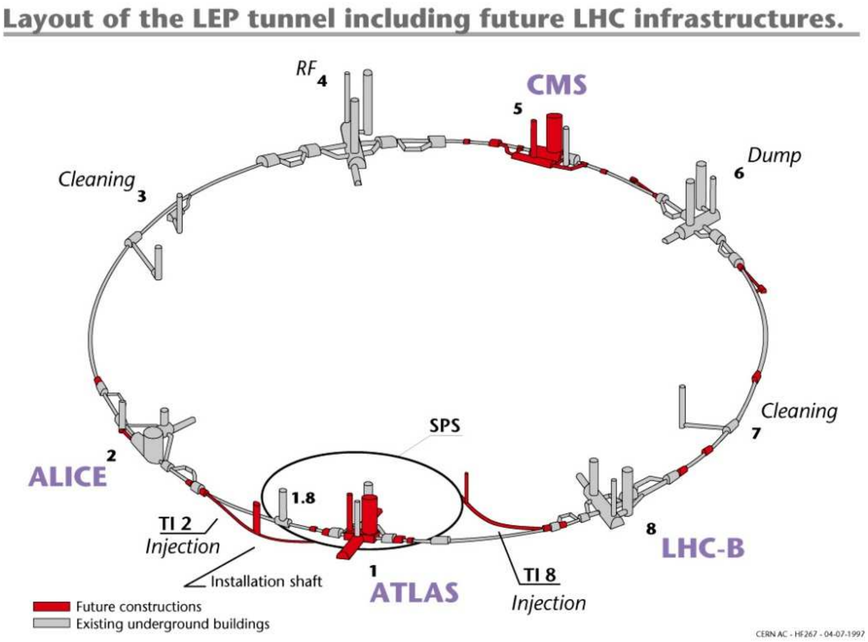


Figure 1: LHC layout

Prior to being injected into the main accelerator, the particles are prepared through a series of systems that successively increase the particle energy levels. The first system is the linear accelerator Linac2 generating 50 MeV protons which feeds the Proton Synchrotron Booster (PSB). Protons are then injected at 1.4 GeV into the Proton Synchrotron (PS) at 26 GeV. The Low-Energy Injector Ring (LEIR) will be used as an ion storage and cooler unit. The Antiproton Decelerator (AD) will produce a beam of anti-protons at 2 GeV, after cooling them down from 3.57 GeV. Finally the Super Proton Synchrotron (SPS) can be used to increase the energy of protons up to 450 GeV.

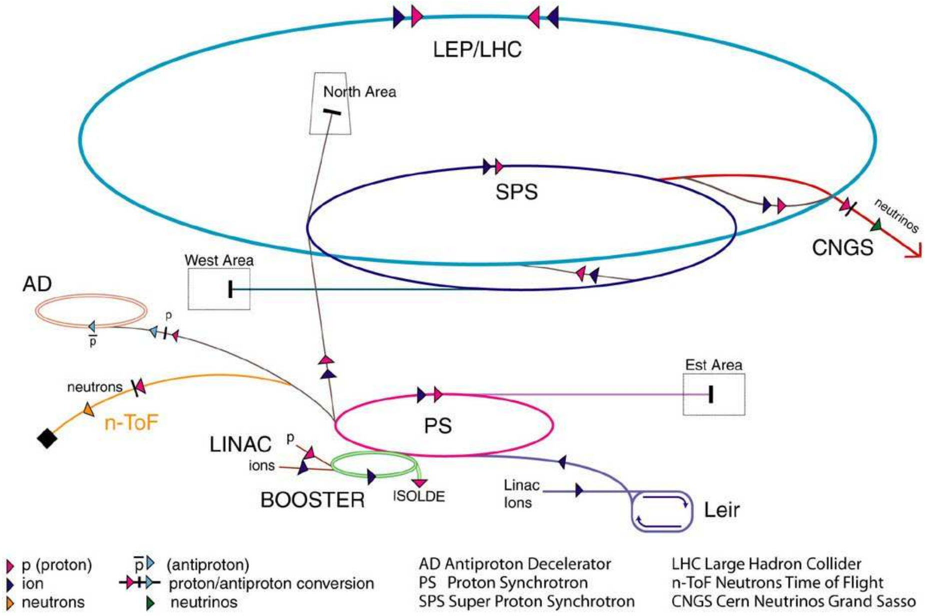


Figure 2: Accelerators at CERN

When switched on, it is hoped that the collider will produce the elusive Higgs boson particle — often dubbed the God Particle — the observation of which could confirm the predictions and 'missing links' in the Standard Model of physics, and explain how other elementary particles acquire properties such as mass. This will allow scientists to penetrate still further into the structure of matter and recreate the conditions prevailing in the early universe, just after the "Big Bang".

This accelerator will generate vast quantities of computer data, which CERN will stream to laboratories around the world for distributed processing (the GRID technology). In April 2005, a trial successfully streamed 600MB per second to seven different sites across the world. If all the data generated by the LHC is to be analyzed, then scientists should achieve 1,800MB per second during LHC operation. The start of the LHC is foreseen by Spring 2008.

1.3. The Detectors

The experimental apparatus and accelerator facility that physicists use to study particles are called detectors. Detectors can be thought of as enormous microscopes, so powerful that they can probe the tiny atomic nucleus, and make fundamental article activity visible to us. Six detectors are being constructed at the LHC. They are located underground, in large caverns excavated at the LHC's intersection points. Two of them, ATLAS and CMS are large, "general purpose" particle detectors. The other four (LHCb, ALICE, TOTEM, and LHCf) are smaller and more specialized. Each detector is prepared by a collaboration of more than 100 scientific institutes and 1500 collaborators throughout the world. A detector is an assembly of many different sub-detectors, which can weight between a few gram and thousands of tons. The whole detector contains for example magnets, mechanical structure and electronic components. Each detector will contain of the order of 10 million components and have a volume larger than 500m3. Figure 3 presents the interaction of particles in different layers of a detector. Neutrinos are not shown because they rarely interact with matter, and can only be detected by missing matter and energy.

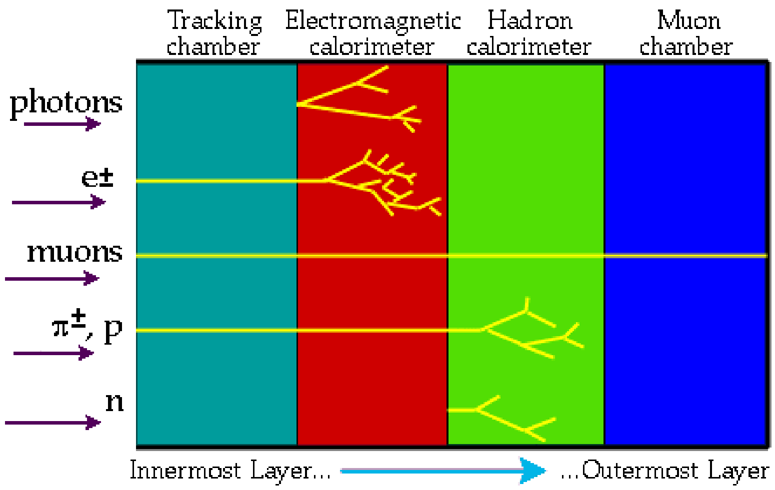


Figure 3: The interaction of various particles with the different layers of a detector

The LHC will contain four detectors with different purposes:

- ATLAS: A Toroidal LHC Apparatus (http://atlasinfo.cern.ch/Atlas/Welcome.html).
- ALICE: A Large Ion Collider Experiment (http://www1.cern.ch/ALICE/welcome.html).
- CMS: The Compact Muon Solenoid (http://http://cmsinfo.cern.ch/Welcome.html).
- LHC-b: Study of CP violation in B-meson decays (http://lhcb.cern.ch:8080/servlet/Nocchie).
- LHCf: LHC-forward
- TOTEM: Total Cross Section, Elastic Scattering and Diffraction Dissociation (http://totem.web.cern.ch/Totem/)

1.4. Other activities at CERN

But CERN is not only about particle physics. It is given the world advances in areas of cancer therapy, medical and industrial imaging, radiation processing, electronics, destruction of toxic products and maybe the most famous, the World Wide Web. The WWW began as a project called Enquire initiated by Tim Berners-Lee and Robert Cailliau in 1990. Together in collaboration with Fellows, Technical Students and Summer Students they developed a way of using hypertext to facilitate information sharing between researchers.

1.5. Applying PLCS to EDMS/MTF

The *mandate* of the *Technical Support Department* is to provide support for the technical infrastructure of CERN, accelerators, experiments and services related to the site operation and maintenance. The department has the responsibility to ensure monitoring and operation of the technical infrastructure of the whole site 365 days per year and 24h/24. It is in charge of the maintenance of sites and buildings, cleaning, gardening and access surveillance to sites and provide general services as transport, mail and housing.

The Access, Safety & Engineering tools (ASE) group is responsible for developing software for engineering and equipment data management, controls for technical infrastructure and in particular safety systems as access control, fire and gas protection and systems associated to the fire fighting brigade. Since the design and construction of the LHC, the ASE group has the mandate for managing the LHC data.

The Large Hadron Collider is composed of millions of high-tech components installed in a 27 km long circular tunnel, 100 meters below ground. Many of these very expensive components will have to be maintained and individually traced during 20-25 years in a partly irradiated environment.

LHC engineering and equipment data is today managed using a single Web portal accessing a combination of two powerful commercial data management systems. The system is designated EDMS/MTF, EDMS or Engineering Data Management System takes care of document management while MTF or Manufacturing and Test Folders does the asset tracking and management of all equipment related actions.

The construction of the LHC involved the participation of many external entities, from other research institutes to private companies. During the whole lifecycle of each piece of LHC equipment, data is captured, incoming from different sources and centralized in EDMS/MTF. Drawings, specifications, technical details, measurements, all is stored and managed by the system individually for each piece of equipment.

However, the input or import of data by external entities into EDMS/MTF is done using strict procedures and formats imposed by CERN. These entities had consequently to adapt their internal business processes and adopt this “new way of working”.

For some years now, an ISO standard called STEP is available that enables the representation and exchange of product data. Although, it has been mainly using for exchange of CAD design data, this ISO also provides a way of representing product data during its whole lifecycle. This component of STEP called Product Life Cycle Support provides an information model that can be used to manage complex equipment data from design to disposal.

The purpose of this report is to:

- Present EDMS/MTF as CERN's Product Life Cycle Management system, its functionality
- Introduce the ISO 10303 (STEP) standard, scope, goal and how it can be applied to a in real
- Introduce PLCS as an ISO 10303 component for data representation and exchange during a product lifecycle, scope, goals and main benefits
- Application of the PLCS standard to EDMS/MTF as a neutral file exchange format

1.6. Disposition of the report

This report is divided in eleven chapters to which a bibliography and appendixes have been attached. In this first chapter a brief introduction is made to CERN and to the context in which the *Applying PLCS to EDMS/MTF* project is in.

In the second, third, forth and fifth chapters, fundamental concepts are introduced for a more clear understanding of the EDMS/MTF system and its functionalities as a CERN's Product Lifecycle Management system.

The seventh, eighth, ninth and tenth chapters present core concepts of this thesis. These chapters provide some detail analysis of the *Extended Enterprise* concept along with STEP and PLCS standards.

In chapter eleven a case study is presented on how the PLCS standard could be implemented in the EDMS/MTF system. The first four parts focus on the particular case of how EDMS/MTF manages Cryodipole equipment and how CERN interacts with external entities during the manufacturing, installation and operation of Cyodipole equipment. A emphasis is made on the workflow and information flows during these stages. The rest of the chapter provides an hypothetical implementation of the PLCS standard to EDMS/MTF, considering two possible implementations. Additionally, as an exercise, some examples of the representation of some EDMS/MTF features in PLCS are also given.

2. EDMS at CERN

In 1995, a Web based system, designed to manage and archive the Organization's drawings, the CDD service was introduced for general use. Large volumes of drawings are produced by design offices at CERN or by subcontractors. CDD evolved and supplemented with a powerful commercial data management system is now capable of handling all types of documents required for a technical project. Today the system has become CERN's Engineering and Equipment Data Management system (EDMS[2]), managing almost 700.000 documents with their associated files. A document in EDMS may have one or more files associated to it.

In 1999 the need for the management of equipment manufacturing data became manifest in the LHC project, spawning a new tool, the Manufacturing Test Folder (MTF[3]). A major user requirement was that the MTF had to support a complete and detailed tracking of an equipment manufacturing and assembly process worldwide.

The scope of the MTF has subsequently been progressively extended to support the LHC installation and commissioning phases. When LHC shall enter in an operational phase, the MTF will be there to support the operation and maintenance phases in the life-cycle of the equipment.

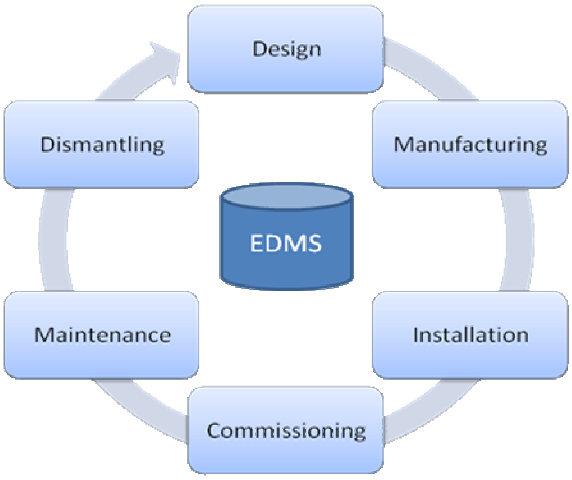


Figure 4: Equipment traceability through its lifecycle

Today, MTF helps the LHC engineers to manage more than 350.000 individually named equipment. Eventually, equipment data about the entire collider complex will be registered in MTF.

The EDMS and MTF facilities are based on commercial products. The core of EDMS is Axalant currently known as Agile PLM while MTF relies on Datastream's D7i Asset Tracking system (already used for the CERN equipment technical infrastructure maintenance follow-up) which is now known as Infor EAM. The EDMS team integrated these two products by developing a common system interface layer with a unique Web user interface. As both products use the Oracle database as repository, the development is definitely feasible.

The resulting implementation offers an easy-to-use Web interface providing the LHC project participants with a practical, easy-to-use tool to trace equipment evolution from design, manufacturing to installation and operation/maintenance. Having all equipment information and documentation easily available on a single computer system will also permit the vital knowledge transfer between the machine builders and those who will operate and maintain it later.

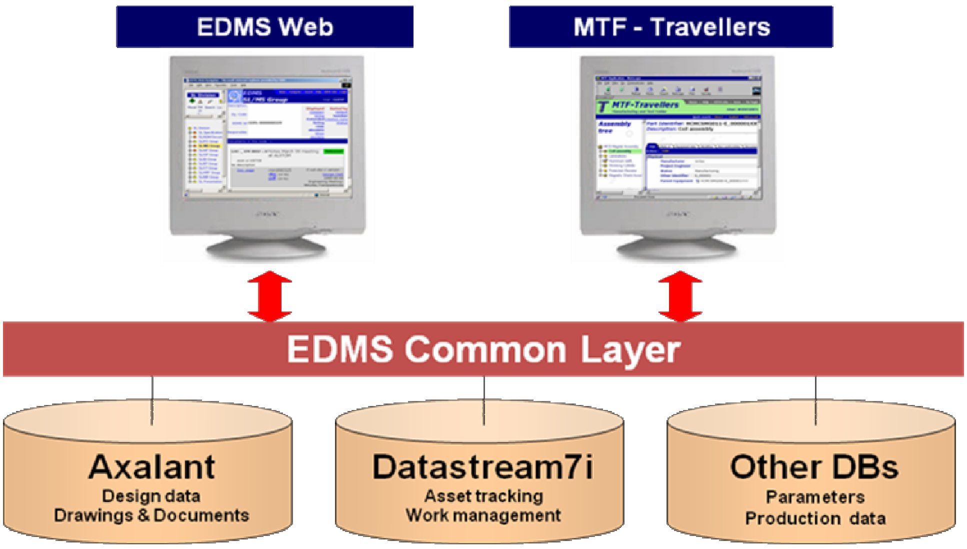


Figure 5: EDMS/MTF architecture [4]

Axalant[5] was introduced at CERN in 1997 as CADIM, its previous name. It was adopted after a study of user requirements, market survey, benchmarks and pilot projects carried out by the EDMS taskforce. At the time of the EDMS taskforce, Axalant was selected because of strong version and variant management, integrations with the CAD-systems AutoCAD and EUCLID and the level of maturity and engineering know-how that was built into the system. It was also seen as an advantage that Axalant was running on Oracle, which is widely used at CERN.

Axalant is an evolution of CADIM and at the same time a new product. The objective of Eigner+Partner, Axalant's supplier, is to be able to offer a smooth transition for existing CADIM customers while bringing out a modern PDM system to gain market share and new customers, notably in the USA. CADIM users will find Axalant familiar, but the user interface and underlying architecture have changed. Instead of the former Entity-Relationship data model within DataView, Axalant includes an object-oriented "federated repository" that allows communication between Axalant modules and other applications through Corba services. The DataView and SQL-binding to the relational database persists, though.

However, Axalant has evolved since it is now a PLM. Eigner+Partner was bought by Agile which currently is part of the Oracle group. Consequently, the name of the product also changed and it is now known as Agile PLM. Nevertheless, CERN still uses Axalant, an old version of its modern equivalent Agile e6. These products are used by companies such as B/E Aerospace, Heinz, Dell, among others.

For asset tracking EDMS/MTF uses D7i[6] which stands for Datastream7i. D7i is a comprehensive Oracle-based Asset Tracing and Maintenance Management System from Datastream (which was recently acquired by Infor). It is designed to manage physical assets and maintenance functions effectively. It is not only a maintenance tool, but also an assets management and tracking tool.

There are two modules in D7i that are particularly important: Asset management and Work management. In D7i, data is accessible through the use of forms. Each form shows records of individual information items.

There are mainly four types of D7i objects, *Assets, (Functional) Positions, Systems* and *Locations*.

- Objects may be organized in a hierarchy with multiple levels.
- *Asset* - Generally a physical object
- *Functional Position* - Function performed by a general kind of asset
- *System* - Logical collection of positions and/or assets that work together
- *Location* - Physical location of assets, positions and systems

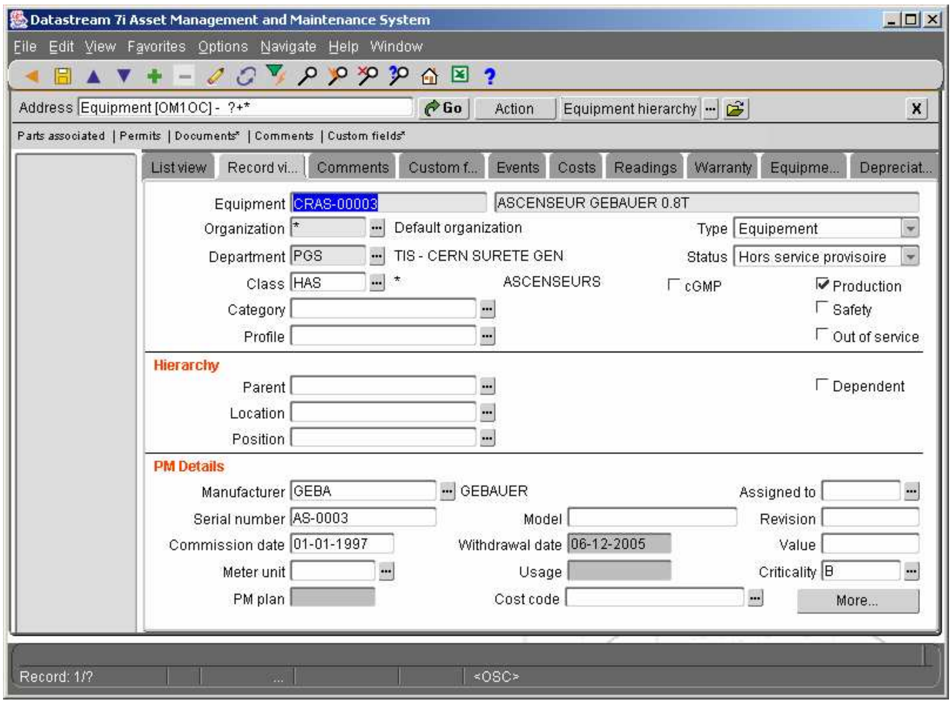


Figure 6: D7i screen shot - object form

Object main attributes are: serial number, description, object type, status, etc. Objects are divided in classes and categories. A class is a set of entities that share certain characteristics, e.g. all objects belonging to it have a set of common parameters. A category is a subclass. Default values for attributes can be defined for each category and objects in the same category will share those attributes values.

3. EDMS Fundamentals

3.1. Projects, Items and Documents

There are three main entities within EDMS [7]:

- *Items* – Represent physical objects. An item consisting of other items form an ABS.
- *Projects* – Hold information concerning a certain project. Can consist of several items, projects and/or documents. A project consisting of other projects form a PBS, *Product Breakdown Structure*.
- *Documents* – Work as a container for actual data, e.g. text files, 3D models etc. Carry metadata.

As shown in the Figure 7, EDMS allows the definition of relationships between different entities. Using these relationships, complex and flexible structures can by represented. Unlike hierarchies defined by a set of codes, structures based on relationships are not limited to be tree-like, and can easily be modified without having to worry about 'breaking' codes which are already in use.

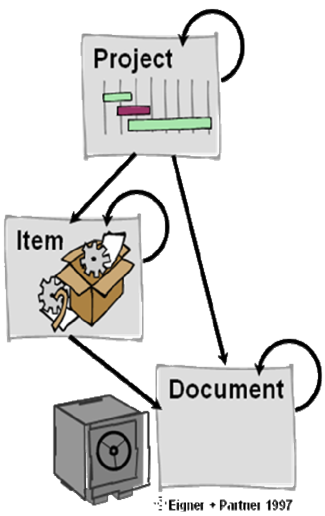


Figure 7: Relationship between projects, items and documents

3.2. Project Management

In EDMS, projects can be seen as “folders”, which may contain items, documents and even sub-projects. By using the project entity with a breakdown structure, EDMS allows the creation of a breakdown of a product where each node represents a sub-project or simply a container for a set of documents or activities related to the product. The aim is to define a project entity that leads to the definition and creation of a product.

A project may set milestones and cost planning allowing users to do project monitoring and to calculate costs of projects. “By linking projects bids/orders, and actual product descriptions, the user is able to monitor the project by controlling time, cost, and resource necessary for the project.” [7]

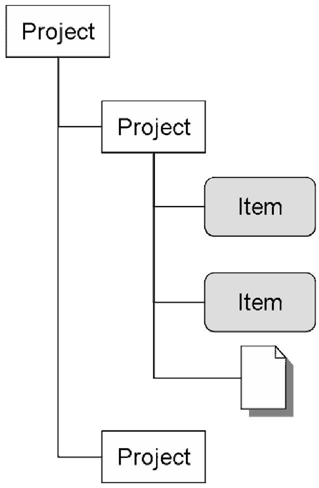


Figure 8: Project management

3.3. Item Management

The detailed designs and specifications of LHCs assemblies and components are being represented in the EDMS as Items. Through the use of a product breakdown structure to represent Item structure, a Bill of Materials (BOM) is obtained. This dynamic item structure evolves during the course of the design phase in a project providing important data to engineers for the manufacturing phase.

3.4. Document Management

The Document entity in EDMS does not represent a document file but a container either for document files or other sub-Document structures. Document structures are useful for representing the structure of compound documents (e.g. an assembly manual as an ordered collection of assembly and quality control procedures). EDMS's document managed allows the creation, update, control and distribution of a document over its lifecycle.

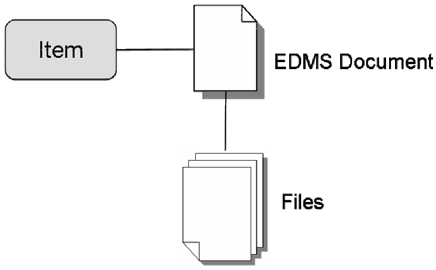


Figure 9: Document management

3.5. Document Workflow and Release Procedures

An important feature of EDMS is the capability of managing approval procedures. These approval processes are particularly useful to avoid situations in which, for example, a drawing is sent to production without previous approval.

During its workflow, a document can pass through different statuses. In EDMS, *In Work*, *Under approval*, *Released*, *Obsolete* are some of the possible status. In this document status manipulation, user rights play an important role since a user must be authorized to change the status of a particular document.

Another important facet is that different types of documents may have different release procedures and, consequently, transitions between statuses behave differently.

Figure 10 shows an example of a release procedure highlighting possible transitions between document statuses.

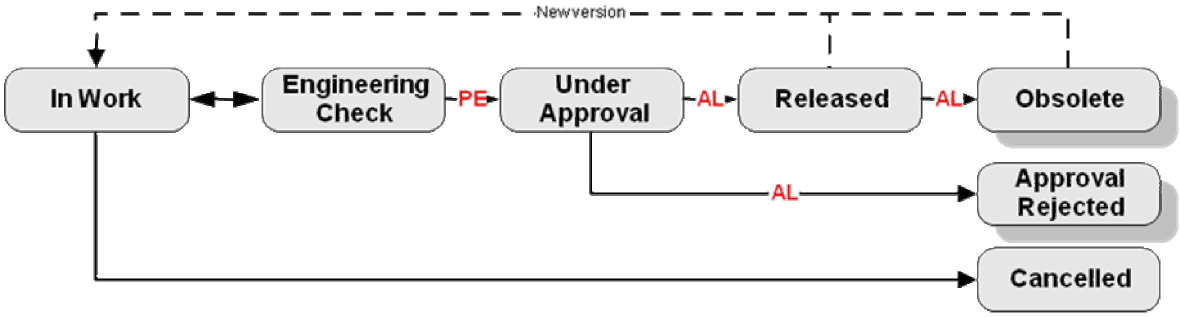


Figure 10: Example of a release procedure [4]

In addition to different release procedures, EDMS also provides a document versioning feature. The version of documents can be created manually by the user or automatically by a trigger set up to for new version creation for a specific document status change. These triggers can also be used for user notification by email.

Procedures for Engineering Change Requests (ECRs), can also be set up to control the design and how it is being modified, as being used in the LHC.

4. MTF Fundamentals

In addition to equipment traceability, MTF can also provide work management functionalities.

The system allows the management of works orders from creation, to planning and scheduling tracking all aspects related to work performed on assets. With this tool, users can keep track of preventive maintenance inspections and jobs, including step-by-step instructions or checklists, lists of materials required, and other pertinent details.

A more recent feature is the safety inspections management. Whenever a safety inspection is done to a building or equipment, it can also be created in MTF. In this way, the inspection job is associated with the corresponding asset. Users can consult all inspections performed along with report documents produced with the test results.

Some short facts about the EDMS/MTF:

- >500.000 documents & drawings.
- >10.000 new documents/month.
- >300.000 registered LHC components.
- >15.000 new equipment are registered/month.
- ~2000 active users every month.
- >60.000 file downloads/month.

5. PLM, PDM & ERP

5.1. PLM FUNDAMENTALS

Product Lifecycle Management is the process of managing the lifecycle of a product from its conception and design until its disposal. The PLM concept it is not simply a piece of technology. It is a business approach to solving the problem of managing information concerning a product through its lifecycle. This approach consists of a set of business solutions that support the collaborative creation, management, dissemination, and use of product definition information.

This approach, however, can and usually has practical implementations that recur to technology.

A PLM typically consists of four main areas:

- Product and Portfolio Management (PPM)
- Product Design (CAx)
- Manufacturing Planning (MPM)
- Product Data Management (PDM)

The Product Data Management is the most relevant and will be explained in detail in the next section.

5.2. PDM

The PDM abbreviation stands for Product Data Management and it describes computerized systems that provide means to improve the management of engineering data, of engineering activities, of engineering changes and of product configurations.

A PDM focuses primarily on managing and tracking the creation, change and archive of all information related to a product. The information being stored and managed will include engineering data such as CAD design models, drawings and their associated documents. These documents are typically requirements, specifications, manufacturing plans, assembly plans, test plans and test procedures.

With these systems, companies are able to get control of engineering information, and to manage activities in several departments. In the long term, systems will allow companies to get control of all their engineering information, and manage the overall engineering process. These characteristics set them apart from systems such as CAD that aim to improve the productivity of individual tasks in one functional area. In fact, PDM stems from traditional engineering design activities that created product drawings and schematics on paper and using CAD tools to create parts lists (Bills of Material structures - BOM). The PDM and BOM data is used in enterprise resource planning (ERP) systems to plan and coordinate all transactional operations of a company (sales order management, purchasing, cost accounting, logistics, etc.)

When the first CAD systems appeared in the market, they revolutionized the way CAD designs were created. However, there were still problems in managing CAD files, especially in large scale project. To solve this and other data management problems, the PDM system was introduced around the middle of the 1980's.

PDM systems have been evolving ever since and now provide:

- Document management
- Integration of manufacturing systems, share of production data
- Product Structure and Data management
- Process and Workflow management
- Collaboration

Nowadays, large engineering constructions like a modern aero plane, consists of millions of parts and details, it would be almost impossible to track, modify, update, manage, coordinate etc construction drawings and documents without computerized PDM systems.

5.2.1. Engineering Data

"The term 'Engineering Data' includes all data related both to a product and to the processes that are used to specify, develop, produce and support it." [8]. The typical examples of engineering data are specifications, schedules, process plans, technical manuals, project plans, Bills of Materials, test and quality results, calculations and measurements, computer programs, photographs and drawings.

Engineering Data can become very complex and voluminous. For these and other reason PDM systems can be very useful in managing this kind of information.

First of all, a PDM provides a central service for managing *electronic data*. Before the first PDMs or electronic data managers data was handled manually. In large scale projects, such as the LHC, the advantage of using a single data managing system instead of having to manage thousands of drawings, specifications, measurements, in paper format is pretty obvious.

But even using simple electronic format, the organization has little or no control over engineering data and engineering process. In an organization without a data managing system, many problems may arise.

It may be difficult to find the information needed and, if found, may not be the most recent one. Large scale companies and projects produce a large volume of data. Finding a particular document in terabytes of unstructured information may be similar to finding a needle in a haystack. Moreover, the scope of engineering data is quite wide and so many file and data representation formats may be used.

The problem gets worse when data entry is poorly controlled. Data can be lost, created with errors or duplicated.

Engineering data is used by many people, inside and outside a given organization. The same data may be handled by different people, in different locations and different purposes. Data has then to be made available and protected against unauthorized access. However, in an environment where several users and organizations are involved, several concurrent copies may be maintained. There is no master copy and so when one is changed these changes do not propagate. This confusion with different data versions may lead to loss of data and consequently a end result that was not expected.

The implementation of a PDM system as a single source of data in an organization, avoids situations in which different groups working in the same project overlap activities which inevitably will generate problems.

5.2.2. Bill of Materials

Bill of Materials is the term used to describe the "parts list" of components that are attached together to form a product. When a company produces a product, it must keep track of the materials and components used in its creation. In a way, a bill of materials is similar to a recipe. Every ingredient used is listed in the bill of materials. Strict record keeping is the key rule to creating a bill of materials, because no item can be skipped. This is largely because the bill of materials can be used later to narrow down issues for repairs of a product if necessary. In addition, a bill of materials is essential when ordering replacement parts or when new versions of those parts are to be created.

A BOM is a hierarchical structure by nature where the top level represents the sub-assembly or end-item.

A bill of material can define products as they are designed (“as designed” - Engineering Bill of Material), as they are ordered (Sales BOM), as they are built (“as-built”- Manufacturing bill of material), or as they are maintained (“as-maintained” - Service BOM).

A BOM provides the following details of a product setup:

- Identity
- Designation
- Amount, numbers of
- Material

5.2.3. Product Structures

An essential part of a PDM system is product structure management. A breakdown structure provides means for project organisation and definitions of relations between systems. By organizing information in a structured way, a company can easily allocate resources, organize work, describe relationships between different entities, while providing a structures way of managing information.

There are four types of breakdown structures:

- Product Breakdown Structure (PBS): provides a hierarchical tree structure of components that constitute a product
- Assembly Breakdown Structure (ABS): provides a hierarchical tree structure of how parts were assembled to build up a product.
- Work Breakdown Structure (WBS): provides a hierarchical tree of the work defined by the project scope identifying all deliverables
- Organizational Breakdown Structure (OBS) – identifies and provides organizational relationships within an organization.

5.2.3.1. Product Breakdown Structure

As mentioned earlier, the product breakdown structure displays components of a product using a tree structure. In industrial applications such as EDMS, this kind of structure is particularly useful to manage design of parts to be manufactured.

Each node in the tree represents a part of the final product. To each part one can associate documents such as drawings, specifications, measurements, etc. As a result, the tree will be rich with information, not only providing the structure and components of a product, but also engineering data useful for its realization.

The structure itself is built by creating logical links between different parts, documents and additional data, building up an assembly of a product. These links are very useful if one part is modified for some reason. The impact of this change will reflect itself in the whole tree.

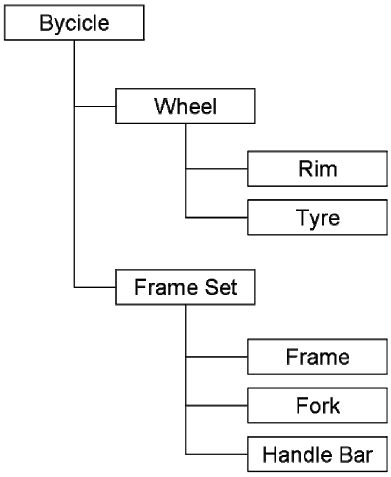


Figure 11: Product Breakdown Structure

A PBS tree provides information on the product's configuration while establishing a link between the different components and their technical description. This may include drawings, specification and other technical documentation. Additionally, the tree structure allows the management of components individually while belonging to a system.

The PBS structure can be considered as a subset of a Work Breakdown Structure. In generic terms, the PBS is meant to save information that belongs to projects, sub-projects, documents and items etc. On the other hand, the purpose of the WBS is to hold information about organisation and resource requirements of the tasks (activities) that is to be performed during production. The PBS and WBS used together will then enable the planning of the activities to be performed to produce the final product, providing engineers information they'll use in the production line.

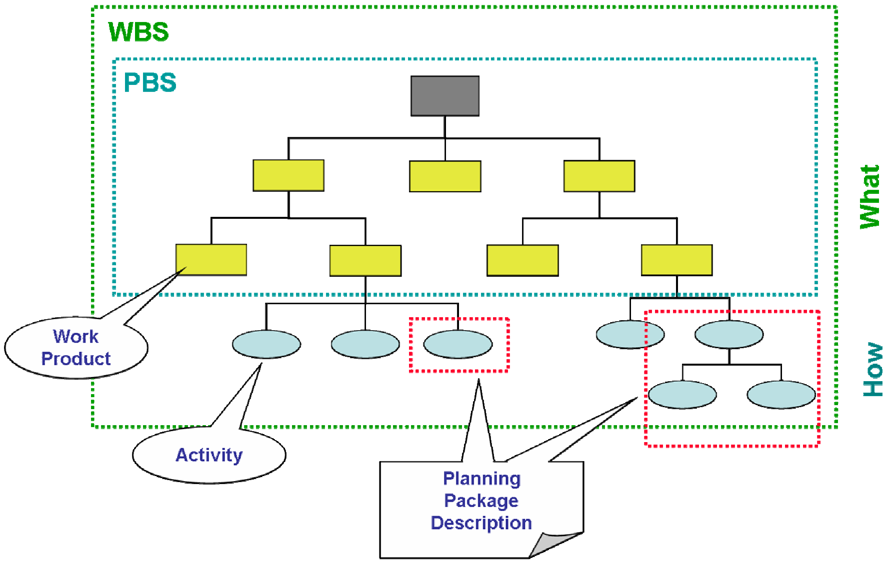


Figure 12: Relationship between PBS and WBS [10]

5.2.3.2. Assembly Breakdown Structure

The ABS structure complements the PBS one by providing engineering and equipment data for the manufacturing and assembly phases. The as-built structures provide information on how parts fit together, complement the information on parts and activities contained in a PBS structure and information on activities, infrastructure and logistics which are useful for managing work.

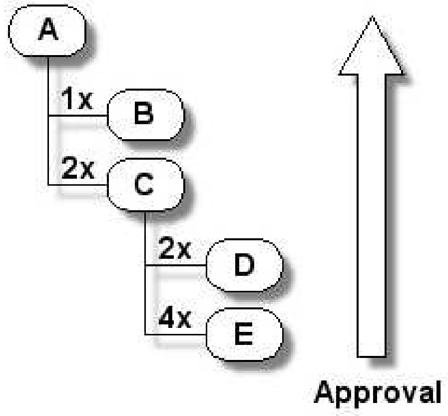


Figure 13: Assembly Breakdown Structure [11]

Each node in the ABS represents an assembly, a subassembly or a component. The links between these nodes provide a description of how the different components can be put together and how many of each components that are needed.

5.3. Enterprise Resource Planning, ERP

Enterprise Resource Planning (ERP) represents a wide set of activities supported by application software that helps an organization to manage the important parts of its business, such as product planning, purchasing and managing inventories, financial data, vendor relations, customer service and sales management.

ERP systems are used to optimize internal business processes and they aim at integrating several data sources and processes of an organization into a unified system. A typical ERP system will use multiple components of computer software and hardware to achieve the integration. A key ingredient of most ERP systems is the use of a unified database to store data for the various system modules. The benefits of consolidating multiple functions within one application include standardization, lower maintenance costs and easier, more-in-depth reporting capabilities.

All basic functions of an enterprise are, nowadays, covered by an ERP system, regardless of the organization's business. A software can be considered an ERP when it provides the same functionality as at least two systems. For example, a software package that provides both payroll and accounting functions could technically be considered an ERP software package.

Examples of modules in an ERP which formerly would have been stand-alone applications include: Manufacturing, Supply Chain, Financials, Customer Relationship Management (CRM), Human Resources, Warehouse Management and Decision Support System.

6. Extended Enterprise Concept

"If you think you can do it alone in today's global economy, you are highly mistaken."
– Jack Welch

The industrial development that occurred during the 1980's and 1990's has changed market conditions and requirements. Companies are no longer limited by their traditional boundaries and start partnerships or collaborations with other companies. For this reason, there is a need for sharing and integration of information, coming from partner companies.

Traditional information sharing was typically done through one-to-one exchanges of data between a sender and receiver. These information exchanges are usually implemented via dozens of open and proprietary protocols, message and file formats.

Nowadays, however, with the need for sharing and integration of information, this point-to-point exchange is not admissible anymore.

Along with the first partnerships or alliances of companies, came the term "extended enterprise". This concept is meant to represent a company that encompasses not only its employees, its board members, and executives, but also its business partners, its suppliers, and even its customers.

The extended enterprise can only be successful if all of the component groups and individuals have the information they need in order to do business effectively.

Extended enterprise applications are applications which span company boundaries. They include a web of relationships between the company and its employees, managers, partners, customers, suppliers, and markets. Supplier's stock needs to be known in real time. Customers and partners need to know about your latest products or services as soon as they're available. Company administration need to have the latest financial information to make decisions. The extended enterprise is an intricate, interconnected network of information, and you need true enterprise-strength solutions to tie all these together.

To implement this web of relationships technology plays a very important role. The extended enterprise is a complicated and highly connected network of information, and true enterprise-strength solutions are needed to tie all these together.

Often a major barrier is the difficult task of integrating PLM, ERP, and Product Support systems across partners in an ever changing market. The cost of a technical solution for these problems are often high and commonly lead to the adoption of a single solution making collaboration with other businesses/systems very limited.

The following sections try to describe the main ideas behind the "Extended Enterprise" concept as explained by Dr Mathis Johansson in [12].

6.1. Information sharing in the extended enterprise

The "Extended Enterprise" is a loosely coupled, self-organizing network of firms that combine their economic output to provide product and service offerings to the market [13]. Firms in the extended enterprise may operate independently or cooperatively.

The Extended Enterprise can alternatively be referred as a supply chain. A supply chain is the system of organizations, people, technology, activities, information and resources involved in moving a product or service from supplier to customer [14]. Supply chains include every company that comes into contact with a particular product. For example, the supply chain for most products will encompass all the companies manufacturing parts for the product, assembling it, delivering it and selling it.

In a traditional supply chain, however, the different companies are not interested in the remaining intervenients in the supply chain. They simply seek to maximize their revenue in their "area of interest" but may have little or no knowledge or interest on what is going on around them. In this typical "engineering supply chain", only information is managed and information flows between different companies are not established.

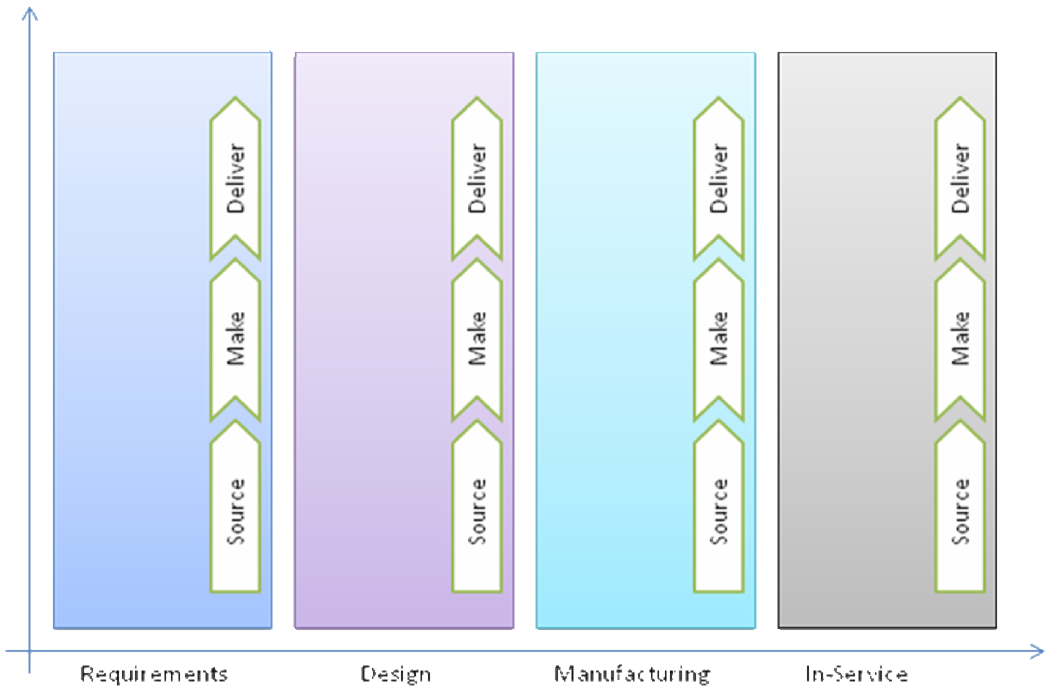


Figure 14: Traditional supply chains [12]

On the other hand, extended enterprises are based on the assumption that information flows freely among companies involved in a partnership. They no longer act completely independent, merely limited to their core competency, but they exchange information between each other. This, however, is not that obvious and simple to implement. Information format incompatibilities and mostly lack of established information flows between partners do not allow and immediate exchange of information.

An information flow specifies information shared between all partners, and which information has to be restricted or even confidential, during the life-cycle of the partnership.

The extended enterprise can only be successful if all of the component groups and individuals have the information they need in order to do business effectively. In order to achieve this, information must run freely among the business partners. However, this does not mean that information will be indiscriminately exchange between companies. Business rules, information flows and a common information set must be defined beforehand. These definitions are particularly important in an extended enterprise where partners, information ownership and software systems may change over time and affect the information consistency and validity.

It is important then to establish how the extended enterprise is organized and structured and its policies and mechanisms for the exchange of information. One essential step is to define a valid information set and rules for data representation must be agreed between all involved partners. This information set should be kept external from partners' own internal business processes and should be located within a common data repository.

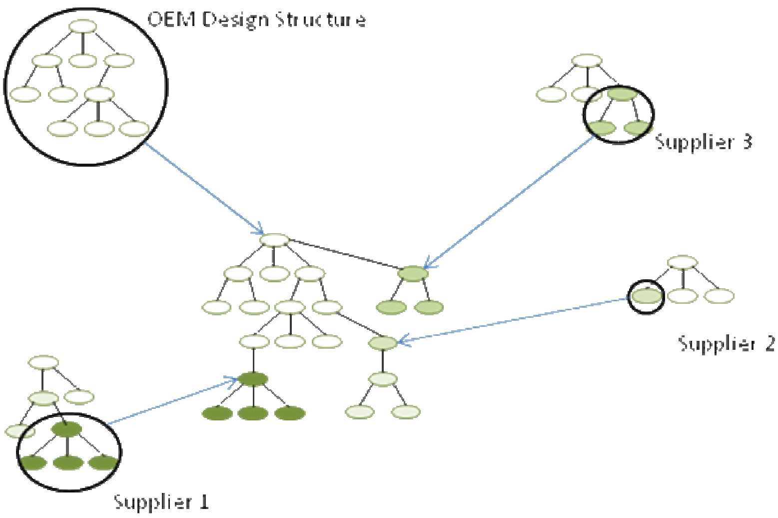


Figure 15: Extended Enterprise information [12]

This enables involved entities to separate their own internal process from the shared one in the partnership. Thus, companies maintain their internal model and are able to use the same information in the various collaborations.

6.2. Information sharing through product's life

In contrast with the traditional supply chain, which link value chains vertically, a supply chain through a product life-cycle develops horizontally. Previously organizations have been thought of as linear entities, each with a linear value chain that consisted of all the activities required to design, market, sell, produce, deliver and support products and services.

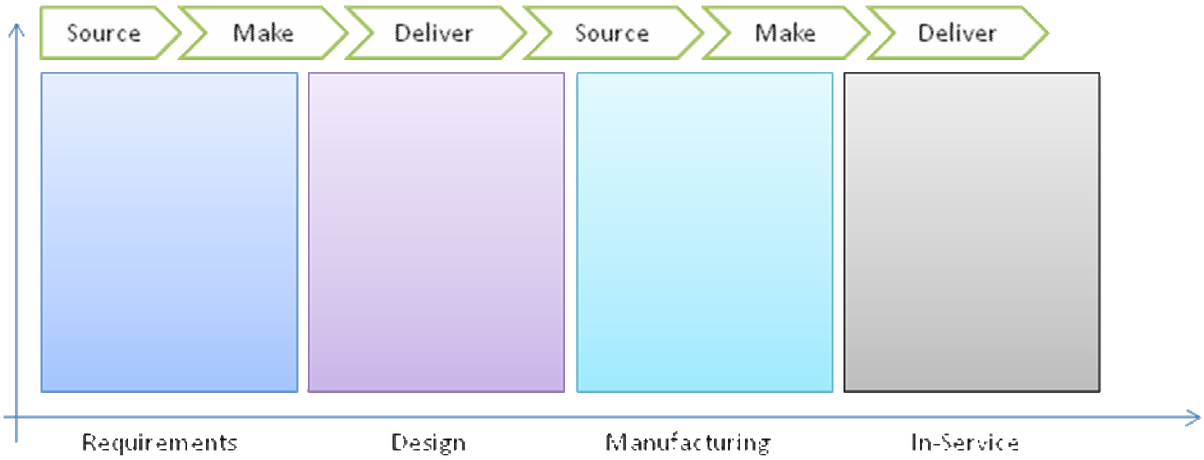


Figure 16: The Engineering Supply Chain through the product's life-cycle [12]

Suppliers and customers were thought to be outsiders of the organization's domain. These recent approaches of collaboration with partner companies brought the need to exchange and share information. This need, however, is not limited to simple data exchange since a common and shared business process is needed as well as and software system that can manage all information being "traded" between partners.

A virtual enterprise may change over the product's life cycle without participants knowing *a priori*. Companies may join or leave the partnership, the business context surrounding the collaboration may change, and even the aim of the project may suffer modifications. This requires information to be managed neutrally but with a strong emphasis on ownership and access control. Additionally, a partnership is composed of heterogeneous companies with their own internal business processes and their own scope within the project. Information has then to be managed with a neutral way, through the life cycle, decoupled of any particular business process of a participant.

6.3. The Information Standards

At the moment, companies generally use commercial products to manage their internal business processes. Typically, PLMs, ERPs, CRMs use their own proprietary formats and standards which makes communication between different products complicated if they do not belong to the same development company. In a good case scenario, these proprietary standards may become *de facto* standards which, by their importance in the market may be informally accepted.

In environments where collaboration between companies is done through the exchange and share of data along the life-cycle of a product, the use of standards is essential. In a worst case scenario, all partners can use different systems which use different standards and formats, make exchange impossible.

In the extended enterprise scenario, the adoption of a neutral exchange standard is essential. The ISO10303-239 or PLCS was created exactly to fill the need for a neutral information transfer mechanism along the product life-cycle. The standard covers the management of product data using defined data representation format. Many of the solutions and information management scenarios will most probably be based on this standard in the near future since it has been largely adopted in the last few years.

The PLCS standard or ISO10303-239 will be presented in chapter 8.

In the next points are presented different types of data sharing and data exchange.

6.4. Common information source

One of the main presuppositions of “Extended Enterprise” consists in making information flow between all partners in the collaboration. Information should then be available to all companies thought-out the product’s lifecycle.

The key concept behind an extended enterprise is the sharing of assured product and support information across the enterprise, making it available to partners over time and across organizational borders. In this context, exchange is not nearly as important as sharing since point-to-point information exchanges should not take place in this kind of scenario.

To realize the “Extended Enterprise” concept in full, it is necessary to consolidate the shared information set. It is vital that the information from a number of different sources is consolidated to provide the virtual enterprise with a “complete picture” valid for all partners. What is meant by “complete picture” is the integration of all information coming from all sources forming a consolidated single information set. Thus, the “complete picture” means that all partners will have the same view the whole data available. This, however, that all information will be available to all partners without any regard for ownership or access rights.

6.5. Point-to-point information transfer

The traditional way of exchanging information in a collaboration scenario is the point-to-point transfer. In this case, the communication is restricted to two endpoints. A direct link is established between sender and receiver and through this channel, information is transferred either using file transfer or direct data access interfaces.

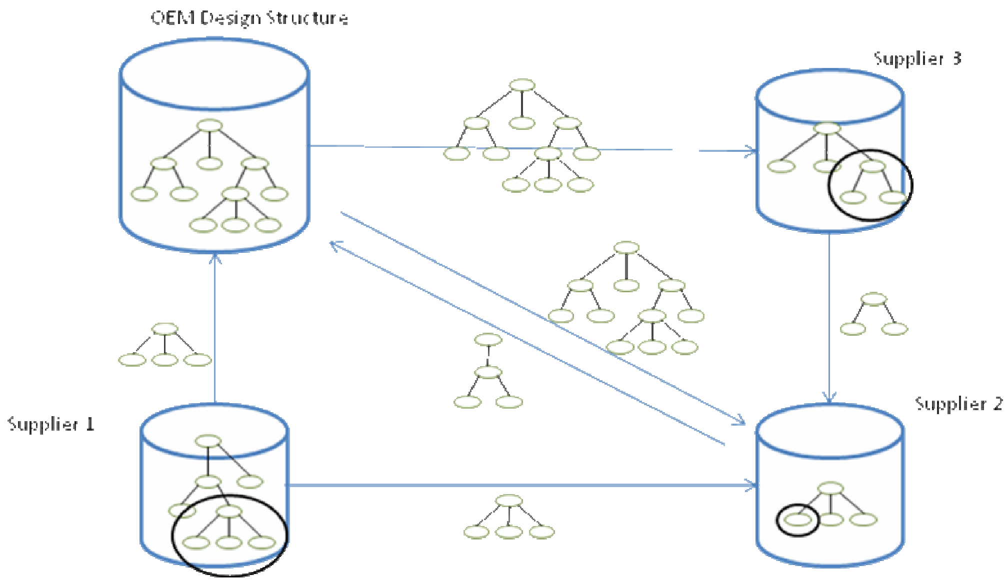


Figure 17: The Point-to-Point information management [12]

In a extended enterprise context, this is not a viable solution. In a collaboration with various partners, if data is transfered between only two entities, the remaining will not have access to it. None of the companies will have the “complete picture” and additionally, none of them will know where the most recent and valid piece of information resides. It is then impossible with this approach to achieve a single information set.

6.6. Information broker

An information broker may be introduced to overcome the limitations of point-to-point transfers. The broker acts as an intermediate between the two points, channelling the information to a predefined recipient. Yet, the remaining partners still would still be “blind” and wouldn’t have access to the information being exchanged. The complication concerning the location of the latest piece of valid information would still be present.

6.7. Information manager

The solution to the problem of lack of “complete picture” and information inconsistency and validity may be the implementation of an information manager. This entity takes the role of a single central information repository which can be easily accessed by all partners in the collaboration. Information created internally in the companies, typically through the use of PLMs, ERPs, etc, needs then to be consolidated in this single source of truth.

This solution, however, raises some issues regarding ownership and access rights to information. The “Extended Enterprise” advocates free flow of information between companies, but this full access is only theoretical. Although they are involved in a collaboration, companies are still companies and they are fighting for survival in a very aggressive market. The way a company manages information is vital to the success and so information to be shared is carefully selected as else as the partners information will be shared with. As the lifecycle of the product evolves, partners and information ownership may change. Consequently, companies may be reluctant in providing access to information and that later be used “against them”.

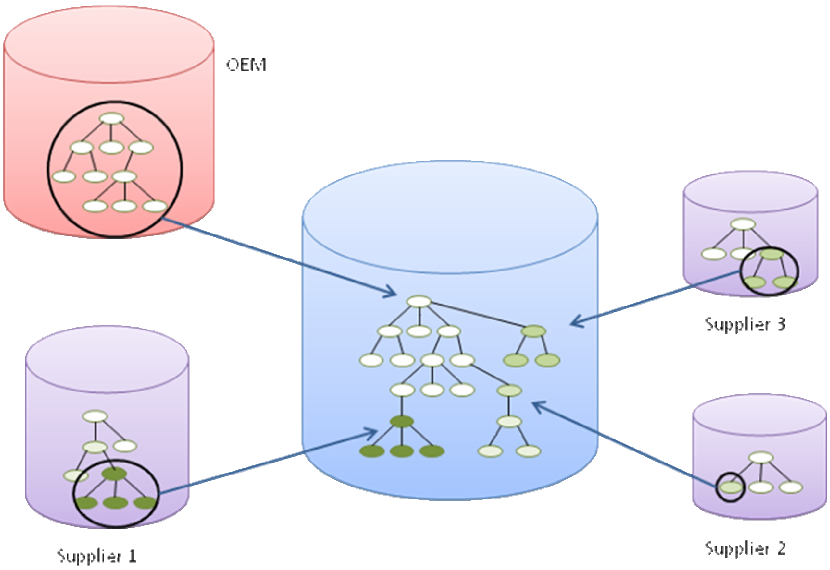


Figure 18: Consolidated information for Extended Enterprise [12]

The data model to be used in such information manager is a very important component in the whole implementation. To consolidate and integrate information coming from all the different sources, the data model much be rich and take into account all internal systems' data models of each partner. The aim is to obtain a valid set of information and a poorly built data model may lead to data deterioration, which does not serve the "Extended Enterprise" purpose.

The obvious main advantage of this solution is that it provides a neutral common set of information that manages data within a partnership. This information is managed independently of the internal systems of each partner.

The most important requirements for implementing an information manager are [12]:

- its ability to manage a rich information without interfering with the basic information constructs
- its ability to consolidate information from multiple sources and with different contexts
- its ability to manage and consolidate information from different parts of the product life-cycle
- its ability to manage ownership and access control

The information manager must also be neutral and complement the partners existing software systems so that new partners can easily be adopted.

7. STEP

7.1. Introduction

Since 1984 the International Organization for Standardization (ISO) has been working on the development of a comprehensive standard for the electronic exchange of product data between computer-based product life-cycle management systems. Initially, the development was concentrated on design and manufacturing applications. The final result was the standard, ISO 10303 [15], informally known as STEP (STandard for the Exchange of Product model data). The scope of this standard is much broader than that of other CAD data exchange formats, notably the Initial Graphics Exchange Specification (IGES. While IGES was developement specifically for the exchange of pure geometric data between computer aided design (CAD) systems, STEP is intended to handle a much wider range of product-related data covering the entire life-cycle of a product.

The development of STEP has been one of the largest efforts ever undertaken by ISO. Several hundred people from many different countries have been involved in the work for the past sixteen years, and development is as active now as it ever has been in the past. STEP is increasingly recognized by the industry as an effective mean of exchanging product-related data between different CAD systems or between CAD and downstream application systems. The first parts of the standard were published in 1994 after some ten years of work by members of the relevant standards subcommittee, ISO TC184/SC4, whose responsibility is `Industrial Data'. Much of this effort went to developing an infrastructure for the standard that would ensure its extensibility in the future.

ISO 10303 covers a wide variety of different product types (electronic, electro-mechanical, mechanical, sheet metal, fiber composites, ships, architectural, process plant, furniture,...) and life-cycle stages (design, analysis, planning, manufacture,...). This range is continually expanding as new parts of the standard are issued. These parts are referred to as ISO 10303-xxx, where xxx is the part number, and each is a standard in its own right, despite being a component of a larger whole and interdependent on other parts. Currently the overall standard is composed of about 40 parts, though many more are in development.

7.2. Scope and Domain of Application

ISO 10303 is an ISO standard for the computer-interpretable representation and exchange of industrial product data. Its official title is "*Industrial automation systems and integration - Product data representation and exchange*", and it is also known as, STEP as previously mentioned.

The objective of ISO 10303 is to provide a means of describing product data throughout the life cycle of a product that is independent from any particular computer system. The nature of this description makes it suitable not only for neutral file exchange, but also as a basis for implementing product databases and for archiving data. In practice, the standard is implemented within computer software associated with particular engineering applications and so its use and function will be transparent to a user. The descriptions are information models that capture the semantics of an industrial requirement and provide standardized structures within which data values can be understood by a computer implementation.

Typically STEP can be used to exchange data between CAD, Computer-aided manufacturing, Computer-aided engineering, Product Data Management/EDM and other CAx systems. STEP addresses product data from mechanical and electrical design, Geometric dimensioning and tolerance, analysis and manufacturing, with additional information specific to various industries such as automotive, aerospace, building construction, ship, oil and gas, process plants and others.

The implementable parts of ISO 10303 are known as *Application Protocols* (APs). Each AP is applicable to one or more lifecycle stages of a particular product class. However, the APs themselves built using *Integrated Resources* (IRs) which are basci constructs. The APs are numbered in a complex way. The 01x series of parts concern definitional resources and the 02x series implementation methods. Standards in the 03x series deal with validation of translators for conformance with the standard. The 04x series contains *Integrated Generic Resources*, which provide the basic building blocks of the standard. However, there is also a higher-level set of Integrated *Application Resources* in the 1xx series. The *Application Protocols* appear in the 2xx series, though some of their basic reusable components have now been separately standardized in the 5xx series - these are known as *Application Interpreted Constructs* (AICs). The AIC philosophy is currently being pursued further, and the intention is that in future application protocols will be assembled from sets of reusable standard *Application Modules*.

ISO 10303 is a collection of inter-related documents which form a multi-part standard. The parts are grouped into the following sections [16]:

- The *description methods* (Parts 1-19) provide the specifications of the languages that are used for creating the standards.
- *Implementation methods* (Parts 20-29) support the development of software implementations of the standards.
- *Conformance testing methodology* (Parts 30 -39) and framework documents specify how an implementation of ISO 10303 should be tested for conformance to the Standard.
- *Integrated generic resources* (Parts 40-49), as a group, provide a single information model for a manufactured product.
- *Integrated application resources* (Parts 100 - 199) are specializations of the Integrated Generic Resources for some general engineering requirements.
- *Application Protocols* (Parts 200 - 299) specify the requirements for data for a specific engineering application in a standardized representation derived from the Integrated Generic Resources. Application Protocols are implemented for use with the relevant engineering application software.
- *Abstract Test Suites* (Parts 300 - 399) describe the tests to be used to determine if an implementation conforms to the related Application Protocol. Each Application Protocol has an associated Abstract Test Suite with the number 3xx, where xx represents the second and third digits in the number of the Application Protocol. For example, ISO 10303-207 has an associated Abstract Test Suite with the number ISO 10303-307.

- *Application Interpreted Constructs* (AIC) (Parts 500 - 599) are sections of data models that describe concepts that are common to more than one Application Protocol. These parts are therefore intended for use by developers of new data models for the ISO 10303.
- *Application Modules* (Parts 1000 -) are small information models that are intended to be reusable in the development of future Application Protocols (AP). These parts are therefore intended only for use by developers of data models.

A list of the STEP Application Protocols (AP) as of June 2004 is given in Appendix I.

The ability to support many protocols within one framework is one of the key strengths of STEP. All the protocols are built on the same set of Integrate Resources (IR's) so they all use the same definitions for the same information. For example, AP-203 and AP-214 use the same definitions for three dimensional geometry, assembly data and basic product information. Therefore CAD vendors can support both with one piece of code.

Later on, a particular focus will be given to the Product Life Cycle Support standard, AP 239, which is the main subject of this document.

Each Application Protocol includes:

- a scope describing its purpose
- an activity diagram describing the functions that an engineer needs to perform within that scope, and
- an Application Requirement Model containing information requirements of the activity diagram. These requirements are mapped to a set of Integrated Resources and the result is a data exchange standard for the activities within the scope.

7.3. Goal

The ultimate goal of STEP is to cover the entire life cycle, from conceptual design to final disposal, for all kinds of products. However, it will be a number of years before this goal is reached. The most tangible advantage for users of STEP today is the ability to exchange design data as solid models and assemblies of solid models. Other data exchange standards, such as the newer versions of IGES, also support the exchange of solid models, but less well.

STEP led the way with three dimensional data exchange by organizing an implementation forum for the CAD vendors so that they could continually improve the quality of the solid model data exchanges. The history of this success is relatively interesting since it shows that the initial reluctance of vendors to implement user-defined standards can be overcome with enough perseverance.

In 1997 Ford, Allied Signal and STEP Tools, Inc. demonstrated the first successful data exchange of 3D geometry using STEP [17]. Once this basic capability had been demonstrated, a pilot project called AeroSTEP, was organized by Boeing and its Aircraft engine vendors to test the first translators by exchanging data about where an engine fits onto the airframe [18]. This project started out by exchanging simple faceted models but eventually demonstrated the exchange of models with great complexity.

The AeroSTEP project showed that STEP data exchange of 2D and 3D model data was both feasible and valuable.

7.4. Application of the standard

The ISO 10303, STEP, uses EXPRESS as the formal specification of the required data and its relationships.

STEP is primarily defining data models using the EXPRESS modeling language. The EXPRESS family of languages has been, and is being, developed under the auspices of ISO TC184/SC4[1]. EXPRESS itself is lexical language for information modeling, with some object-oriented characteristics. The EXPRESS Language Reference Manual also defines a graphical subset of the lexical language called EXPRESS-G which allows data specifications to be developed and reviews in graphical form. EXPRESS was originally developed to provide a formal, and computer processable, means of defining the data necessary to describe a product (anything from a microchip to a battleship) throughout its lifecycle, from time of conception through its manufacture to its time of disposal.

EXPRESS has two main features [19]:

- it provides a very general and powerful inheritance mechanism (much more than is provided in OO programming languages), for the modeling of data and data relationships, and
- it includes a full procedural programming language which is used to specify constraints on data instances.

EXPRESS-G is a subset of EXPRESS as it does not include the constraint portions of the lexical language. EXPRESS models may be written in the style of Entity-Relationship, CODASYL, Relational, Object Oriented, or other kinds of data modeling. It may also be considered to be a Set Theoretic specification language, and some have even gone so far as to indicate that it might be classed as a higher order predicate logic language.

Models described using EXPRESS are intended to be implementation independent. As well as providing some unique capabilities, EXPRESS has borrowed from many other languages including Ada, Algol, C, C++, Euler, Modula-2, Pascal, PL/I and SQL. It straddles both programming languages and database specification languages. Being lexical, the language can

[1] TC184/SC4 Committee of experts for “Setting the Standards for Industrial Data”

be compiled, and there are a number of both commercial and public domain compilers available. Typically, these compile EXPRESS into another high level language. Compilers have been developed on the one hand to generate C, C++, Prolog, etc., and on the other hand to generate DDLs (Data Definition Language) for both Relational and OO databases, such as Oracle, ObjectStore and Versant, as well as SQL. Some companies are using EXPRESS as the definition language for Data Warehouses. Software tools are also available that support modeling using either EXPRESS or EXPRESS-G and enable the transformation between the EXPRESS and the EXPRESS-G representations. EXPRESS-G is a subset of EXPRESS as it does not include the constraint portions of the lexical language.

7.5. Current Status and future perspectives

Several problems have emerged in recent years relating both to the huge scale of the ISO 10303 standard and to its incapability for following new technological developments as they occur. The latter is not easy to solve, because the ISO standardization process a large scale effort of many involved entities. International consensus-seeking and lengthy review and balloting procedures are to be followed for each standard. In consequence, different approaches for bringing documents to the International Standard status are being tried, and the entire structure of the STEP standard is currently in process of revision to make the development process faster and more flexible while requiring less voluminous documentation.

In information technology systems, the impact of STEP was great. The standard brought the “open data” concept in a area were proprietary formats still reigned. Users were provided an alternative when limited by applications typically used in design, engineering and manufacturing.

Nowadays, the use of XML technology seems to be the next step in the standard’s evolution for capturing and transferring STEP information. The ISO10303-28 application protocol already specifies the use of the Extensible Markup Language (XML) to represent EXPRESS schema (ISO 10303-11) and the data that is controlled by those EXPRESS schema. However, as stated in [20], by sharing information the original data is divided which makes it difficult to interpret in its XML format. This way XML looses its self-documenting properties, a situation that should be overtakenin the near future.

According to [20], the main issue responsible for stalling STEP’s development is the lack of commitment of both industry and governments. The government doesn’t interfere with the industry’s solutions, and in the industry, companies do not like to fund open standards that may be used by their competitors. Consequently, investment is done only in situations of desperate need or profits are assured.

8. PLCS

8.1. Introduction

The initial components of STEP were focused on the design and manufacturing phases of the product lifecycle and specifically on the need to exchange CAD files. An emergent need to represent a product through its life cycle, including requirement management, product operation and support phase originated the creation of a new international standard. A consortium called PLCS Inc. was founded in 1999, with the goal of developing an international standard that allowed the representation of information needed to operate and maintain an evolving product in an environment containing heterogeneous organizations, processes and IT systems.

The PLCS Inc. sponsors have continued to work together since the PLCS standard was published in 2005. This work has taken place in the context of OASIS[2], a non-profit, international consortium that supports the development, convergence, and adoption of e-business standards. The OASIS PLCS Technical Committee (TC) continues to provide the main guidelines for the development of Data Exchange Specifications (DEXs), and other assets important for PLCS development.

In the 21st Century, digital data is a valuable business asset. New business opportunities are emerging which require increasing focus on managing information throughout the complete product lifecycle. For example, many manufacturers wish to extend their core business and “go downstream” in hopes of generating additional revenue from the supply of lifecycle support services.

As product lifetimes and complexity increase, businesses are focusing on delivering specified availability and additional support services while the cost of ownership. Requirements for agility in the face of change are also increasing, both to combat product obsolescence and to continually improve products already in service. Another trend, driven by globalization, growing complexity, and the need to reduce lead times is the requirement to cooperate within an “Extended/Virtual Enterprise”.

These market trends create a business need for sharing information in real time. It is no longer practical, or realistic, to mandate the use of a single common system to overcome the problems with information exchange across an *Extended Enterprise*. A different approach is required.

[2] Organization for the Advancement of Structured Information Standards

8.2. Scope

The ISO 10303-239 [21] or PLCS is a major deliverable from a joint industry and government initiative to accelerate development of new standards for the exchange of assured product and support information, i.e., the information needed and created during the use and maintenance of complex products. The PLCS initiative was sponsored and managed by an international consortium of leading government and industry organizations, known as PLCS Inc. The ISO Technical Committee 184/SC4/WG3/T8[3] is the ISO working group responsible for the initiative. All information modules related to the new standard are complete and in the process of publication by ISO.

The main aim of PLCS is to provide means of managing essential information for product support. The AP239 is particularly appropriate for support of complex product such as ships, aircraft, engines, or oil platforms. These are intricate systems which are in constant change and so it is important to main information over their complete life cycle from concept through design and manufacture to operation and disposal. This extension to the STEP standard not only addresses the requirements of configuration management but also requirements needed for definition and supply of product life cycle support for complex assets. This includes the following requirements [22]:

- The identification and composition of a product design from a support point of view
- The definition of documents and their applicability to products and support activities
- The identification and composition of realized products;
- Configuration management activities, over the complete life cycle;
- The properties, states and behavior of products;
- The activities required to allow product operability;
- The resources needed to perform such activities;
- The planning and scheduling of such activities;
- The capture of feedback on the performance of such activities, including the resources used;
- The capture of feedback on the usage and condition of a realized product;
- The definition of support resources such as equipment, people, organizations, skills, experience and facilities;
- The definition of classes of product, activities, states, people, organizations and resources.

[3] OASIS Product Life Cycle Support Technical Committee

8.3. Goal

The goal of the PLCS initiative is to provide an information exchange standard that served as an extension to the ISO 10303 STEP capabilities. The following capabilities are enabled by the PLCS standard [22]:

- *Activity Management* – Capability to request, define, justify, approve, schedule and capture feedback on activities (work) and related resources.
- *Product Definition* – Capability to define product requirements and their configuration, including relationships between parts and assemblies in multiple product structures (as-designed, as-built, as-maintained).
- *Operational Feedback* – Capability that describes and captures feedback on product properties, operating states, behavior and usage.
- *Support Solution and Environment* – Capability to define and maintain the necessary support solution for a product in a specified environment including the opportunity to provide support (scheduled downtime), tasks, facilities, special tools and equipment, and personnel knowledge and skills required. PLCS will also relate organizations, personnel and facilities with the product needing support.

8.4. PLCS Concepts

According to [23], there are three central concepts in PLCS: product, activity, and resource. To these concepts, properties, states, locations and conditions may be added to characterize them. Conditions may be applied to relationships between these concepts. Many entities, attributes, and relationships can also be qualified by applying:

- Classification: using class attributes. External class libraries may also be used as reference data.
- Characterization: retaining information on who defined a particular value and under which circumstances.
- Justification: explanation for a current value or relationship.

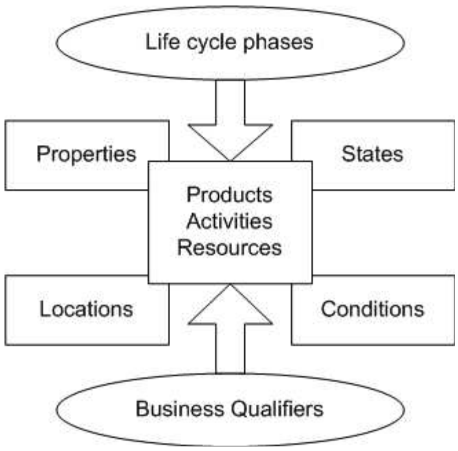


Figure 19: Central concepts in PLCS [23]

8.5. PLCS Components

PLCS has two main components:

- An **activity model**, which contains a set of processes which support optimization over the full product life cycle.
- An **information model**, which specifies the information necessary for integration and exchange of data required by the processes described in the activity model.

The ***PLCS activity model*** illustrates the processes and information flows in the PLCS scope. It shows how product support information can be managed along the product lifecycle, while providing a single information source. Although not mandatory when applying the standard, the activity model is particularly useful when defining processes and responsibilities in collaborative projects and within organizations.

The *Activity Model* that has been developed for a STEP AP serves as the basis of an information model which represents the information produced by or used by the activities described within that *Activity Model*. It is this information model that provides a formal representation of the data structures that are shared or exchanged during the life cycle of a product. The Figure 20 shows how a STEP AP defines the scope of the information to be exchanged or shared, and offers a range of exchange formats independent of technology to support that exchange. These formats define the requirements for translators applied to specific systems.

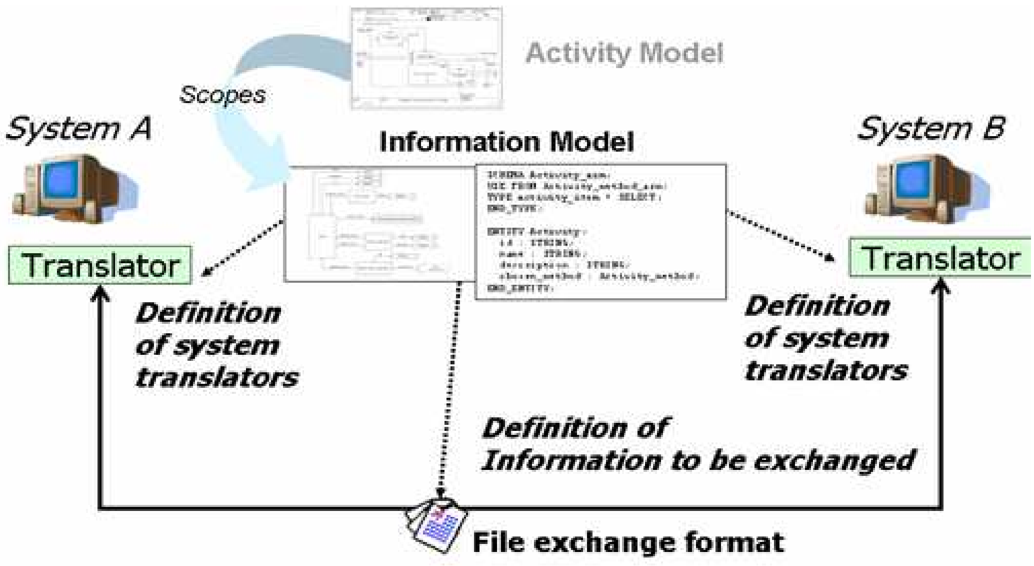


Figure 20: The role of a STEP AP [24]

The information model of ISO 10303-239 PLCS is capable of representing all the information required to cover the entire life cycle of a product. It is not however, aimed at any particular business or industry domain. Because it is designed generically enough, it can be used in many different business applications. While this fact ensures that ISO 10303-239 PLCS has a wide range of application, this also implies that different users will focus on different areas of the model. Additionally, different users, with different information requirements, may interpret the information model in different ways. In order to address these issues, the PLCS community has developed a number of mechanisms for partitioning the information model into smaller components and for providing additional (and more precise) semantics that add business specific terminology.

The first mechanisms developed are the *Data Exchange Specifications* (DEXs). These are subset of the ISO 10303-239 PLCS model that support information exchange and sharing requirements of particular activities. These are built from reusable components (called *Templates*) that ensure interoperability between DEXs.

A set of *Reference Data* has also been developed, which provides greater semantic context to the ISO 10303-239 PLCS model and which may be further tailored to particular business needs. The set data is stored in a *Reference Data Library* (RDL).

To ensure that the PLCS standard is used correctly, a set of usage guidance documents called Capabilities has also been created.

The ***PLCS information model*** (see Figure 21) covers [24]:

- The identification and composition of a product design from a support viewpoint.
- The definition of documents and their applicability to products and support activities.
- The identification and composition of individual products.
- The configuration management activities over the complete life cycle.
- The definition of activities required to sustain product function.
- The resources needed to perform such activities.
- The planning and scheduling of such activities.
- The capture of feedback on the performance of such activities, including the
- resources used.
- The capture of feedback on the usage and condition of a product.
- The definition of the support environment in terms of people, organizations,
- skills, experience, and facilities.

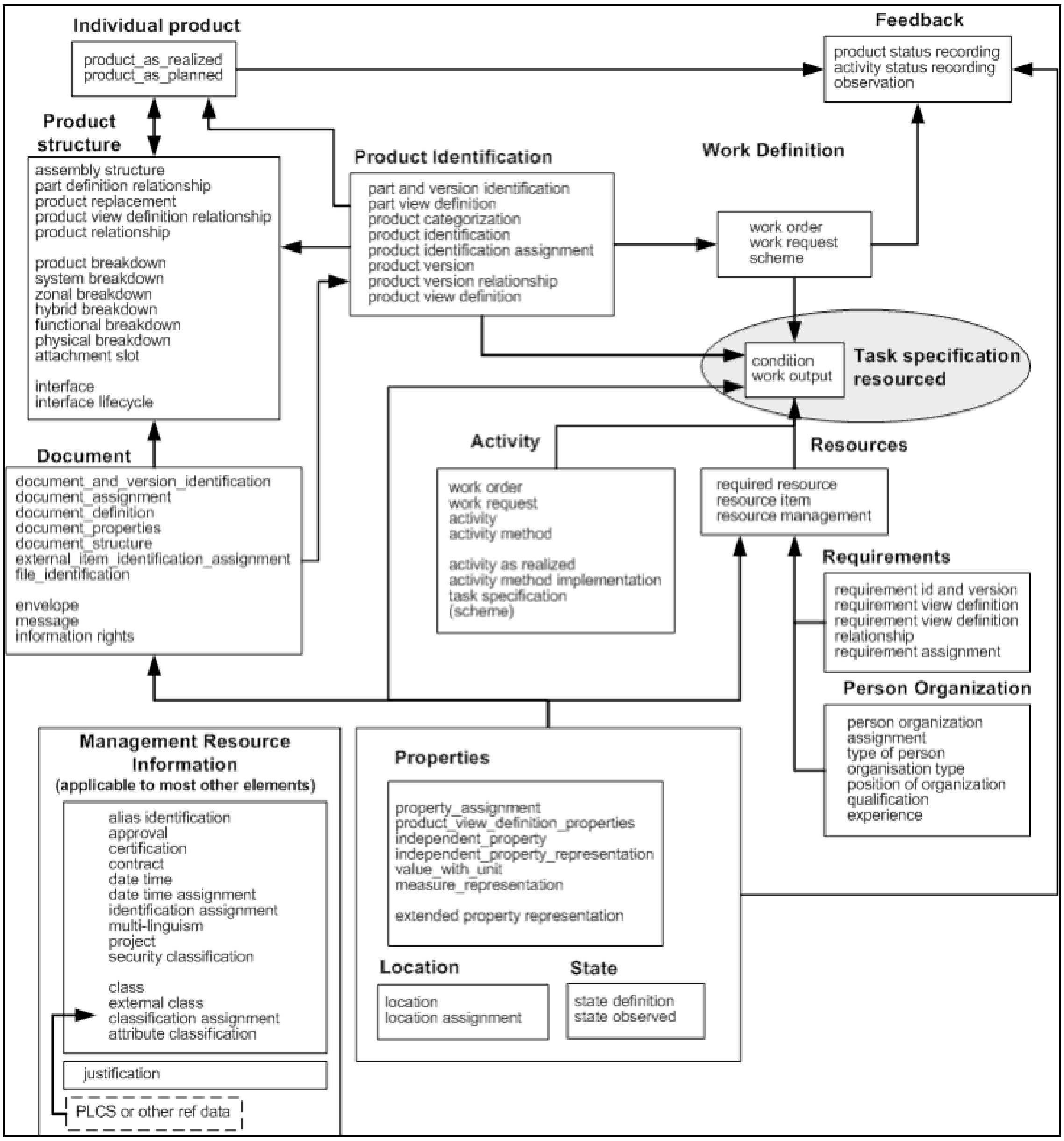


Figure 21: Schematic representation of PLCS [24]

8.6. PLCS Implementation

8.6.1. Principles

The information model in PLCS is meant to be a generic model that can be easily tailored and extended to a specific domain situation and so it is intended for use in many different business domains. Its extensibility is provided by using reference data which are used to apply classifications thus specializing the generic model.

One example can be given using the “activity” concept. This concept is provided in the information model in a very generic way. If needed, this concept can be specialized if for example, it is important to distinguish between an “operational activity” and a “support activity”. Then, “operational activity” and “support activity would be two new classes with their own reference that are applicable to the generic concept “activity”.

8.6.2. DEX

A *DEX* represents a “chapter” in the ISO 10303-239 (PLCS) information model. Each DEX is a section of the standard suited for a particular business context. A *DEX* provides guidelines and rules on how to use and combine entities with external *Reference Data*, for usage in data exchange. Each DEX includes a complete EXPRESS schema, a subset of the ISO 10303-239 schema that is used to define and validate a data exchange file. A DEX defined and managed by the OASIS PLCS TC may be referred to as a "PLCS DEX", as opposed to a "Business DEX". A DEX can be considered as the primary resources for implementers. DEXs can be also used to contract for information, and software applications may declare conformance with a particular DEX or set of DEXs, thus ensuring interoperability.

The DEX specification [24]:

- identifies the subset of the PLCS standard information model that is applicable for the given business context;
- provides guidance for the application of the PLCS standard in the given business context;
- details extensions to the PLCS standard are provided by the DEX for the given business context.

The first step in implementing PLCS for a given exchange requirement, is to discover whether an existing DEX meets that requirement. It is then necessary to determine whether that DEX needs to be adapted to suit the local business context. Experience shows that some local tailoring will almost always be required in order to reflect the precise nature of the information to be exchanged, any local business rules, and any elements of reference data which are specific to the local context.

The PLCS development community worked on methods, tools and guidance to assist implementers to adapt existing DEXs or develop their own. This way the community and developers in general take a common approach in PLCS development by using the same base of DEXs.

8.6.3. DEX Architecture

The PLCS DEX architecture is encompasses of the following components:

- AP239
- DEX Specifications
- Templates
- Capabilities
- Reference Data

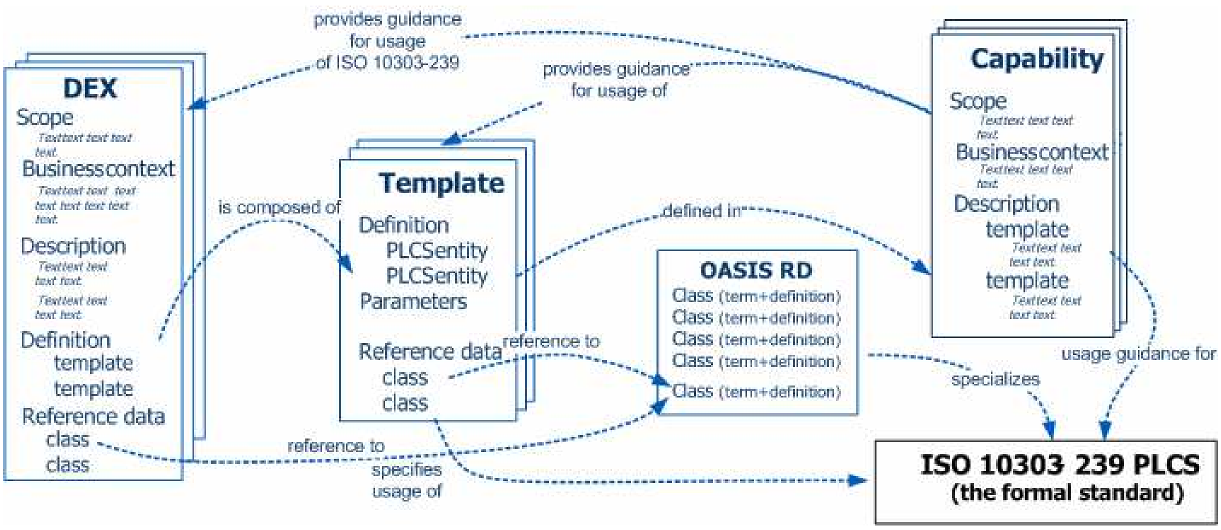


Figure 22: DEXs, Capabilities, Templates, and Reference Data [24]

As already mentioned, the PLCS architecture can be easily extended to include specific business domains. To achieve this, developers can use the following components:

- Business DEXs
- Business Templates
- Business specific Reference Data

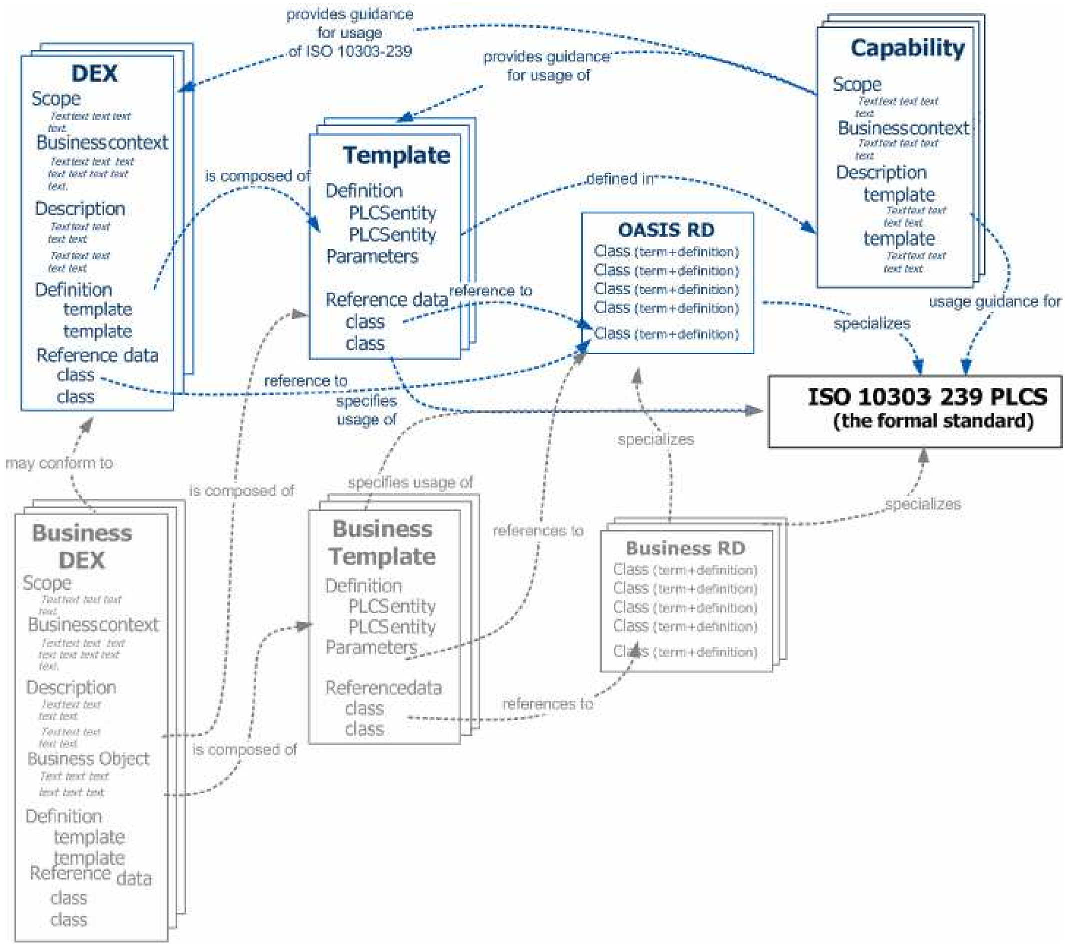


Figure 23: DEX architecture extended with business specific constructs [24]

8.6.4. Capability

A *Capability* is a description of how the PLCS standard information model can be used. The *Capability* provides guidelines on how *Templates* should be used to represent different entities in the information model.

It provides guidance and rules on what *Entities* should be used to represent a given concept, how the entities should be related, and what *Reference Data* should be used. Additionally, information is also given on which set of *Templates* are defined within a *Capability*, in order to specify which *Entities* should be instantiated to represent different concepts.

A *Capability* is a description of how EXPRESS entities are used to represent a given concept It provides guidance and rules on what Entities should be used to represent a given concept, how the entities should be related, and what *Reference Data* should be used. As well as general guidance, a set of *Templates* are defined within a *Capability* to provide precise specifications of which Entities should be instantiated to represent identified concepts.

8.6.5. Template

A *Template* provides a model for the specification of how entities and their attributes can be instantiated with the PLCS standard. Additionally, it also indicates which *Reference Data* should be used, to represent a concept in a *Capability*. *Templates* which are provided and managed by the OASIS PLCS TC are simply referred as “PLCS Template” while user defined ones are “Business Template”.

The role of the *Template* with the PLCS architecture is to describe and express functionality with a *Capability*. While a *Capability* documents and provides information on concepts and hw they related, *Templates* describe the concepts themselves. An additional particularity of the *Template* is that can be reused in other *Templates* which are referenced by other *Capability.*

For example, the *Template assigning_reference_data* is used in most other Templates (assigning classification data is a fundamental part of ISO 10303-239 PLCS in OASIS).

8.6.6. Reference Data

As aforementioned, the PLCS standard was designed to be easily extendable and adaptable to many different business contexts. For this purpose, PLCS provides basic constructs which represent basic entities. Through the use of these constructs and by building relationships between them, one can create more complex and specialized semantic definitions. For example, using the basic constructs part and date, one can created a more specific entity like “creation date” which is not part of the set of basic constructs. Thus, complex or specialized constructs are not represented directly, but precision can be achieved by adding a classification scheme to the basic constructs using *Reference Data*, thus refining the meaning of the entity.

For example, there are many different types of properties associated with components that are relevant to product life cycle support. The information model defines the basic representation of a property on a part and enables the properties to be classified. The classification mechanism is then used to define the specific type of property. The reason for using classification in this way is that any set of specific property types, such as "mean time to failure" that could have been provided by explicit modeling in the standard is likely to be incomplete. Furthermore, as business practices change, different properties are likely to be required over time and these can be introduced by means of a new class, without having to re-issue the standard.

Reference Data can be defined as a set of basic class definitions representing concepts that can be used to defined entities in the information model. This way, *Reference Data* is used to give more precision of entities making them suitable for the given business context. These classes are kept externally from any particular information model. A *Reference Data Library* (RDL) manages the collection of classes available as Reference *Data*.

8.6.7. Business DEX, Templates and specific Reference Data

A Business DEX is a DEX that has been defined outside the OASIS PLCS TC. It is, generally speaking, more business specific and may extend or specialize a PLCS DEX. A Business DEX is defined and managed within a specific Business Context. A Business DEX is a full specification of the technical aspects of a data exchange, and fulfills the technical requirements of an exchange agreement made by two or more Organizations.

In the same way, a Business Template is very similar to a PLCS Template, in that it defines PLCS entities and their relations, and uses PLCS Templates as needed to simplify this. The only difference is that a Business Template may use both PLCS Templates and Business Templates whereas a PLCS Template can only use other PLCS Templates.

A Business RDL is a Reference Data Library that specializes the OASIS RDL further in order to achieve precision to describe a specific business context. A Business RDL is a Reference Data Library that specializes the OASIS RDL.

In summary:

- A DEX defines a subset of the ISO 10303-239 (PLCS) information model against which software can be conformant to and data exchange contracts can be written;
- The building blocks of DEXs are Templates that are reusable across different DEXs and which ensure consistent representations of the same core concepts;
- Capabilities provide guidance on the use of the information model and partition the information model into different high-level concepts - which are further detailed as Templates;
- Entities are ultimate "atoms" of the PLCS information model;
- Reference Data adds more specialized semantics to the information model Entities;
- Business DEXs define business specific Templates using extensions to the core PLCS Reference Data.

8.7. Example - Aviation Maintenance DEX

One of the DEXs available was developed for a very specific purpose, the maintenance of aviation equipment. The aviation maintenance DEX [24] specifies the use of the PLCS standard while representing work that has been carried out on complex products such as aircraft. This DEX represented information such as:

- the item on which work has been done;
- the work that has been done to item;
- the state of the item;
- properties of the item, such as number of flying hours.

As mentioned before, PLCS has two main components, the activity model and the information flow. Activity Model is a representation of business activities using the IDEF0 definition language. The activity model is presented as a set of diagrams and a definitions of the activities and their associated information flows. A DEX supports a subset of the activities and information flows. Next figures present the activity model and information flow for the support activities during maintenance. The arrows highlighted in yellow are the information flows that are supported by this DEX.

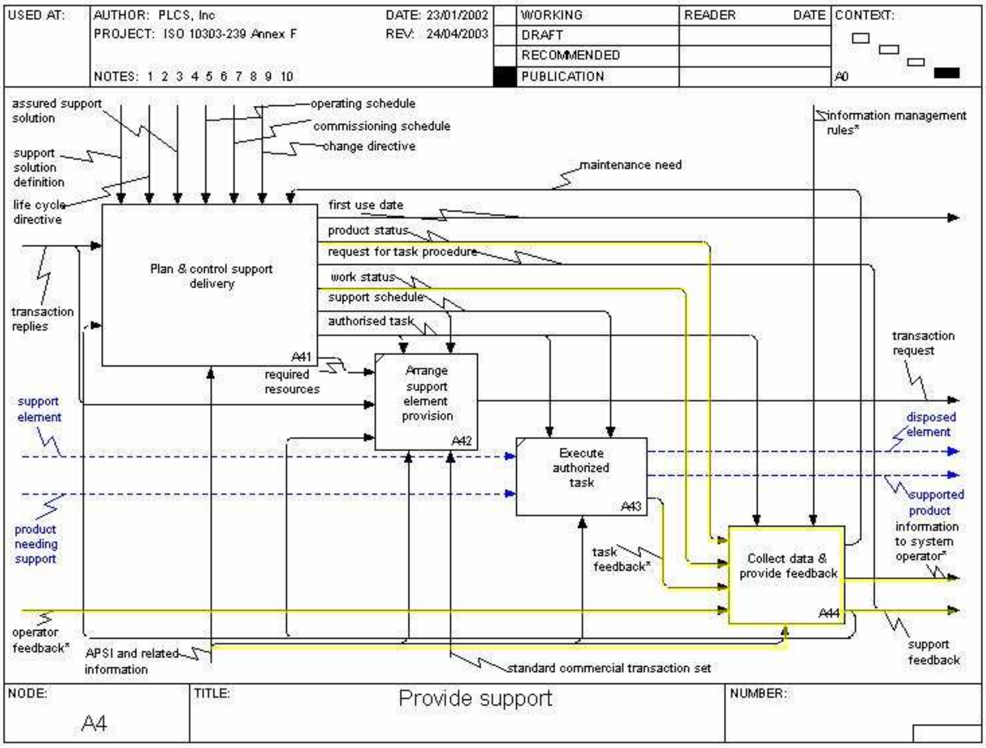


Figure 24: Activity model for "Provide Support" [24]

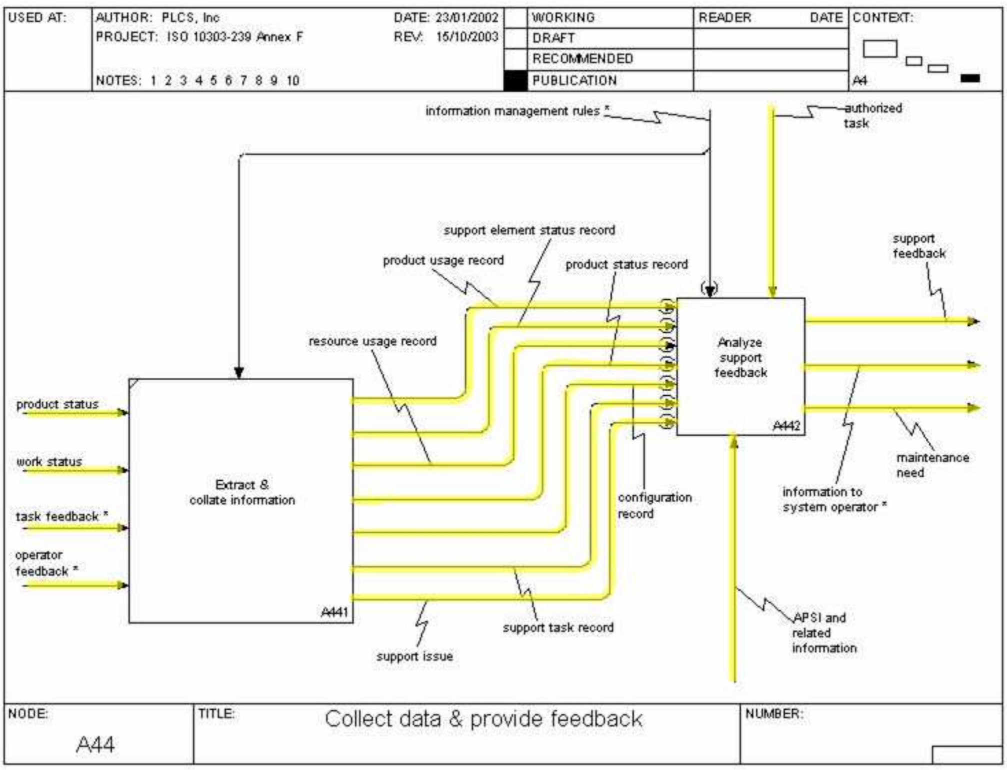


Figure 25: Activity model for "Collect data and provide feedback" [24]

The information exchanged by the Aviation Maintenance DEX is normally between an aircraft operator and a maintainer. It is assumed that each party has access to the *product operational information*: the procedures and technical information needed by operators and support personnel to operate, maintain and dispose of the product. This includes operating procedures, security procedures, maintenance procedures, as built product structures, spare parts lists, and disposal methods. This means that *product operational information* may need to be exchanged between DEXs. Each DEX can have a set of related DEXs defining complementary activities. The aviation maintenance DEX, for example, has the “Product Breakdown for Support”, “Faults related to products”, “Product as individual” and “information related to a single task” DEXs related to it.

8.8. Business Model

As mentioned earlier, a DEX is defined for a particular business context. The business process supported by the *Aviation maintenance* DEX is shown in the UML activity diagram in Figure 26 which shows a product being used, the authorization of scheduled or unscheduled maintenance, undertaking the maintenance activity and then updating the aircraft status.

Information about the activities stored into the DEX including details of the maintenance activity undertaken, by whom, on what date. Information about the product on which the activity was done, (the reportable item), such as what was replaced, what were the life measurements of the product, e.g. hours flown.

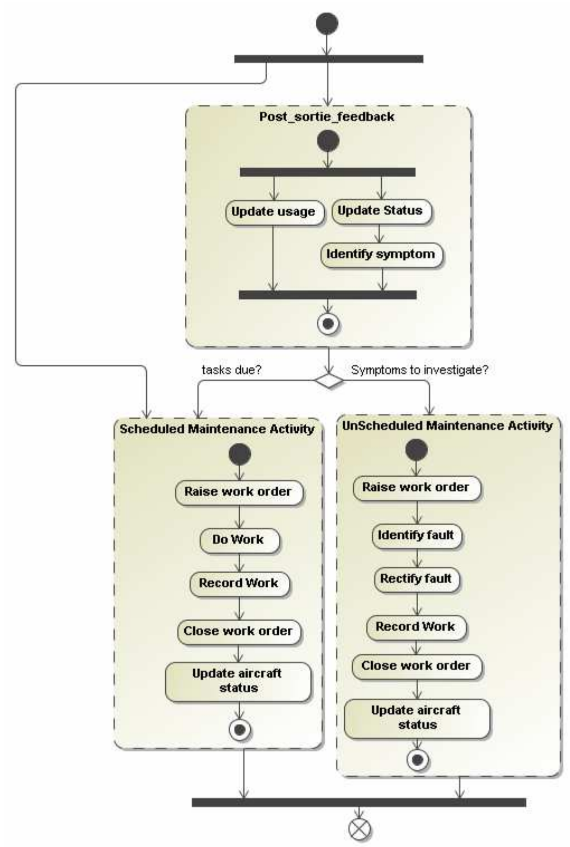


Figure 26: Typical business process using the Aviation Maintenance DEX [24]

8.9. Data Exchange

Before the DEX is deployed, it is necessary to analyze the exact business process being supported identifying possible data exchange points along the process. This may vary from business to business. Figure 27 shows the previously provided activity diagram with some examples of possible data exchange points identified [24]:

- Sortie (flight of a combat aircraft on a mission) / post usage feedback: reporting after an aircraft has flown a sortie;
- Consumption of life: reporting when the life of a reportable item has been consumed;
- Ownership change: reporting after an aircraft has changed ownership;

- Geographical location change: reporting after an aircraft has changed geographical location;
- Serviceability change: reporting after an aircraft has been removed from/returned to service;
- Symptom: reporting when a symptom has been identified;
- Scheduled maintenance: reporting when scheduled maintenance starts and finishes.

Unscheduled maintenance consisting on: reporting when unscheduled maintenance starts and finishes. Depending on the business there may be a series of triggers reporting of activities performed during the maintenance, when parts are fitted, removed, exchanged, faults identified etc.

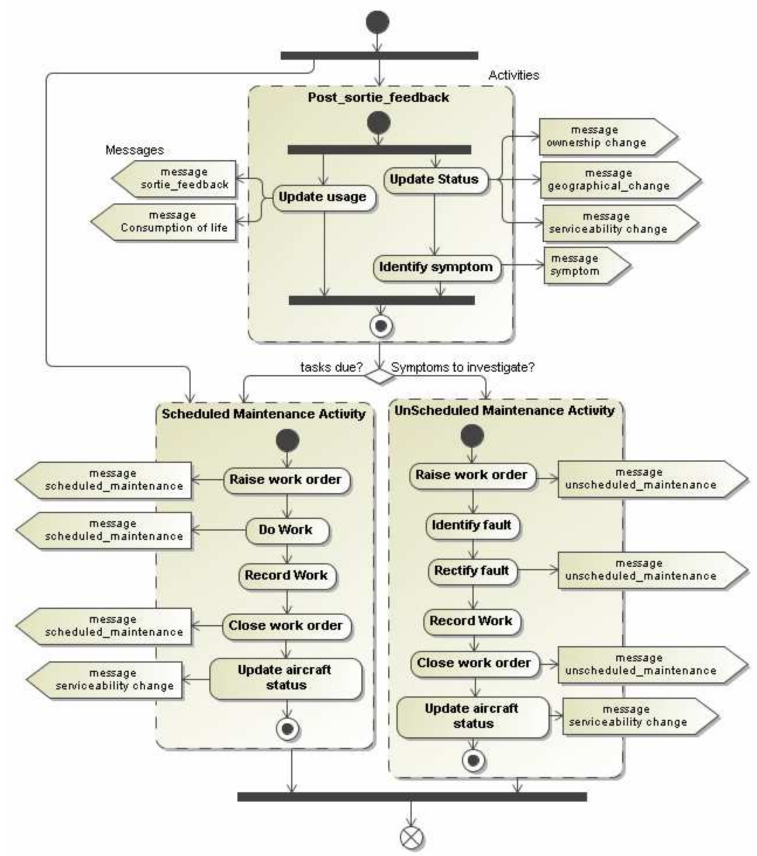


Figure 27: Example exchange points [24]

Based on the Business Process and Data Exchange representations, and using already defined data representation templates, a PLCS representation of maintenance feedback information can be built.

The following Figure 28 provides a detailed description on how to represent an aviation maintenance feedback DEX using ISO 10303-239 PLCS. This is defined using PLCS capabilities, templates and PLCS reference data. This definition may be further tailored by specific parties by extending the reference data defined in the PLCS reference data library.

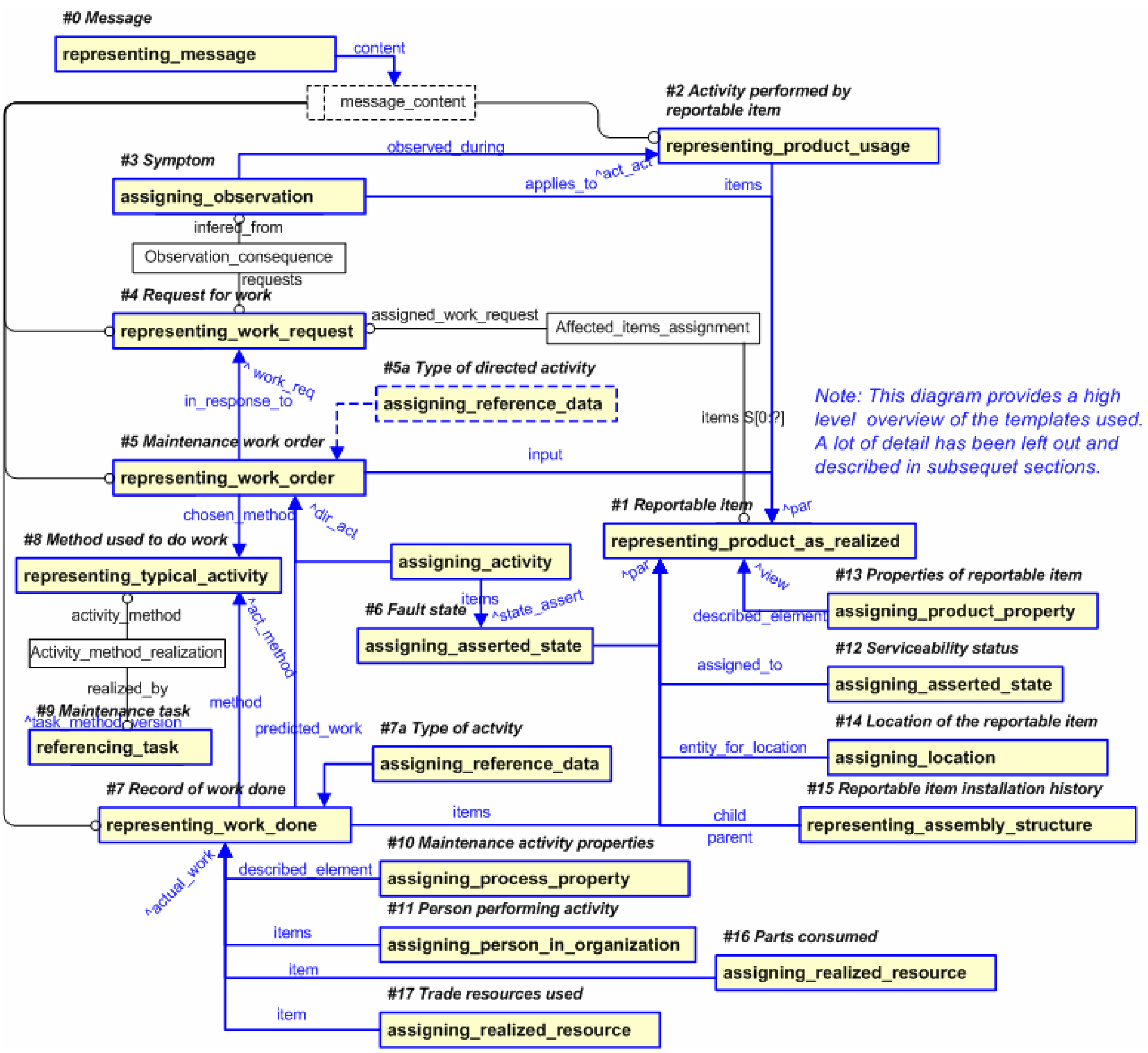


Figure 28: Overview of PLCS constructs used to represent maintenance feedback information [24]

The previous example uses the "assigning activity" template. This template describes how to assign an activity to something in a given role. For example, a maintenance activity that has been performed on a product is recorded by assigning an *Activity_actual* to a *Product_as_realized*. The "assigning activity" template is represented as show in Figure 29 using the EXPRESS-G notation.

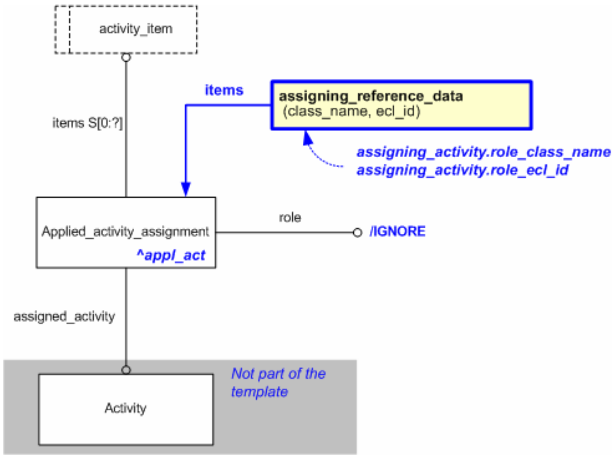


Figure 29: An EXPRESS-G representation of the information model for "*assigning_activity*" [24]

8.10. Implementation Methods & Tools

As a standard, PLCS does not force implementers to take particular approaches to PLCS implementations. However, there are some more common than others and, naturally, it is for the most popular ones that most of tools are developed.

Under OASIS[4] the standard language used for DEXs is XML according to ISO 10303-28 while the STEP file format is using EXPRESS language (ISO 10303-11).

There are some PLCS toolboxes available in the market, which simplify programming and provide high-level interfaces for DEX template handling. These are particularly suited for creating interfaces to legacy systems such as ERP and PDM and enable import and export of data in PLCS format. The PLCS toolboxes for C++ and for Java platform have been deployed by Eurostep and are available in their website [25].

A set of PLCS Web Services have also been proposed to provide a standardized interface to multiple applications. These services have already been used in several projects as part of a Service Oriented Architecture. The PLCS Web Services Client Software Development Kit (SDK) from Eurostep provides an environment which facilitates the rapid development and deployment of PLM applications. It can be used to create customized applications for PLM systems, without any need for customization of the PLM system itself. Using the PLCS Web Services Client toolkit, a programmer can easily develop PLM applications (in Windows) which can access data from a PLM, and other applications, using the PLCS Web Service API. Standardization of

[4] Organization for the Advancement of Structured Information Standards

implementations via the PLCS Web Services reduces the number of APIs required, with subsequent savings in development and maintenance costs.

9. Business Benefits of PLCS

The recent changes in the market brought more competition between companies. OEMs (Original Equipment Manufacturers) can no longer be limited to their core activities. Nowadays, they have to reach the whole product lifecycle in order to maximize the return from their products. By going beyond the initial product sale OEMs have new opportunities to potentiate the return on the investment done on the product. So, companies are naturally turning to product support seeking increase of revenue. By covering this part of the product life cycle OEMs are not only trying to make more profit but also to differentiate themselves from other competitors, expecting an increase of their customers' loyalty.

Additionally, by providing product support, these companies gain knowledge on how their product behaves in real life scenarios providing useful information that can be applied in the design of the next generations of the product.

To be able to provide product support, OEMs need to have access to information about the product after its delivery. These are the typical questions that need to be answered to provide an efficient support [23]:

- What products are delivered and where?
- How is each asset currently configured?
- What state is it in, regarding faults, wear and usage?
- What work is needed and by when?
- What resources – people, facilities, spares, tools and test equipment – are required to perform this work?
- Are modifications or upgrades due?
- Are changes needed to the role configuration?

Effective support needs access to complex and durable information that usually reside in the owners/operators internal systems. Consistent profits in the aftermarket businesses rely heavily in establishing connections to customers in order to allow them to access and manage the total information set appropriately. These data is highly interconnected, reside on many different IT systems, ownership is tricky to handle and data evolves at fast pace. Data may have a life cycle of minutes, or need to be managed for decades.

The key concept in PLCS is the creation of a "single source of truth". In an implementation point of view, this concept represents a central common repository where data coming from different sources will be integrated and consolidated. In an *Extended Enterprise* scenario, to realize this concept it is vital to consolidate information created in the different IT systems from the different partners.

The use of the PLCS standard in the automation of data exchange enables companies to reduce their costs significantly. The automation itself reduces error occurrence and consequently data quality is improved. According to [23], by using STEP standards the US automotive, aerospace, and shipbuilding industries have reached accumulated benefits of $1.1Bn for an investment of $100M over 16 years while current annual benefits are estimated at $156M per year. The potential for gains in the aftermarket are of similar, if not greater, scale.

The automation and standardization of interfaces not only potentiates the re-use of data but also leads to savings on resources used in re-generating the same information for different users. Manual entry of data as a major source for error occurrence is also eliminated and consequently the manpower previously expended will also be less.

The main factor in favor of standardization is the re-use of data. Through the use of a standard format exported data can be easily re-used by others. Using standardized interfaces allows systems to be upgraded or replaced without affecting other systems. This is particularly important in an extended enterprise scenario where data comes from many different systems and owned by different organizations.

The most immediate consequence of PLCS application is data exchange. With the consolidation of data coming from different sources into a single environment, business partners are able to communicate much more efficiently. They can share the necessary information, notify partners of changes and receive notification of changes to data published by others.

Business intelligence also benefits from the data consolidation. Within the collaboration environment, value can be added by recording important relationships between different data sets. This enriches the information held and enables valuable queries across information from many sources. For example, by combining information from PDM, ERP, and Logistic Support Analysis systems, it is possible to identify which individual assets, and which maintenance documentation, are related to a specific part version, and, hence, to a potential change. Since business decision making is high-dependent on the amount and quality of information it is provided, it can only benefit from a single source of consolidated and exploitable data.

Effective information management combined with fast and efficient communication between business partners will lead to successful collaboration in the face of change. The price of using erroneous or out dated information can be high. In a competitive market such as today avoiding unnecessary costs may lead to huge savings.

PLCS also potentiates the upgrade of legacy systems. By using web services outdated systems can easily improve or expand its functionality. It is important that the different systems of the business partners provide a common set of functionality so that data exchange even for all participants.

In conclusion, these are the business benefits of PLCS [23]:

- product data can be re-used by other companies
- improved data quality allows for better decisions in all aftermarket activities
- the right work is done using the correct information
- Maintenance costs are reduced
- Improved feedback
- The modification process can be accelerated and effectively managed
- Adaptability to changes in product performance or intended usage

10. Share-a-Space, an example of Extended Enterprise concept

10.1. Principles and implementation approach

One of the most famous examples of the implementation of the Extended Enterprise concept, through the use of the PLCS standard is the Share-a-Space system. The Share-a-Space can be seen as both a PLM-enabling solution and a business process for a neutral environment where companies in a partnership may work together throughout a product's life.

The system is a commercial tool, mainly focused on collaboration and enabling companies to work together in their own systems and internal business processes while working together in an inter-enterprise environment. It complements many existing engineering and product-related solutions like Requirements Management, CAD, CAE, PDM, ERP, Integrated Logistics Support (ILS), Technical Document Management providing additional value to the enterprise and its business partners.

The Share-a-Space [26] system is built on the STEP standards including the STEP Product Life Cycle Support (PLCS) initiative or ISO10303-239 making it suitable to be used in service-focused industries such as aerospace/defense, automotive, machinery and telecommunications. So, in the core of this system there is a STEP-based information integration, consolidation and sharing solution that use several STEP Application Protocols such as AP214, PDM schema, AP203, 210, 212.
Share-a-Space can be seen as the Information Manager described in the Extended Enterprise concept chapter. In generic terms it is a data hub and a set of system adapters that centralize information from multiple heterogeneous systems.
The system is partly implemented using a Web-based Enterprise Application Integration (EAI) technology. This technology can be described as the use of software and computer architectural principles in order to integrate a set of enterprise computer applications.
EAI is particularly used to link enterprise applications such as Supply Chain management, CRMs, business intelligence and other types of applications that typically cannot communicate in one another. Like this, such applications are able to share data to simplify and automate business processes to the greatest extent possible, while at the same time avoiding having to make sweeping changes to the existing applications or data structures.
As mentioned above, at the core of Share-a-Space is a rich STEP model which earlier releases used STEP's AP214 and more recently with PLCS is using AP239. This enabled the system to

cover the full lifecycle collaborative work integrating product data, consolidating it and providing an exchange platform.

To successfully support the complexity of today's virtual organizations Share-a-Space must provide business process independence.

10.2. Architecture

The following information is digested from [27]

Share-a-Space's architecture is essentially based in XML for data exchange and a set on Web-base technologies to provide the platform independence such system needs.

The foundation of Share-a-Space is a Oracle database based on ISO 10303-214 and 239 standards providing a neutral information repository. It also provides a very rich data model with around 500 database objects implemented that allows the representation and consolidation of information gathered from multiple sources without disturbing the integrated systems and processes. Share-a-Space uses the STEP's EXPRESS modeling language for the definition of its database. The EXPRESS model is then used as a base for the UML model that will support the development of the application server's functionality.

As illustrated in Figure 30, Share-a-Space is based on a layered client-server architecture.

The Oracle database is run on Linux, Unix or Windows environment with the Share-a-Space application server being run on a MS Windows machine. The Web Services layer is not only the most important but also the most used. This mechanism is particularly used for XML data transfer from and to any other Web-enabled system. Share-a-Space's Web Service layer currently supports ISO10303-239 as well as ISO10303-214. This layer can be used for both remote controlling the system using an API or as a Service Oriented Architecture (SOA) component.

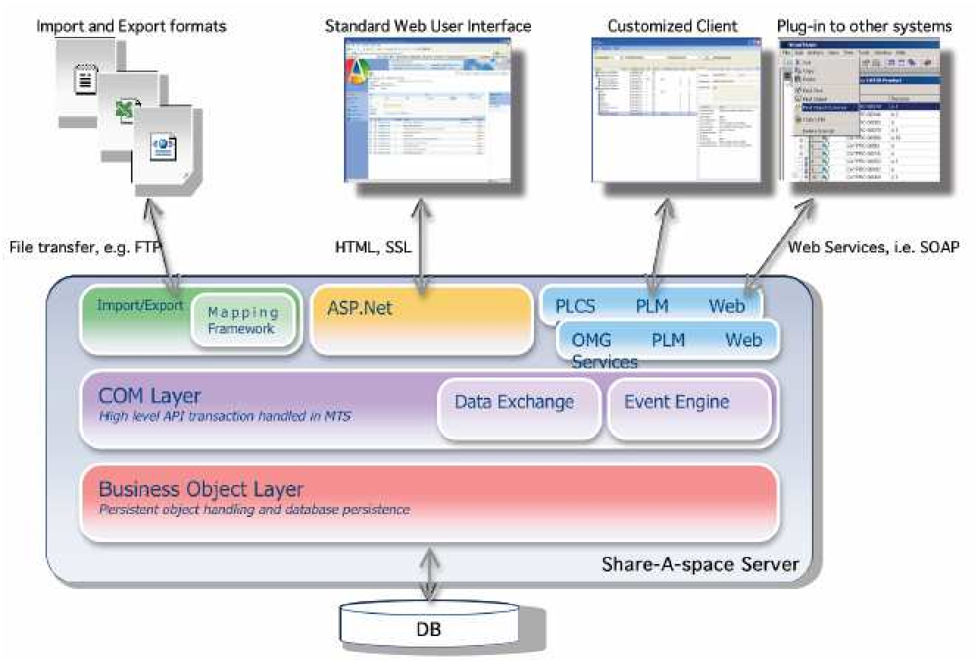


Figure 30: Share-a-Space architecture [27]

11. Case Study

The industry is evolving and data exchange is nowadays a very important issue in business making. Project partnerships are made every day and information sharing can have a crucial role in the success and efficiency of collaborations. Data exchange between different companies is not a new phenomenon. Business partnerships appeared a long time ago and so did the need for information flow among partners in a project. However, until fairly recently data exchange could be quite a tricky and "painful" task. There are many information system solutions in the market and they are scattered in all types of industries. Moreover, within the same industry the same software product can have different configurations in different companies. Consequently, different companies may use different format to represent data which in the end can make exchange quiet difficult.

As presented before, the industry itself came up with the solution to this problem by developing standards that enable and boost information sharing between partners. The concept behind STEP and later on PLCS is that data exchange can be done among different companies without having to change their internal business processes. If they all commit to the same standard, if a international one even better, they can all profit from an easy way of sharing data while keeping their own internal "way of working" intact.

For the construction of the LHC, the biggest machine in the world, CERN had to establish partnerships with external companies, particularly manufacturers. To manage and provide access to all data concerning this project, an information system, EDMS/MTF, presented earlier in chapter 2, was developed.

The purpose of this chapter is first to give a general view of how data is managed within EDMS/MTF, presenting its information flow and system's architecture. The final aim is to study the need and feasibility of applying the PLCS standard to such information system.

11.1. The role of CERN in the supply chain

The Large Hadron Collider (LHC) project is the single most ambitious scientific experiment ever created. Even CERN, as the world's largest scientific research center, simply did not have the capacity to single-handedly design, build and manage an instrument such as the LHC. Consequently, CERN had to establish partnerships between other research institutes and private companies to develop the project.

During the whole process of building the LHC, CERN plays the most important role since oversees and carries out the whole lifecycle of the project from planning to dismantling.

For the cryodipoles as for other LHC components, CERN is responsible for the design but the manufacturing is then contracted to external companies.

However, CERN not only acts as client for external companies and institutes, by receiving the components built and installed, but also acts as their supplier since it's from CERN that manufacturers get their "raw materials".

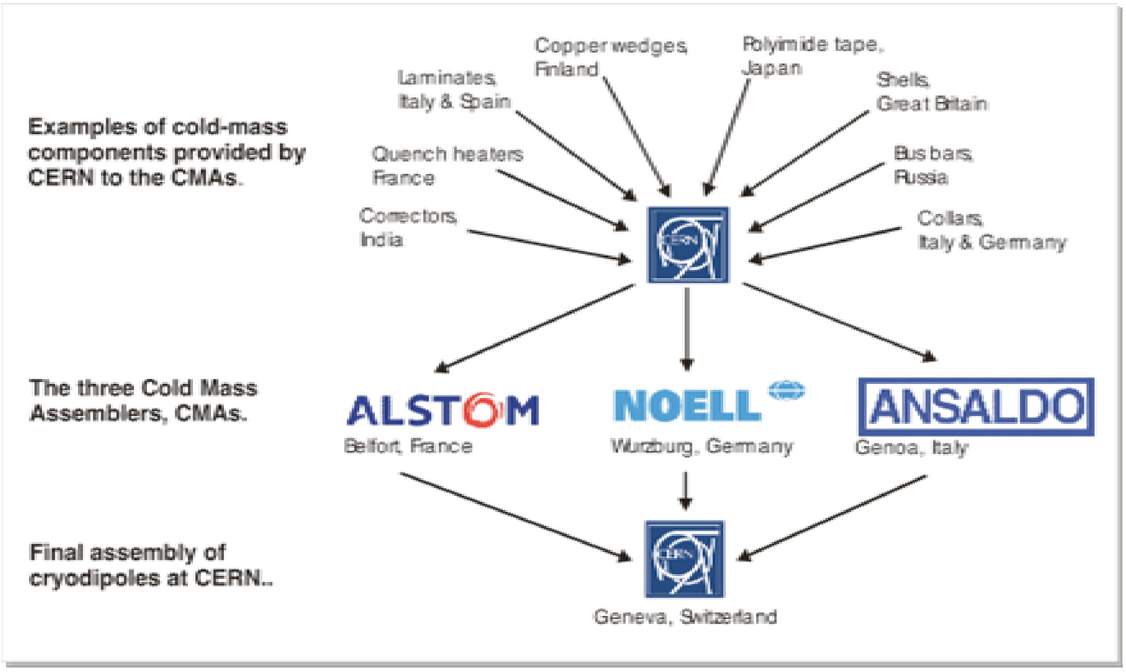


Figure 31: CERN - the supplier of its suppliers [28]

In assembling a Cryodipole, the most important part is the cold mass assembly. At this stage, the assembly work is sub-contracted to three different companies. These *Cold-Mass Assemblers* (CMAs) obtain the parts to be assembled from CERN (supplier of its supplier).

The whole assembly process of LHC components is subject of high quality standards. This is a direct consequence of the complexity of the project, which obviously demands high rigorousness but also due to CERN qualification as an *Installation Nuclear de Base*. Such elevated standards are also expected for collaboration entities within the project.

Thus, CMAs have been carefully selected and their work supervised in respect to the high quality standards defined. The interaction between CERN and external entities did not limit themselves to exchange of supplies and assembled components. During the assembly process, companies had to report with detailed information on manufacturing steps done and test results produced.

Additionally, quality inspectors were contracted to make sure CERN's quality assurance definitions were being respected. These inspectors supervised CMAs production process including components to be assembled and were also responsible for their stock.

11.1.1. Product and information management

The interaction between CERN and external entities has then two components: product flows and information flows. The three main reasons for the high importance of continuously capturing of information throughout the production and assembling for the cryodipoles are the following [28]:

- Cryodipoles are complex components which use advanced technology, are quite expensive and consequently their specifications are very strict. It is important to be able to follow all steps of their manufacturing, particularly in case of non-conformities detected.
- As components of the LHC, is it essential to have detailed technical information in order to integrate them with other components.
- The dipoles' manufacturing and assembly information is relevant for planning and scheduling future work.

To manage all information flow during the lifecycle of a product including production follow-up and test results management, CERN created and application called Manufacturing and Test Folder. As mentioned in chapter 2, this tool as part of EDMS/MTF Service, allows to trace and manage the data of all LHC equipment during its whole lifecycle. This has the way CERN found to control all the manufacturing process while integrating information coming from different sources into one single system.

Figure 32 shows an example of an information flows during the assembly of a Cryodipole and how it is managed within MTF.

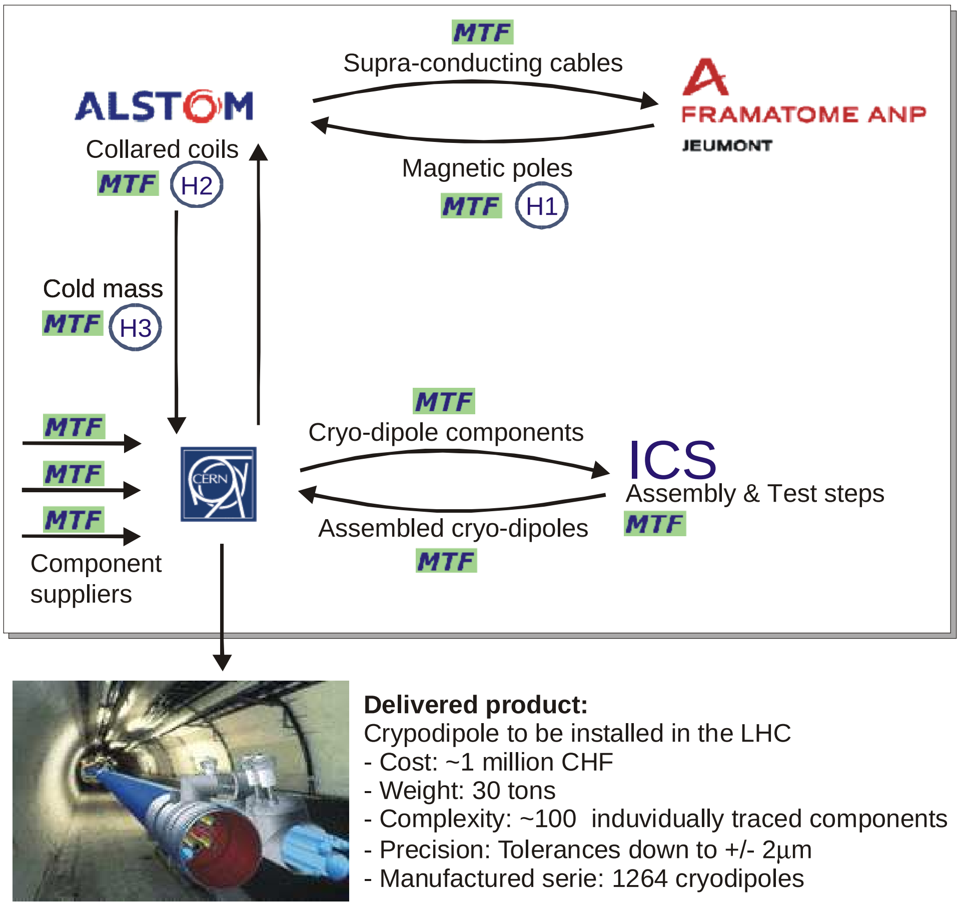


Figure 32: Example of the Supply Chain for the cryodipoles [28]

Two companies, Alstom and Framatome ANP have formed and alliance for the manufacturing of one of the Cold-Mass's components. Alstom is responsible for producing supra-conducting cables which are then provided to Framatome ANP to produce magnetic poles. This company provides these components back to Alstom which will be used in the Cold-Mass assembly. In the production of the magnetic poles, information regarding each supra-conducting cable is traced in MTF, which has to be approved before being used at Framatome. In the same way, the resulting magnetic poles must also be registered in MTF and accepted being used in the Cold-Mass.

During the whole assembly process, MTF traces all pieces of equipment using their identifiers and configuration within the assembled Cold-Mass including assembly steps that might have been reported.

11.1.2. Challenges

With a project of such dimension mechanisms have to be put in place to make sure everything runs as smoothly as possible. The challenges encountered are mostly directed connected with the complexity of the project in hand:

- Large number of entities involved through out the whole construction
- Distribution of suppliers, manufacturers and assemblers – wide geographic distribution makes collaboration more difficult.
- Different internal business processes – every company has its own "way of working" and project management should be flexible enough to integrate all of them
- Quality assurance impositions – companies have different level of quality assurance rules put in place. In this project CERN imposed the rules to be followed and so companies had to adapt.

Resulting from these factors, the assembly process is fairly error-prone. To attack this problem, CERN focused in defining strict rules to control the assembly process, hoping to detect error occurrence as earlier as possible in the process. This was done by requiring exhaustive testing exhaustive testing and verifications of the manufactured equipment as well as by hiring external QA inspectors located in the manufacturers. As mentioned earlier, their job was to supervise manufacturing and controlling components being used in the assembly.

CERN also defined a set of quality assurance rules meant to be strictly followed by manufacturers. By using the quality metrics and demands a quality standard must be met and all non-conformities can be easily detected.

Additionally, CERN created the MTF application which enable full control, traceability and supervision of the activities performed.

11.2. EDMS/MTF Architecture

In chapter 2, the EDMS/MTF was introduced by being presented as a PLM system which uses two commercial tools to achieve full coverage of the product life-cycle.

EDMS/MTF is then composed of three main modules (see Figure 33):

- *Axalant* – a tool for Product Lifecycle Management
- *Datastream 7i* – a tool for asset management
- *As-Built Data Management* – a module that connect the two previous ones. It makes the relation between the representation of an equipment and documents in Axalant with its representation in D7i.

Figure 33: EDMS/MTF main modules [29]

The approach the EDMS/MTF development team used to choose the right tools for the job is called "best of breed". The "best of breed" approach seeks out easy-to-integrate solutions where the product strengths are aligned with the needs and requirements of the organization. Because each solution is not intended to solve the entire problem, but rather a particular part of the problem, it is more difficult to find a weak point in the system.

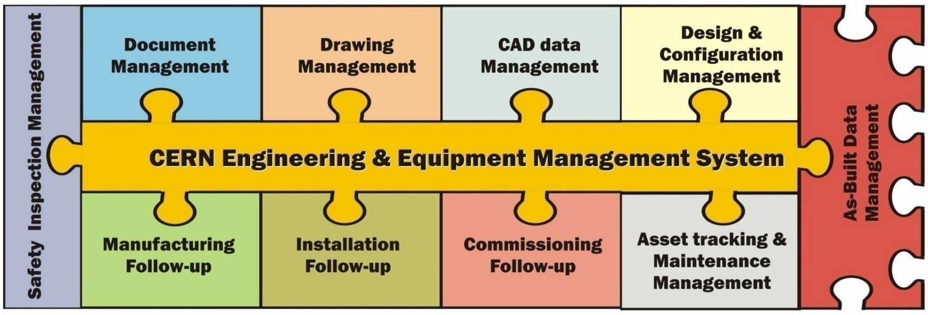


Figure 34: The "best of breed" [30]

The problem with this approach is that a link between the two applications had to be developed, since the idea behind is to obtain one system. In EDMS/MTF this is done both at the database and interface levels.

Both EDMS and MTF share a common interface intended to provide seamless navigation to the user. Although two different systems are being used the user as the illusion of using a single one.

At the database level the link between objects in Axalant and D7i is done in a set of database tables. On the Axalant side there are four tables and a view that make the link between the two systems. The *c_equ_dat*, *c_slot_dat*, *c_sys_dat* and *c_loc_dat* tables are use for *Equipment*, *Slots*, *Systems* and *Location,* respectively. There is also the *c_obj_dat* view that provides the result of all the enumerated objects linked.

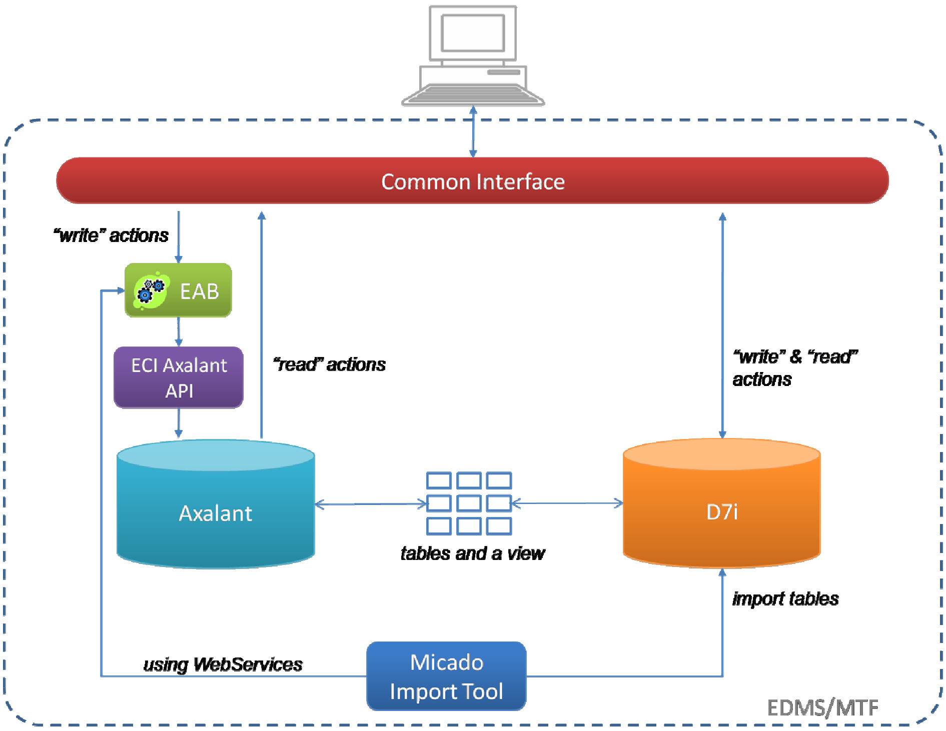


Figure 35: Simple view of EDM/MTF architecture

EAB [31] or EDMS Action Broker is an intermediate application between EDMS and the Axalant system. This application makes the link between EDMS interface and Axalant PLM. It is written in "pure" Java technology and it is the core of EDMS system.

The EAB application uses a multitier non-distributed architecture. Figure 36 shows us the partitioning of the application tiers and the technologies chosen for each tier. For a collocated architecture (where all the application components reside within one JVM), the presentation, business-logic, and integration tiers are physically located in the same Web container. Well-defined interfaces isolate each tier's responsibility. The collocated architecture makes the application simple and scalable.

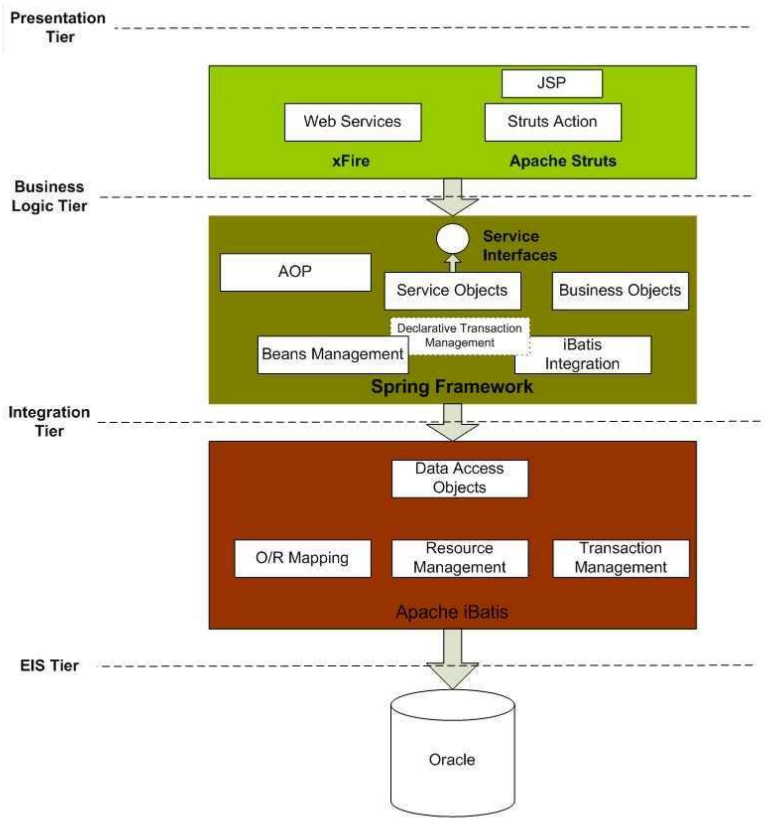


Figure 36: EAB architecture [31]

For the presentation tier, experience shows that the best practice is to choose an existing, proven Web application framework rather than designing and building a custom framework. EAB uses Struts as its Web application framework while for handling Web Services requests the open source framework xFire was chosen. The access control is handled by Acegi Security, a powerful and flexible security solution that provides authentication, authorization, instance-based access control, channel security and human user detection capabilities.

To build the business-logic tier EAB uses POJO or Plain Old Java Objects through the Spring framework.

The integration tier handles the data persistence with Oracle database. It uses the Apache iBatis framework and the DAO pattern for implementation abstracting and encapsulating all access to the data source.

For the data imports, one of the most important features of EDMS/MTF, a specific module was development called MICADO [32]. The designation is an acronym for "**M**TF **I**mport **C**hain to **A**void **D**ata **O**verdose". As specified in this title, the MICADO software system is primarily designed to import data into MTF. The need for such import tool came for the increasing amount of data arriving for import. As the LHC project progressed, more import requests were made, coming from different CERN entities. Additionally, the data verification and import processes consumed much effort from the EDMS Support team, which could be redirected to other important tasks.

Consequently it has been decided to develop MICADO, a software tool that automates the import work as much as possible, and minimizes the intervention of the EDMS support team. The development of MICADO aimed mainly at reducing to a minimum the intervention of the support team and delegating as much as possible the import task to the user itself. Three requirements were established to achieve this goal: the user should focus on the data and consequently the import process should be opaque; the software tools used (Axalant, D7i, etc) don’t allow the user to represent his data in a flexible way but MICADO should; the system should detect and report errors in the data.

Currently, MICADO can handler four main EDMS/MTF entities: Equipment, representing physical equipments as build and installed; Slots, which are “placeholders” where these equipments are physically installed; Steps, which are operations to be performed to manufacture and test equipments; and Jobs, representing operations performed on slots (installation, operation and maintenance).

However, these entities are handled differently by the import tool since the data representation model used is also different. To import data on Equipment/Steps, the Excel file template solution was adopted. The Excel files used to submit Equipments/Steps data follows a strict structure (columns, rows, sheets). These files may contain “hidden” data that is essential for the following verifications required before the data import. The file’s structure is defined by EDMS Support according to the type of Equipment to be imported. This template is then provided to the user who fills it with data and submits it for import. After the submission, MICADO/Equipments converts the Excel file to XML and proceeds with the import. In some rare cases, the users prefer to submit data directly into XML form.

For Jobs/Slots it was decided to use an XML representation for the data exchange. Users are in charge of encoding their data with the XML format imposed by MICADO. This may be a manual or automatic generation process. XML representation of Jobs/Slots is simple and can be managed easily by most users. A XML schema is provided to the users so that they can validate there data against the schema.

More recently, a Safety Inspection import feature was developed. Safety inspections are now performed by external companies and, consequently, there was a need to import data on the inspections. This feature uses the same import mechanism as Equipment data import.

The import process records data in 2 distinct information systems: MTF and Axalant. To import into MTF, “import tables” are used which correspond to the various objects managed by MTF: for instance slots, jobs and their properties. These tables are filled in with import data which, after a verification process, will be used to fill the appropriate Oracle tables. The operation may be rolled back if any errors occur.

For Axalant imports, MICADO uses the EDMS action broker (EAB) when ”write” actions are required such as document creation, file upload, etc. To access data from the database the import mechanism uses special stored procedures.

11.3. EDMS/MTF Workflow

During the whole lifecycle of the Cryodipole (presented in chapter 11.1), as most of the LHC equipment, many entities exchange information between themselves. This information exchange is mostly managed by EDMS/MTF. From the drawings to the safety inspection reports, all documentation related to LHC equipment is stored within the system.

In the two following sections a general view of the whole process all along the Cryodipole life cycle will be given so that later on the information flows between CERN and external entities are presented.

11.3.1. Design

At CERN, two main CAD systems are used for design work and installation, accelerator and detector layouts. The one being used at the moment is Euclid, currently owned by Dassault Systemes. Euclid is able to model in both for 2D and 3D. CERN also used AutoCAD from AutoDesk fundamentally for 2D modelling.

Until the development of EDMS, drawings were managed using an in-house application called CDD or CERN Drawing Directory. CDD provided some of the functionalities currently available in EDMS such as document distribution, reviewing and approval.

CDD proved particularly useful since many designs were developed in outside institutes and industrial companies and the application served as a centralized remote repository where all drawings were stored and managed. In this repository the HP-GL/2 plotter language is used as the neutral file exchange format in addition to the native CAD files. Since there were many intervening entities, particularly due to the LHC project, common standards and repositories to store and manage data should be used. This is the reason behind the HP-GL/2 format adoption as a single file format to be used. Typically, different companies and institutes used their own CAD systems, standards and quality assurance procedures. However, in order to homogenize, the data representation format, the HP-GL/2 was imposed in order to avoid a situation in which each entity uses its own format.

In order to identify a particular drawing when archiving it in the CERN database, a drawing serial number and approval date are inserted in a blank are especially reserved for the effect. Like this each drawing is uniquely identified and associated with the corresponding project. The registration of a drawing in the CERN database can be either done through EDMS Web interface or drawings can be send to CERN and imported by EDMS Support.

By using common standards agreed with external entities for managing drawings, CERN, through EDMS, provided a standardized approach of storing, handling and displaying produced drawings.

After the drawing has been registered in EDMS, it has to do be archived in the system. The archival process can either be launched manually or triggered automatically when the drawing has passed its formal approval process. For drawings produced at CERN's design offices, the storing process starts automatically with the creation in batch mode, of a HP-GL/2 plot file of the drawing in a particular CAD system. For drawings created in other High-Energy Physics (HEP) institutes or design office's, as CERN cannot support all available CAD systems, drawings must be provided in supported plot file formats. When the archiving process starts, the drawing status and approval information is added to the drawing. This modified file in then uploaded to EDMS, where from then on, is available for viewing and edition. For this purpose an application was developed called CERN HP-GL viewer.

During the design phase, unfinished 2D and 3D CAD designs must exchanged between CERN and external sources. Drawings created on-site or in external design offices must be accessible to collaborators who may need to follow being done, in order to be able to perform their part in the project.

As it was impossible to access drawings at CERN without using the Euclid system, a new feature was developed. Since drawings where kept in a Euclid database, this new tool allow users to retrieve them having to recur to Euclid software. Data extraction became system independent. This CAD exchange feature is accessible through a webpage. It basically converts a given CAD model into a selected format using the appropriate CAD tool for the conversion. The *Consult*, as it is called, enables the user to download and upload data from/to CERN using several supported file formats.

11.3.2. Manufacturing, Installation & Commissioning

As mentioned before, due to the LHC project, CERN developed an in-house application to capture manufacturing and test data. Today, the MTF system manages thousands of equipment along with its production data and non-conformity issues. The main requirement imposed for the development of such system was that it should provide traceability of all pieces of equipment and its parts in a geographically distributed environment. The idea behind this requirement was that all manufacturers should be able to entry data concerning the production and assembly of the components. MTF can manage a wide range of data such as information about manufacturer, manufacturing procedures, test results, non-conformities notices and other related documentation have to be stored and retrieved from nearly any place in the world.

In addition, the MTF system has also workflow capabilities. It is able to manage information concerning the various manufacturing steps and test performed, as well as, documentation produced through-out the whole manufacturing process. Information about scheduling, applicable standards, results and possible non-conformities can be stored and managed individually for each step.

Apart from all the manufacturing concerned information, MTF is also able to provide support for the INB (Instalations Nucleaires de Base) traceability regulation activities.

The first phase in working with the MTF for a certain type of equipment is to define its profile in the system. This means defining what the equipment does and how to manufacture it. This involves the definition of all the important properties for the equipment as well as the manufacturing and test steps that are relevant.

Once the equipment profile has been defined, it can be instantiated for a particular piece of equipment. These pieces of equipment are then uniquely identified by the system at the time of their registration. MTF has a mechanism able to generate unique identifier as equipment is instantiated.

In addition to equipment traceability, MTF is also able to associate documents with equipment. This feature is particularly important since it enables users to describe activities performed during and after the manufacturing process by using documents which are then linked with the given equipment. These documents can contain data on workflow, the manufacturing process, tests performed, manufacturing steps and non-conformities detected. Additionally, while already in operation, equipment can be subjected to safety inspections which inspection report may also be linked to the equipment.

11.3.3. Operation & Maintenance

The CERN Safety Commission (SC) is responsible for safety related issues at CERN. SC has amongst other duties the obligation to inspect machining tools, technical infrastructure and personnel workplaces with a certain periodicity for each equipment category. Buildings and traveling cranes, for example, have to be inspected once a year.

All new equipment arriving to CERN has to be subjected to an initial inspection before it is put in operation. It is called a reception inspection.

In total, six different types of inspections are carried out (reception, periodical, on demand etc…). An inspection report describes the objectives, make comments and a conclusion of the inspection. Remarks focused on specific problems found are made along specific recommendations for corrective actions. The inspection reports are transmitted to the person responsible for the equipment or installation (“owner”) and others who may be concerned by one or more remarks. If a corrective action or a modification is required, the “owner” has the responsibility to ensure that the SC recommendations are acted upon. This process is called the SC *remarks solving process*.

During this process, progress and difficulties can be reported. Once the solving process is considered complete, the remarks are “closed” by the “owner” and returned to the SC inspector who verifies and confirm that the problem is, or is not, actually solved.

The report itself, the details and any photos it may contain, must be easily accessible and associated with the equipment or installation data. Full traceability also requires that the information exchanges are recorded.

MTF’s web interface was developed with the purpose of providing its users an easy way of assessing and managing equipment data during the manufacturing process. However a web interface is not particularly useful when large volumes of data are to be imported and manual entry is a painful task.

The first solution found was the use of Excel worksheets by manufacturers, which were then sent by email to EDMS Support, responsible for importing them into MTF. However, the increasing number of request demanded an also increasing effort from support and so a new solution was need.

For this reason an import tool was developed, called MICADO (MTF Import Chain to Avoid Data Overdose). This new tool imports data in batch mode, using “templates” (Excel worksheets) created for each request. These files are then filled in by manufacturers which return them back to CERN to be imported. The import job is added to the import queue, and when it’s ready a mail notification is sent. Data to be imported must be extensively verified and validated before it is inserted in the database and irretrievably changes the equipment information stored.

11.4. EDMS/MTF Information Flow

During the construction of the LHC there were hundreds of companies involved from manufactures to suppliers, inspectors and CERN personnel. Throughout the whole process a huge amount of data was produced and exchange between these entities and managed by the EDMS/MTF service. As seen before this system manages information and traces equipment through the whole life cycle of the LHC.

11.4.1. Planning, Specification & Requirement Management

Before the actual construction of the LHC or its component equipment, planning, specification and requirement elicitation has to be done. As a result of this phase, for the LHC, the *LHC Baseline* was created. This document contains the main guidelines to be used in the construction of the LHC with some basic calculations and measurements. The same procedure was followed for the component equipment. These plans, specifications and requirements are then sent to the entities responsible for their approval and other actors in the project.

11.4.2. Design

In this phase, most of the data exchange is done between CERN and different design offices, both internal and external. The first designs of equipment to be manufactured are usually done at CERN's design offices. After these designs are created, they are sent to external design offices for further detailing. The resulting drawing may be even sent to the future manufacturer's design office so that measurements and other detailing can be done just before the manufacturing starts. The resulting design is also typically the final one which is sent back to CERN for final approval. These are files exchanged in HPGL or even in paper format.

11.4.3. Manufacturing

During the manufacturing phase the information exchange between CERN and external entities is mainly done on issues encountered during the actual manufacturing process of equipment. MTF provides functionality to follow-up the manufacturing of different equipment allowing manufacturers to report on test results performed on equipment, to report on equipment ready for delivery and last but not least, non conformities (NCRs) detected. The EDMS/MTF provides a notification mechanism. The detection of NCRs demands a response from CERN with the solution to the problem found. The NCR solving process is fully traced within MTF. After the equipment as been manufactured, duly tested and if no NCRs have been detected, the equipment is subjected to a final validation by an external Quality Assurance inspector. The final report is also sent to CERN.

11.4.4. Installation

The LHC site is, as expected, is a very busy place. Lots of workers are constantly circulating in the site and equipment being manoeuvred. As a result, the installation of new equipment has to be scheduled taking into account the work being done in the site.

MTF also manages equipment in the installation phase by providing manufacturers the time slots and equipment availability information they need to perform their duties. In parallel, reporting on the installation of the equipment is done by the responsible company using MTF installation follow-up features. Any NCRs that rose during this phase all managed in the system and accessed by both CERN and external companies.

11.4.5. Maintenance

In case of equipment breakdown or need for scheduled interventions, these are also registered in MTF associated with the specific equipment. After the intervention, the responsible entity should provide maintenance information on problems encountered, actions taken and others.

When in operation, equipment can be subject of safety inspections. The inspections are kept in MTF as a job, while in EDMS a safety inspection report is stored containing the tests performed and their result.

The following Figure 37 summarizes the information flow between CERN and external entities:

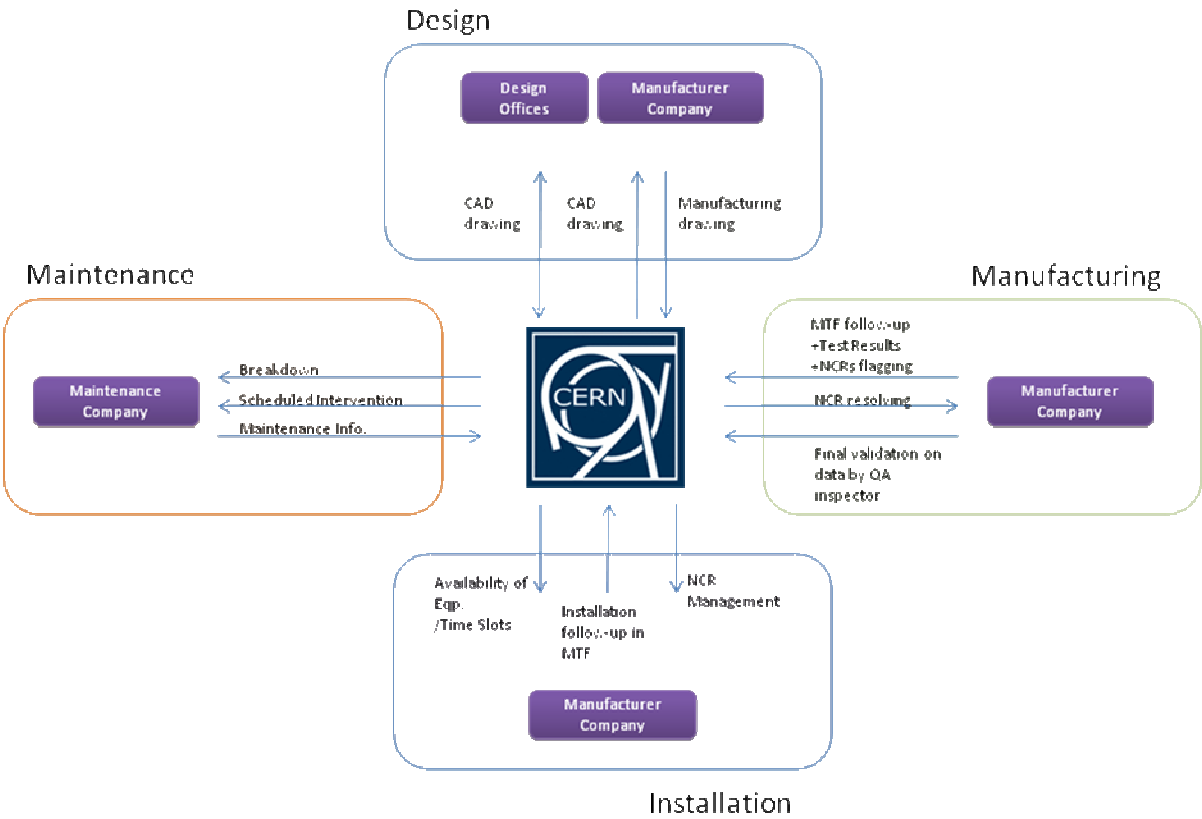


Figure 37: Information Flow

11.5. EDMS/MTF, Collaborative workspace

A collaborative workspace or shared workspace is an inter-connected environment in which all the participants in dispersed locations can access and interact with each other just as inside a single entity. These are typically enabled by a shared mental model, common information and a shared understanding by all of the participants regardless of their physical location. For this common space to exist there has to be an agreement between the entities involved as to how this shared workspace will be managed, what are the rules to use it and how will it impact the internal business processes of each participant.

The virtual enterprise concept requires one single business process, preferably completely independent from the intervening companies' methods. In an ideal situation, this single business process will be defined and adopted by the participants with little or no adaptation of the business processes being used internally by the intervening companies. Unfortunately, this is seldom the case.

So, before starting such venture, an assessment and feasibility study has to be done, studying the pros and cons of such a project.

The longevity of the project may be the first decision factor to take into account. It is obvious that it is simply not worth implementing such a system for a short-term collaboration. Another important aspect is the impact this new implementation will have in the current business processes and IT systems being used by the participating companies. A collaborative workspace that will force companies to reinvent their way of working will surely not be the best of choices.

Therefore, it is important to carefully study and evaluate the balance between the time span of the collaboration and the investment needed to create the extended enterprise trying to achieve a good return on investment.

For the LHC project it was obvious from the beginning that it was going to be a long-term project involving a great number of different companies, many of them never worked together before/after the project. Due to the complexity of the project it was vital that these companies worked very closely together and assuring great rigorousness in their work and a minimum service quality required. One of the components of the service provided was the data. Each operation performed should be electronically registered and eventually shared with other participants that might need it to perform their own tasks. Consequently, there was a need for a system that not only allowed the companies to register the work being performed but also assuring the quality of data being input.

11.6. Applying PLCS to EDMS/MTF

When considering applying the PLCS standard to the EDMS/MTF system, two implementation options appear as the most natural:

- The first option is changing the current system so that it implements the standard.
- The second option is that only an extension to the current system would be developed expanding its functionality but keep the current functionality intact.

These two options represent the two extremes in PLCS implementation, meaning, taking into consideration full implementation of PLCS within the system or the development of an external plug-in to the system implementing the standard. Since ISO 10303 is a modular standard it would be possible to make the implementation gradual or even only implementing the parts that are interesting for the given business model.

The implementation of PLCS involves, as expected, an investment from the owner of the system which value depends on the complexity of the system and how far the implementation of the standard goes.

As in other IT system implementations, a careful study and planning has to be made. Whatever the kind of implementation desired it is important to make a feasibility study and analyze the return on investment.

Furthermore, different approaches to implementation may be taken depending on the specific requirements that are to be met. As part of the initial development of STEP, four different levels of implementation are identified [19]:

- level 1: passive file transfer;
- level 2: active file transfer;
- level 3: shared database access;
- level 4: integrated knowledge-base.

- **Level 1 (passive file transfer)** exchange describes the most simple exchange process using STEP shown in Figure 38. A pre-processor translates data from the internal format of the sending system and encodes the data using the STEP physical file format. The result file is transferred to the receiving system, where a postprocessor translates the data to the internal format of the second system.

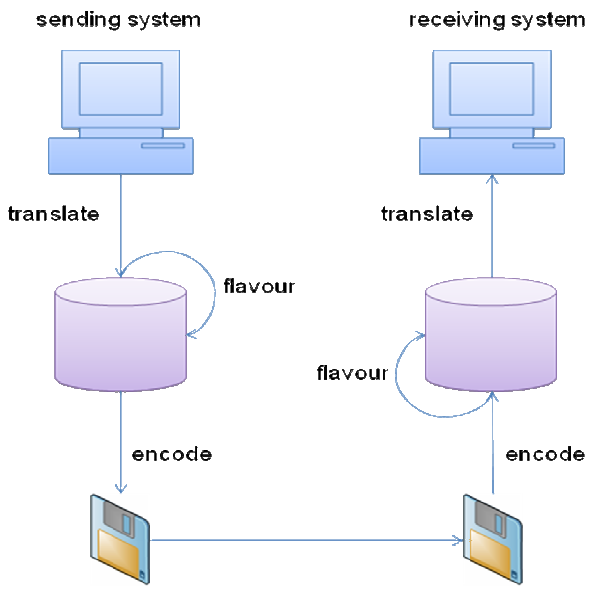


Figure 38: Data exchange process [19]

- **Level 2 (active file transfer)** is an extension to the first level of implementation. In the level 2, the data in the sending system is translated into a "working form" or "intermediate form" that allows selection and modification of the data during the translation process. This is the most common of file-based implementations of STEP since most of them make use of this approach to a greater or lesser extent. The distinction between level 1 and level 2 implementations has effectively disappeared.

Level 1 and 2 implementations apply the advanced features of the STEP standard and its associated technologies to the problem of data exchange between systems using neutral files. STEP and other standard such as IGES are, nevertheless, solving the same problem. However, since STEP standard provides a higher level of functionality and a neutral file exchange format (e.g. IGES file format is based in FORTRAN requirements), it can be considered to be a better solution. Level 3 and 4 implementations, however, represent not only a significant advance in technology, but also address a different set of problems. They don't simply focus on the data exchange problem; data sharing is now the main concern.

- A **level 3 (shared database access)** implementation of STEP combines translation with data access. In a first stage, data is converted from an internal format (CAx system format in a basic STEP implementation) into a STEP format. However, rather than storing the STEP data in a file for the purpose of exchange, the data is stored in a shared database or repository. The data access element of a level 3 implementation is a standard interface to this underlying database, which allows applications to store, manipulate, and above all share the data in a standard fashion.

- **Level 4 (integrated knowledge-base)** implementation envisions the combination of STEP with knowledge-based systems and artificial intelligence (AI) which makes it by far the most complex implementation level. A level 4 implementation may be seen as a component in an "intelligent design environment" in which designers and engineers are supported by advanced information technology. Level 4 implementation remains in the domain of basic research and development.

So, obviously, the type of implementation desired is also a factor to be taken into account when analyzing the cost of the implementation project.

Furthermore, other challenges will be encountered during the implementation phase:

- Understand the PLCS model and the DEX architecture
- Understand AP21 and EXPRESS
- Reference data
- Business domains and work processes
- Source and target models
- Mapping according to PLCS standard
- Coding
- Testing
- Data validation

11.6.1. The Ideal

Evidently, the option of full implementation of the standard would be the best possible decision. As a result, EDMS/MTF would become a complete PLM, with full coverage of a product's lifecyle and applying a ISO approved standard. The problem is that such implementation would also involve full redesign and implementation of the system especially at the database level.

Another aspect to consider would be the implementation of other standards that could complement PLCS. Since this development forces the complete design of the system, why not take the opportunity and aim for an information integration, consolidation and aiming at a sharing solution by going towards a STEP-based system. Applications protocols like AP214, PDM Schema, AP203, AP210, AP212, AP233, and others could only improve the system's capabilities.

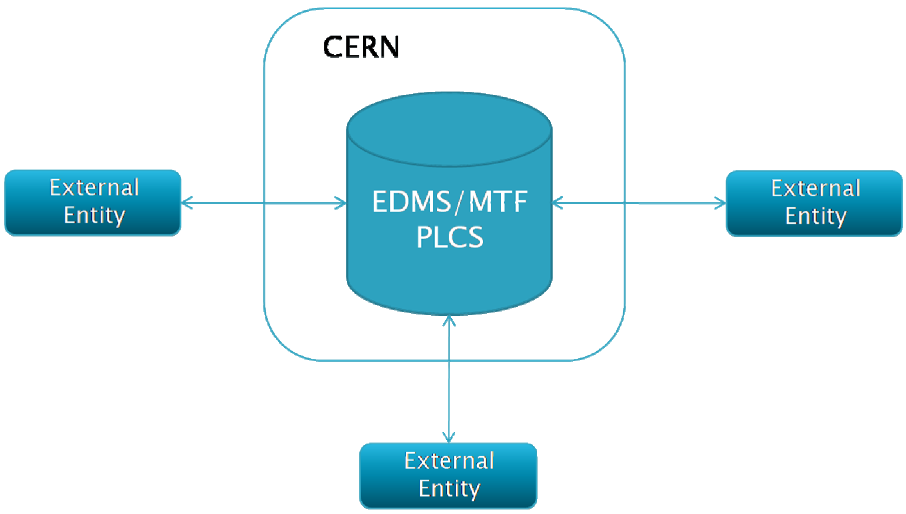


Figure 39: EDMS/MTF with AP239 database compliance

Currently, as previously mentioned, EDMS/MTF is composed of two main databases. One is managed by Axalant and contains all data concerning product lifecycle management. The other, D7i's database, contains data on equipment. Since these are commercial tools, their components cannot be manipulated as pleased. In fact, the database core structure is locked and cannot be changed. So, for EDMS/MTF, the main database structure was kept although new tables and views were added due to particularities of requirements and features developed.

To apply PLCS to EDMS/MTF as it is now, the whole database model would have to be changed, making it impracticable to use Exhalant and D7i databases PLCS compliant.

This huge restriction makes the full application of PLCS standard to the current system rather impracticable. The system wouldn't be able to use the same tools as now and the full implementation of the standard would require a big amount of resources. And once again, the final question would be if the return of such an investment would justify the project.

11.6.2. The Compromise Solution

One of the alternatives to the complete and somewhat unfeasible change of EDMS/MTF is the development of an module that uses PLCS and communicates with the current system. This module would act as a common repository between EDMS/MTF and external entities' own systems. The obvious advantage of this solution is that no modification of current system setup would have to be done. This solution is provided by a data sharing "hub" that provides a lower-cost alternative to costly maintenance of direct interfaces between multiple enterprise systems.

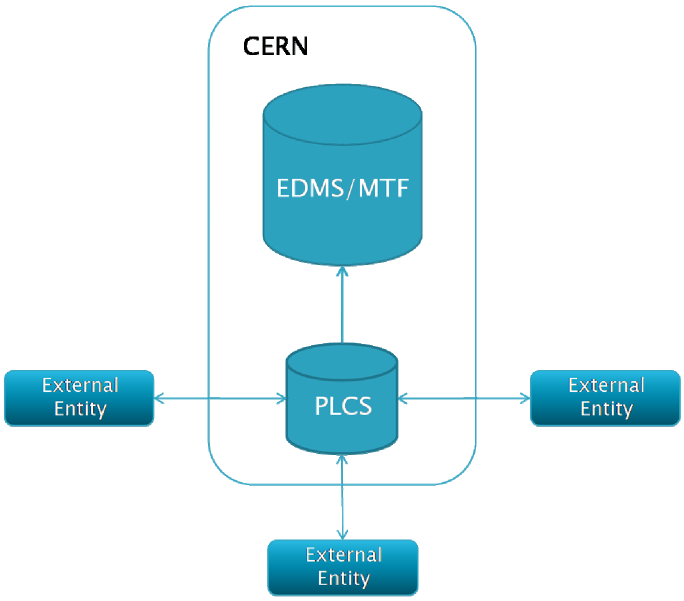


Figure 40: PLCS plug-in

The new module would work as a plug-in or extension to EDMS/MTF specially prepared for data exchange with foreign systems.

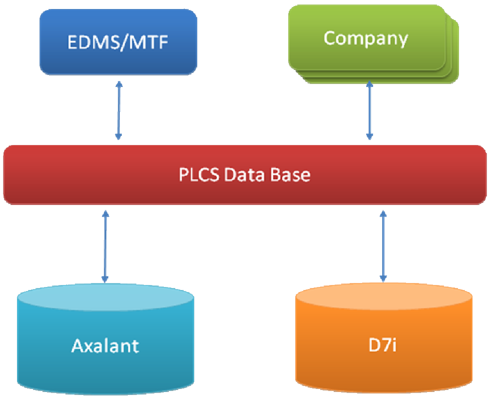


Figure 41: New EDMS/MTF generic architecture

As for the implementation of such plug-in, the range of technologies to use is very wide although there are some favorite choices due to either restrictions imposed by EDMS/MTF or due to ease of integration with existing complementary tools.

The ASE group of the TS department at CERN is responsible for EDMS/MTF development. Although the system is mainly developed under Oracle, the group always privileges"open" technologies when they represent a viable alternative to commercial tools.

Due to this approach and the fact this EDMS/MTF PLCS plug-in would be a web application, the Java platform seems to be the most appropriate for this development. There are lots of free frameworks for web application development on Java platform which would make it easy to task of finding the appropriate tools for this project. Moreover, the development team already has some experience in the development of complementary tools to EDMS/MTF using Java.

Taking the example of the Share-a-Space solution (see Chapter 10), this EDMS/MTF extension could use a multitier architecture separating the functional units of the system.

On top of the Java platform, the *Spring* framework could be used since it provides many extensions and improvements for building web-based applications. This framework also contains a collection of smaller frameworks which may be useful for the development of this plug-in, such as, *Model-view-controller* framework (isolate business logic from interface) and the *Remote Access* framework (Web Services). To apply the *MVC* model *Jakarta Struts* could also be a good option. It is an open source framework for Web Application developing allowing the creation of *Java Server Pages* (JSP). For Web Services implementation the choice falls into

Apache Axis, an open source, XML based Web service framework. It allows implementation of the SOAP server using Java and provides various utilities and APIs for generating and deploying Web service applications.

Finally, the core modules of the plug-in. The COM and Business layers are the actual responsible for the conversion of data representations and database handling.

The COM layer is responsible for handling incoming request from external sources. This layer receives the request and forwards the actions to the Business Object layer, where the actual data handling is done.

The Business Object layer is in charge of data conversion. When imports and exports occur, for example, data needs to be converted from to a particular file format such as XML or STEP file or vice-versa. It is also through this layer that communication with the database is done.

The STEP standard also defines implementation methods such as the *Standard Data Access Interface* or simply SDAI. SDAI defines an abstract Application Programming Interface (API) to work on application data according to a given data model defined in EXPRESS. SDAI itself is defined independently of a particular programming language although there are language bindings done for C, C++ and Java. The ISO10303-22 provides the guidelines for SDAI which is to be used for providing access to shared data.

The use of SDAI provides the facility to access data *as if* the physical model used for data storage is identical to the conceptual model. Additionally, since data is available through the SDAI, the implementation details of the underlying database are hidden. Similarly, the developer of a database system that provides an SDAI need not worry about the details of each individual application that will use the database. Moreover, by accessing the database through the SDAI interface, only the conceptual model is important. The implementation details of the database itself are irrelevant and thus data access is completely independent of the database implementation technology being used.

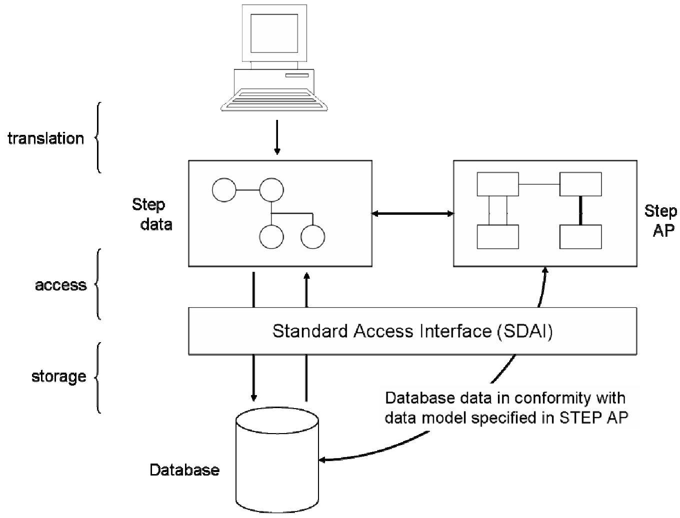


Figure 42: Data access using SDAI

For the implementation of the PLCS plug-in, there is a Java library called JSDAI [http://www.jsdai.net/]. This Java API allows reading, writing and runtime manipulation of object oriented data defined by an EXPRESS based data model. Data can be validated according to the rules defined in an EXPRESS schema. Another practical usage of JSDAI is for designing object oriented data models, where a fast start-up is required to produce prototype to validate concepts (e.g. quality of data model). JSDAI can be used with and a relational database such as MySQL or Oracle.

However, the integration of external information systems is still missing. Web Services would play an important role in the PLCS plug-in since they would be the "neutral interfaces" establishing the link to external systems.
An implementation model can be derived via transformation of the core information model into, for example, a XML Schema definition. A suitable implementation model is a XML Schema that is used as a part of a Web Service Description Language (WSDL) API. A WSDL is platform neutral and can be implemented by any systems that implement the W3C standards SOAP and WSDL. The derived implementation model can be used for exposing the API for new computer systems, but more important is that it can be used as an API wrapping up existing legacy systems.

Another plausible approach to the PLCS plug-in development would be a Service Oriented Architecture (SOA).

There are three issues that need to be covered when creating an integration platform:

- Provide an API with the description and format of the services available
- Provide the location of the services available
- Provide the connection between the systems involved

A SOA implies that services are available for use by clients not known on beforehand. From a PLM perspective a SOA can be used to make access to in-house and commercial PLM systems in an integrated environment, thus, a scalable SOA requires the adoption of new business processes with a minimum of impact on the existing PLM infrastructure. As means to deal with a heterogeneous environment the use of one (standardized) PLM interface would make each PLM system user (consumer) independent of which PLM system is used to deliver the PLM service. Further, this standardization would benefit clients that needs to access PLM information from many sources, thereby aggregating PLM information online. To further accommodate system-to-system integration there is a need to have a standardized API that defines events, which can be sent to a message bus. Hence, if PLM information can be sent (via events) each subscribing system can apply their business (context driven) logic.

In order to plug in existing PLM systems into the PLM environment they need to expose functionality according to a given standardized PLM API. The proposed PLM API is based on standardized ISO STEP PLM standards. However, in order to produce an API there needs to be implementation models generated from the ISO STEP information models (conceptual models). To make a good fit with the SOA such an implementation model should be expressed as a WSDL API.

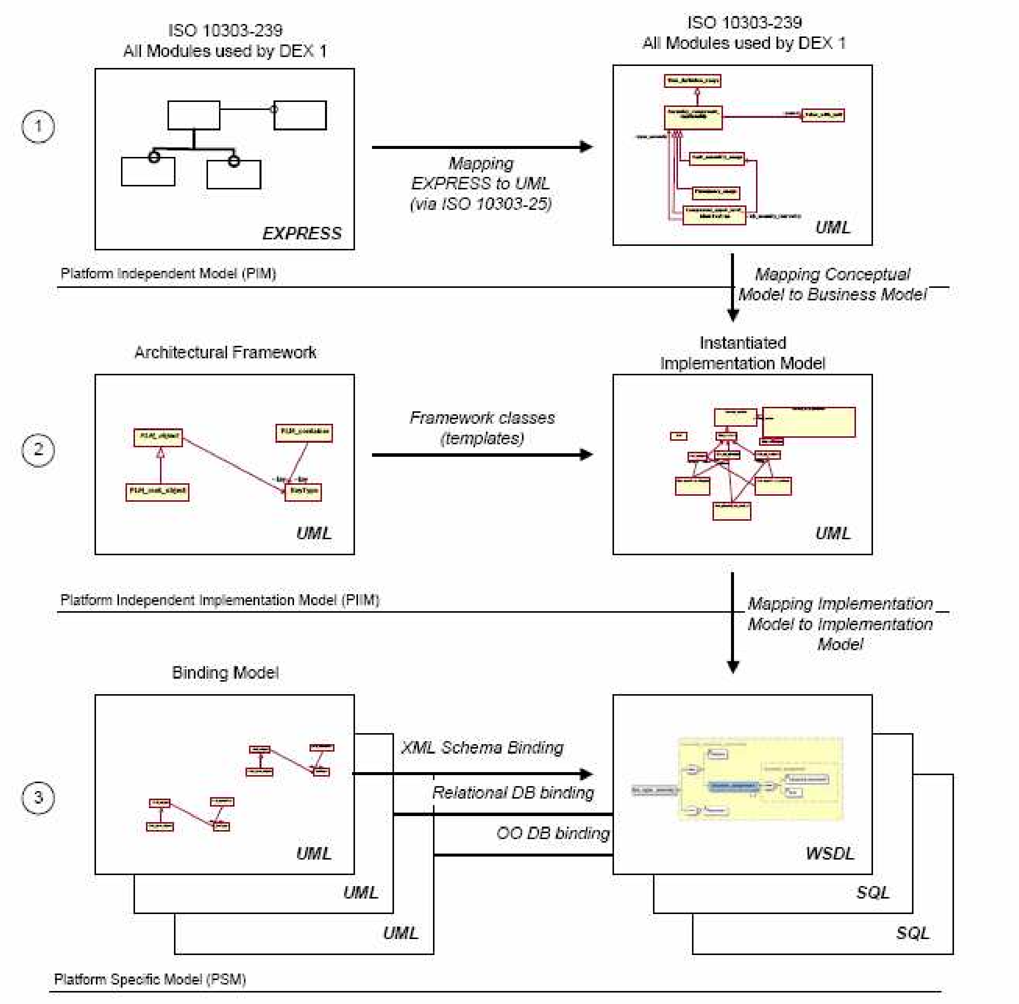


Figure 43: Deriving an implementation model [33]

The proposed architecture for the EDMS/MTF PLCS plug-in is showed in Figure 44.

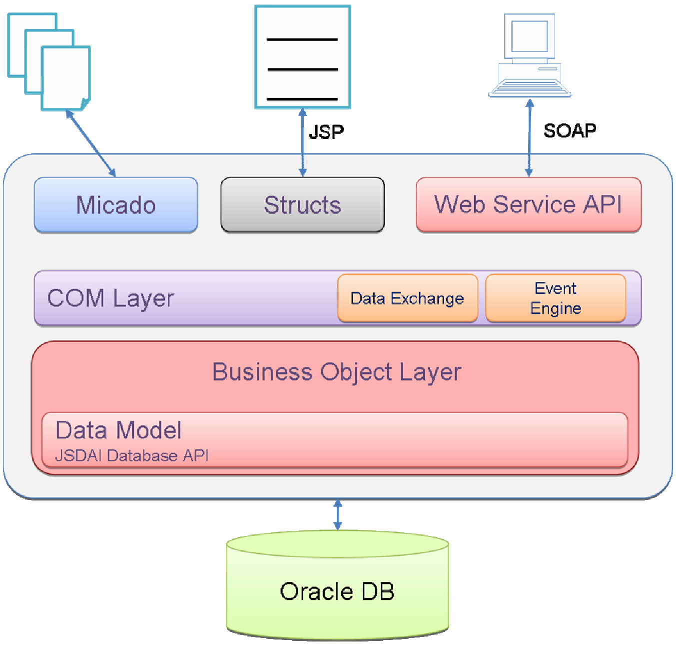


Figure 44: Proposed EDMS/MTF PLCS plug-in architecture

11.7. Data Importing with PLCS

The currently used EDMS/MTF import data mechanism would obviously require a complete re-engineering in order to be PLCS compliant. The following Figure 45 presents in a generic way the import process of data into EDMS/MTF using PLCS.

In a first stage data exported from an external system is translated into a PLCS representation. This data representation converter should ideally work with more than one data input format. Through the use of DEXs and the Reference Data Library the content in PLCS representation is converted to be imported into EDMS/MTF database.

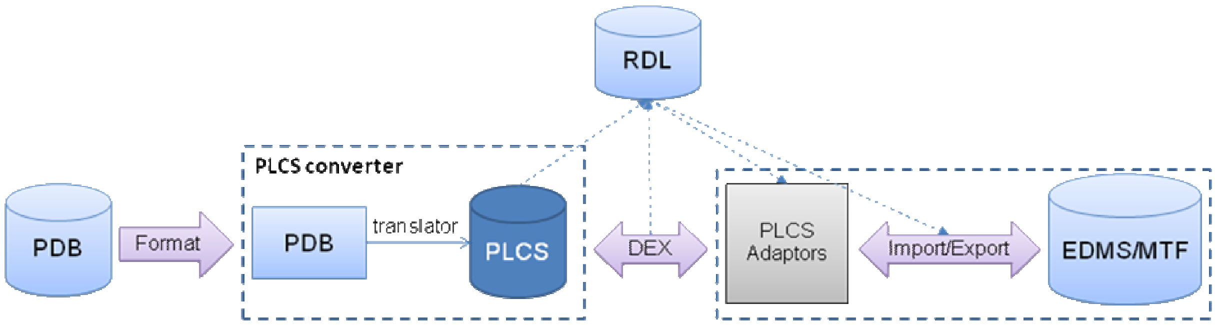


Figure 45: Importing data into EDMS/MTF using PLCS

The data is exported from external entity's database into a data export file in a particular format (e.g. P21). In the destination system the data import is performed in several stages. First, the data to be imported is validated against a particular set of rules dependent of the type of export format being used. This process aims to avoid the import of erroneous or incoherent data which may not be detected in later stages. The validated data is then converted to PLCS using defined DEXs and rules stored in a PLCS/DEX db or dictionary. The PLCS translation is then mapped into the database model and the final result validated again using a defined set of rules.

Since these series of actions aim at full automation of the import process, the set of rules has to be very complete and well structured, in order to avoid the import poor quality data which would mine the purpose of the whole system implementation.

The Figure 46 presents the PLCS conversion process in detail.

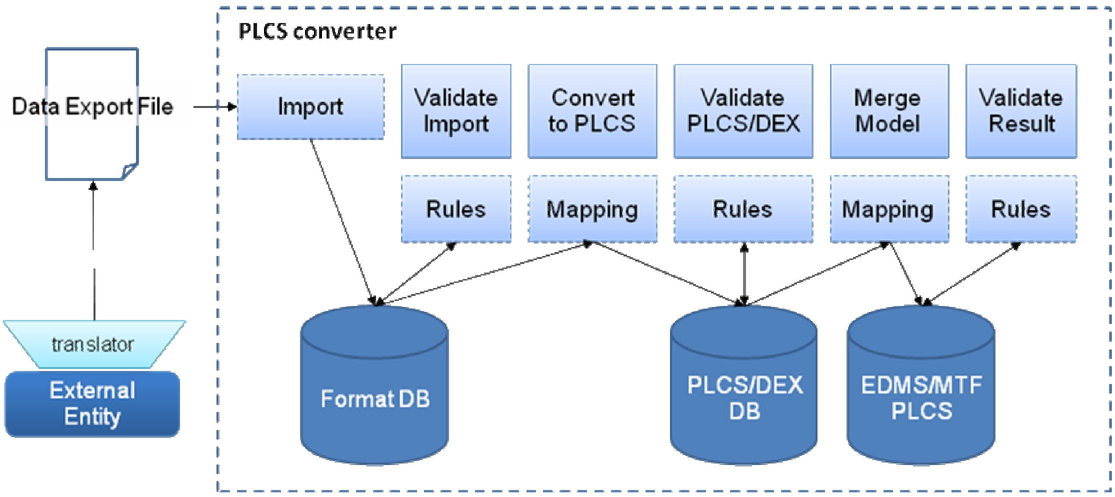


Figure 46: Imported data conversion to PLCS

11.8. Product as Individual

The previous chapter showed that the introduction of the PLCS standard in EDMS/MTF is possible. The STEP standard was developed with a modular architecture so that it could be possible to implement it in a gradual way. Consequently, applying PLCS to EDMS/MTF is more a matter of worth or added value than of practicability or feasibility.

The use of ISO10303-239 (PLCS) in a particular business context is detailed by a DEX or Data Exchange Specification. There are several DEXs available for public use that can be adapted to specific business contexts. The object of the next few sections is to try to apply a particular DEX to a EDMS/MTF feature by adapting it to the system's characteristics. The PLM used at CERN as reached a mature state and currently provides much functionality. Some of them are already covered by the public available DEXs mentioned. One of these features is the product breakdown. The purpose of this exercise is to map one EDMS/MTF features, the product breakdown, into a PLCS representation.

11.8.1. EDMS/MTF Product Breakdown

A product breakdown structure (PBS), as mentioned in section 5.2.3.1 is an exhaustive, hierarchical tree structure of components that make up an item, arranged in whole-part relationship. The tree structure is particularly useful to view the components that constitute a particular product.

The as-designed structure will contain EDMS items and projects e.g. it will be a mixture of an ABS and a PBS. In the design phase, only the definition of each type of a physical part is stored in an as-designed structure. It will be displayed in a hierarchical way, see Figure 47. The design information is stored in Exhalant. Documents can be attached to either the as-designed items or the as-designed projects. There will be links from the design nodes to the as-should- be-built positions (real equipment that is going to be installed).

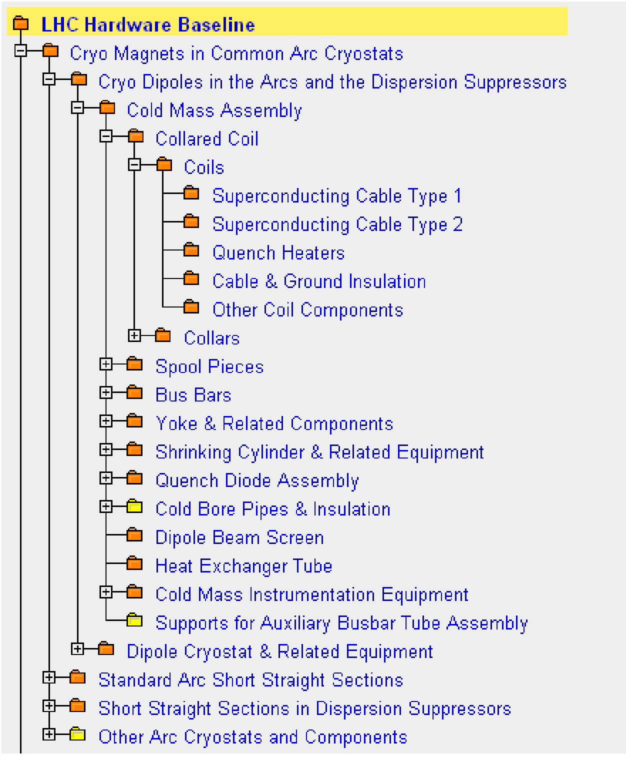


Figure 47: As-designed structure of LHC baseline [4]

Instantiations based on the type definitions will be used and reused during production to form the final as-built structure.

The BOM (Bill of Material) or ABS (Assembly Breakdown Structure), can be seen as the outcome of a project and describes how to build something. A complete BOM with attached documentation contains all necessary information to manufacture the described product. The BOM is the basis for all equipment management in the MTF and represents the design specification for the manufactured pieces of equipment.

To the equipment we attach technical characteristics and documents related to the individual components.

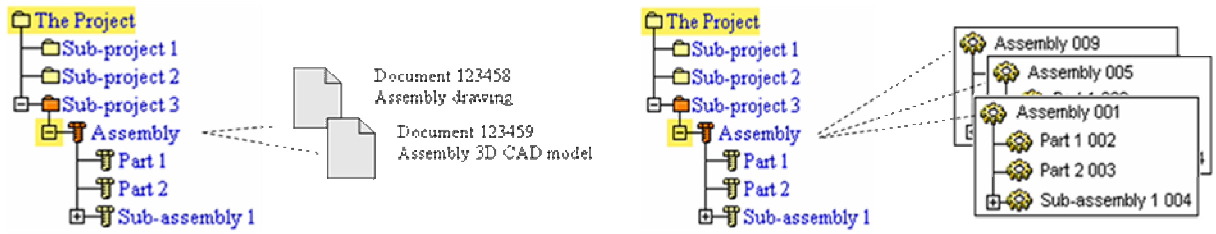


Figure 48: Assembly Breakdown Structure in EDMS/MTF [4]

The as-built structure is a description of the set of real equipment, and could be called as-installed. Data about it is kept in D7i. A real piece of equipment is either installed in as-should-be-built positions, or it is stored somewhere else waiting to be installed later. Maybe, the equipment objects in D7i will be created before they have arrived at CERN, as soon as they have been ordered. Thus the as-should-be-built structure develops into the as-built structure as arrived equipments are installed.

To establish a relation between a drawing and the manufactured equipment a link was created the documents stored in EDMS (Exhalant) and MTF (D7i).

When equipment is created in D7i, the asset will be attached to its functional position. A mechanism will trigger the creation of an item in Axalant that corresponds to the asset in D7i. The tree of functional positions will contain nodes to map both Axalant projects and items, because in EDMS the as-designed structure will contain both. The CERN equipment code (as defined in the Naming Conventions) plus position will be used as code (primary key) for the functional positions.

Assets in D7i must be linked to the corresponding design item in Axalant. Design drawings and specification documents will be stored in Axalant only once, and for each piece of equipment it is to easily find the design documents describing that particular part. Besides, there are other documents specific for each individual part, such as tests, measurements or assembly procedures. Those are also kept in Axalant, even though all data about the equipment is in D7i. These links are transparent for the user. There is a new Axalant entity (as item or project) to represent as-built equipment. The data to populate the new entity will come from D7i. So, when a Axalant user looks for data or documents, or attaches documents to nodes, the user does not know whether the information is actually stored in Axalant or in D7i.

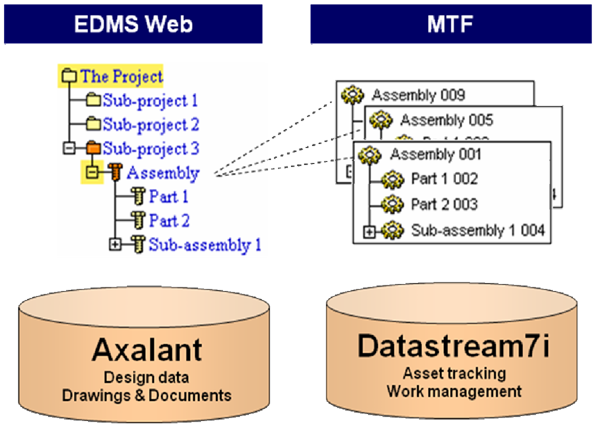


Figure 49: Data Organization in EDMS/MTF [4]

11.8.2. EDMS/MTF Versioning

As mentioned in the previous section, EDMS/MTF maintains two separated structures, as-designed and as-built. The as-designed structure contains the drawings and documents related to the manufactured equipment maintained by the as-built structure.

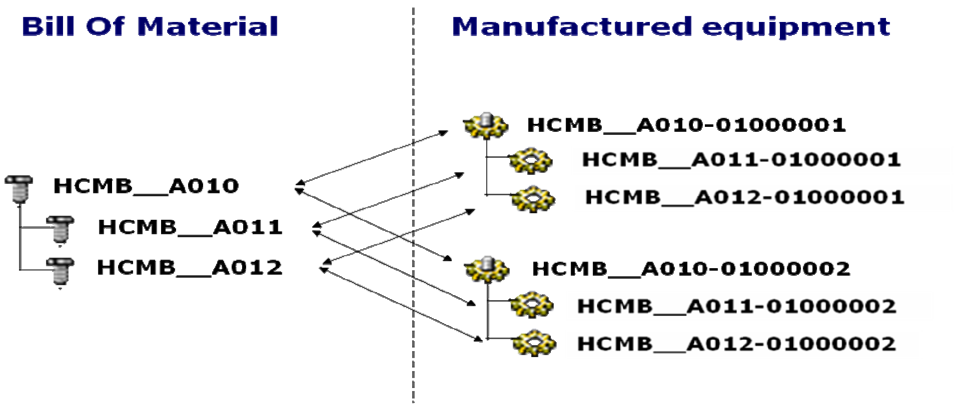


Figure 50: Connection between Bill of Material and Manufactured Equipment [4]

Designs and specifications, as any other type of document in EDMS might have several versions. It may occur that a drawing for specific equipment might become obsolete and a new version is created.

If, for example, the design of *HCMB__A012* changes and the new design is completely interchangeable with the old one – the part number is kept and only a new version of the modified ABS item is created.

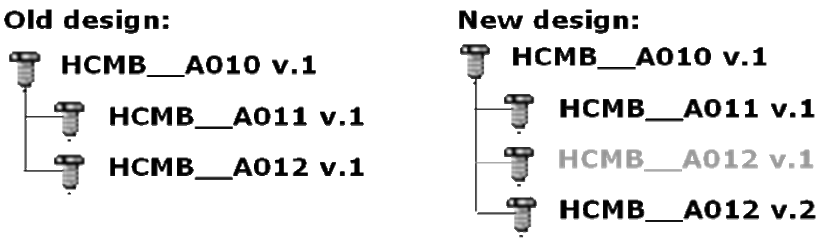


Figure 51: Versioning in designs [4]

Both the part number and the version index for the assembly *HCMB__A010* is kept this since the modification was transparent from its point of view. (Fit, form and function does not change.)

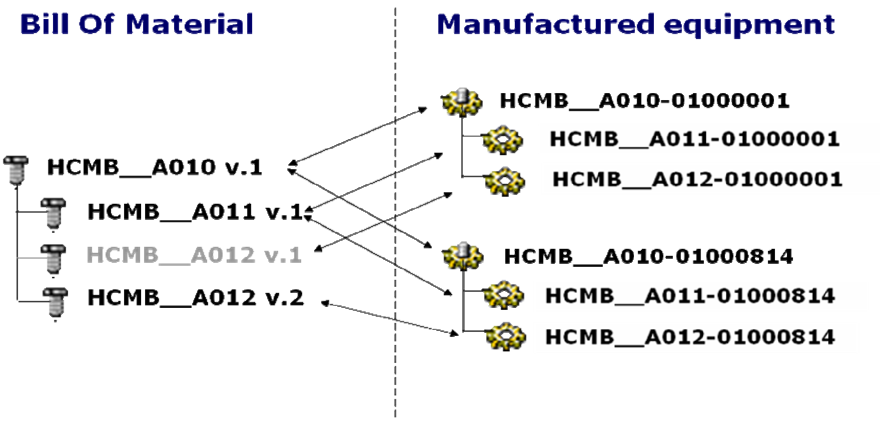


Figure 52: Connection between versions in Bill of Material and Manufactured equipment [4]

If the design of *HCMB__A012* changes and the new design is not interchangeable with the old one– a new part number and new ABS item should be created.

If the change is transparent from the assembly's point of view (having the same fit, form and function) – the part number of the assembly should be kept but we should indicate the change by creating a new version.

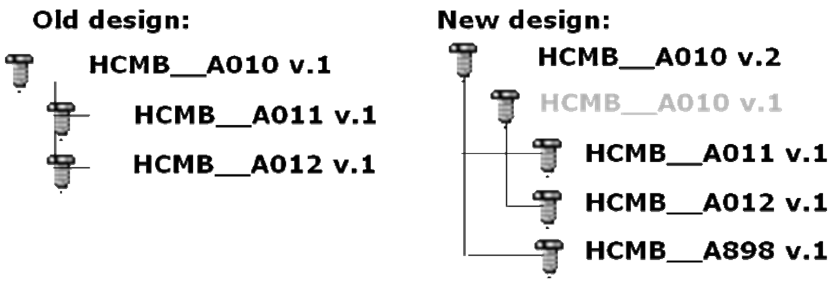


Figure 53: New version in design [4]

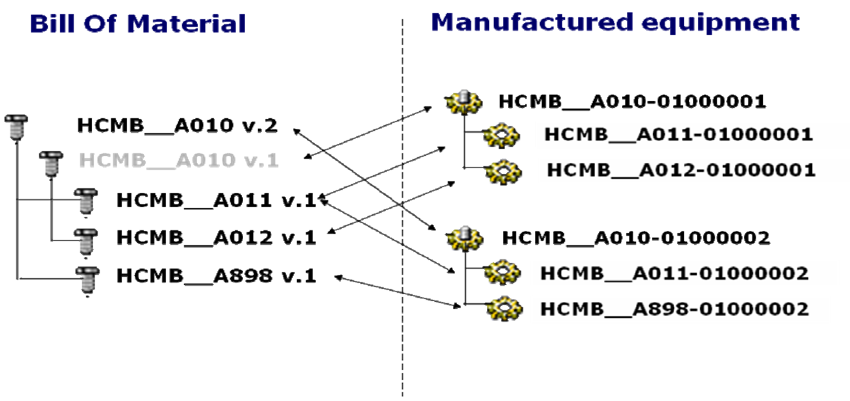


Figure 54: Connection between new versions in Bill of Material and Manufactured equipment [4]

11.9. Product Breakdown in PLCS

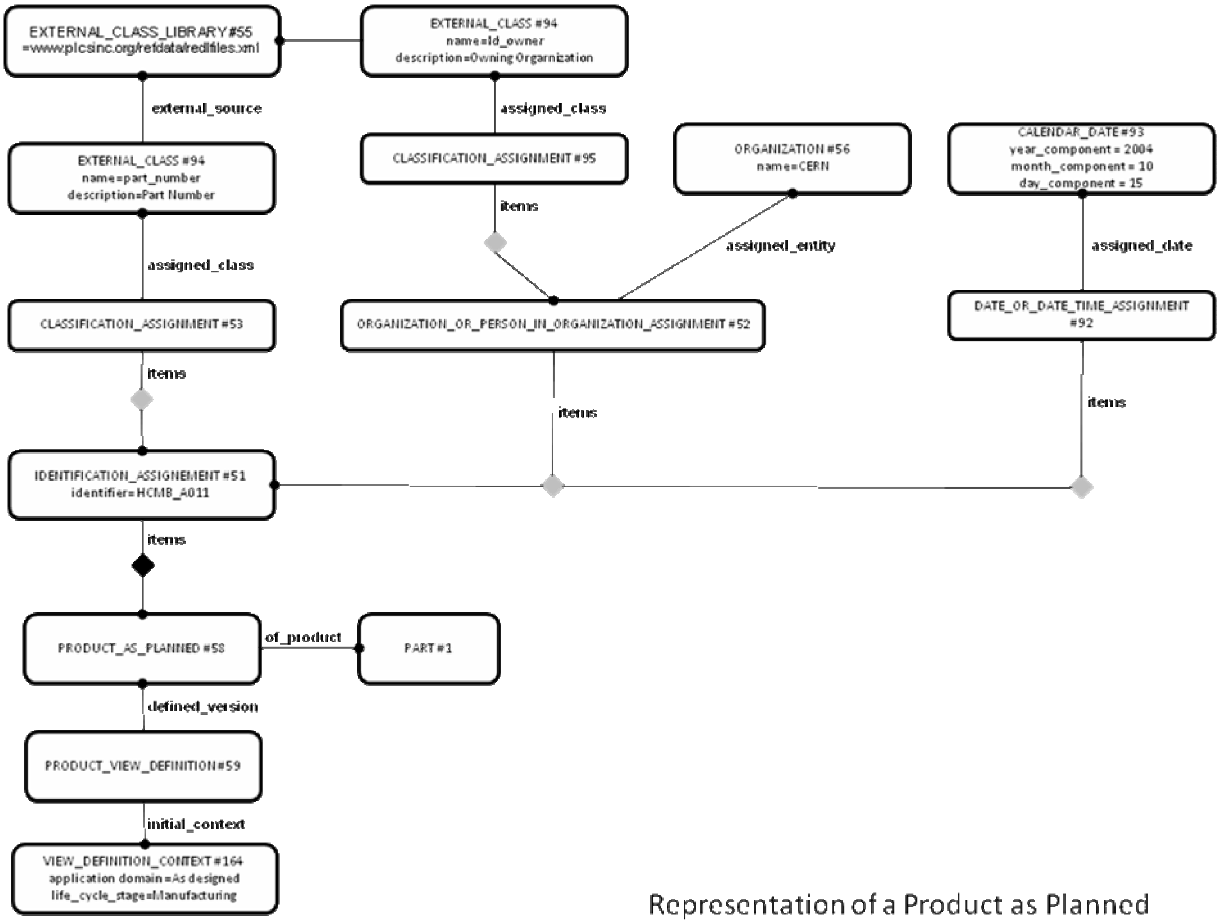
In this section, an example of a possible application of PLCS to EDMS/MTF will be given. The following will try to represent in PLCS the product breakdown and versioning of the system as presented the two previous sections.

For this, the “*product_as_individual”* PLCS DEX will be used which details how to use ISO 10303-239 "Product Life Cycle Support" (PLCS) for this particular business context.

The “*product_as_individual”* DEX allows [24]:

- the representation of a product as planned;
- the representation of a product as realized;
- the identification and classification of the product as realized
- the identification of the design that led to the product as realized;
- the representation of assembly structure of individuals;
- the representation of configuration of a product as realized;
- the representation of breakdowns of individuals;
- the characterization of product as realized structures;
- the representation and classification of properties and values of the product as realized;
- the representation of the documentation associated with a realized product;

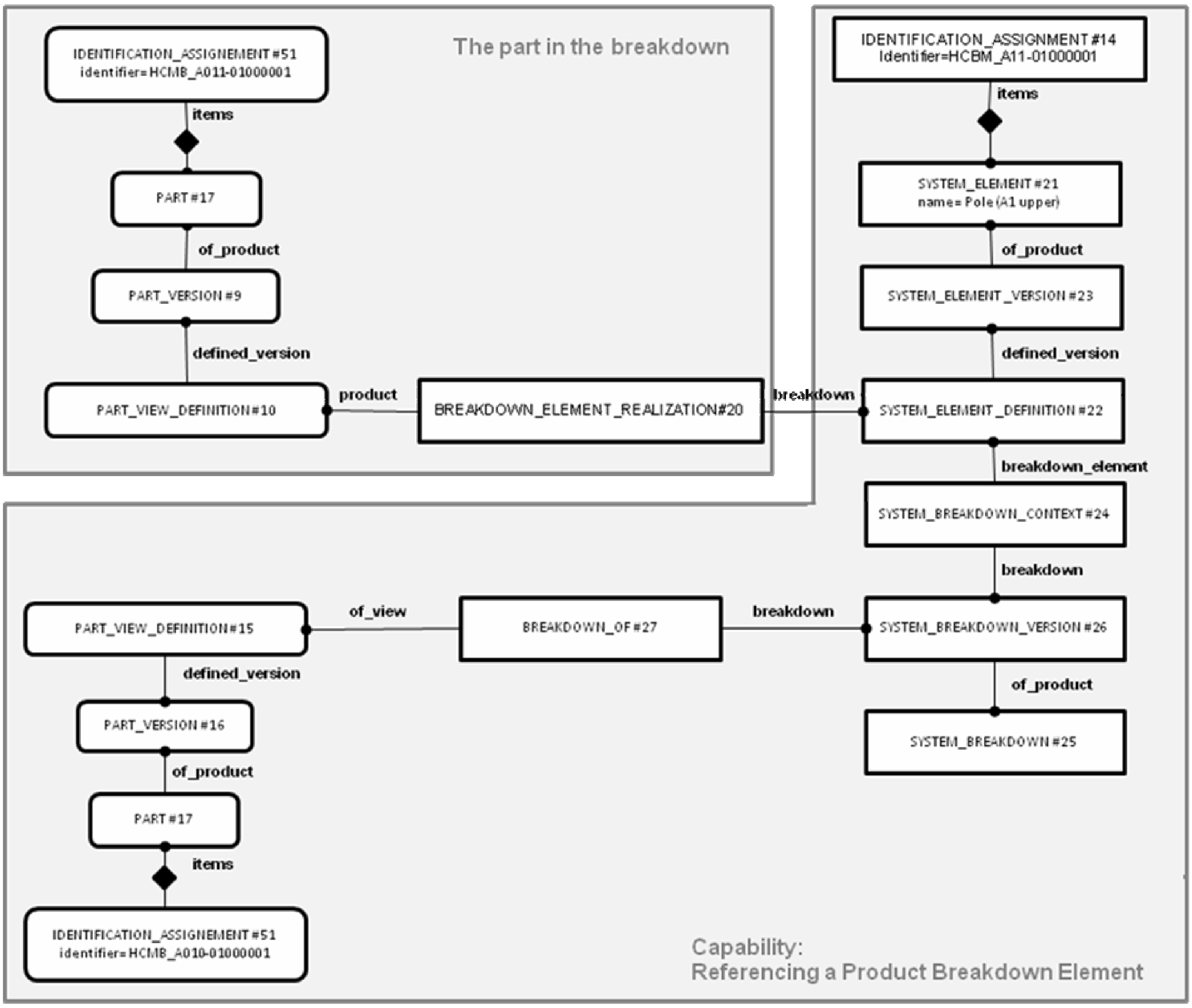
11.9.1. Representation of a product as planned



Representation of a Product as Planned

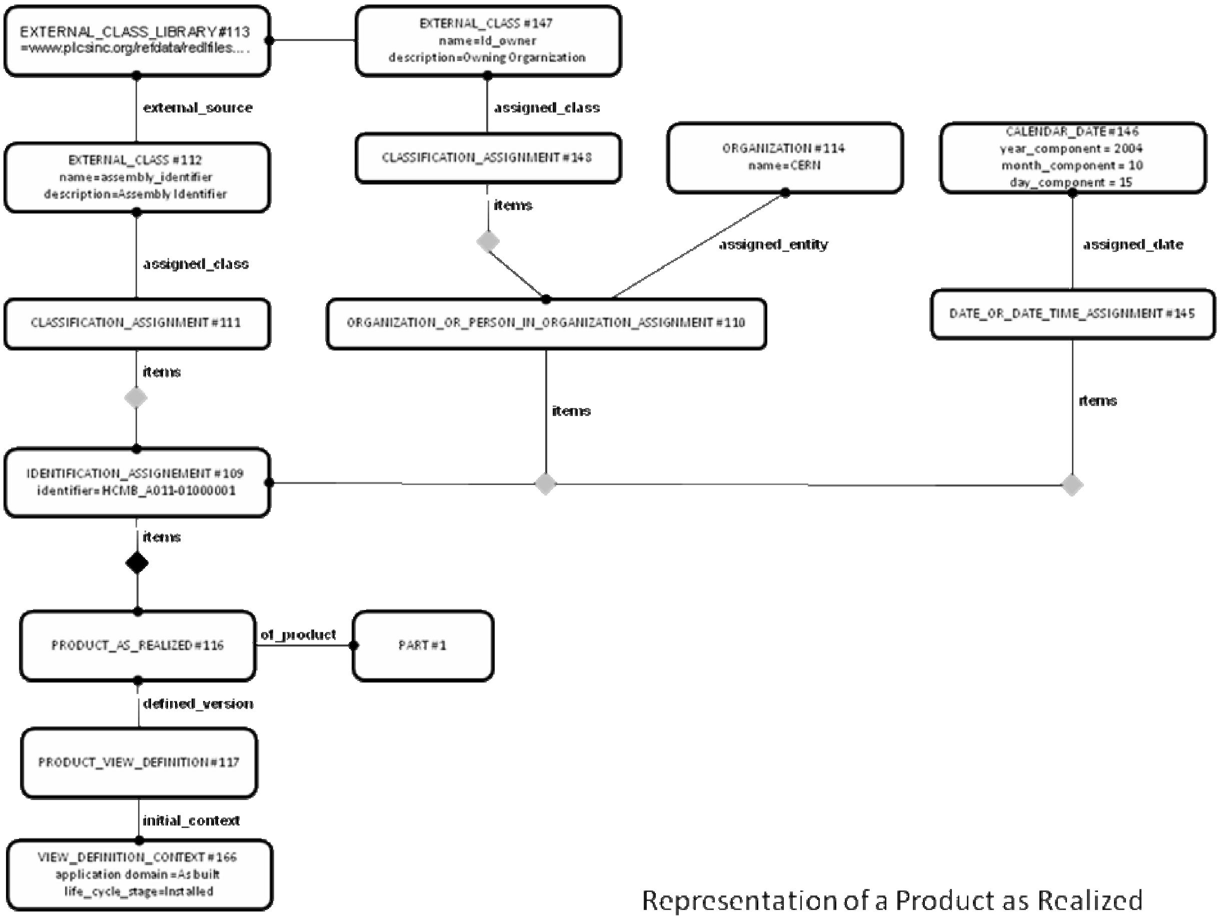
Product_as_planned represents a *as-designed* product which is identified by a part number. The product may be further identified by being associated to an organization and a data/time. All product entities should always have at least one *Product_view_definition* as it is essential to link product information (assembly structure, properties, etc) and external descriptions made through documents.

11.9.2. Referencing a Part in a Breakdown



A product breakdown element is represented in the model as a Breakdown_element or any of its subtypes (Functional_element, Physical_element, System_element or Zone_element). A Breakdown_element may be referenced by its identifier which is assigned by the capability assigning_identifiers. However, identification doesn't guarantee a unique reference in the database and so information on the context must also be provided. This context should include the Breakdown_element_version (product version in the breakdown), the Breakdown_element definition (similar to Product_view_definition) and breakdown version contained in the Breakdown_context.

11.9.3. Representation of a Product as Realized

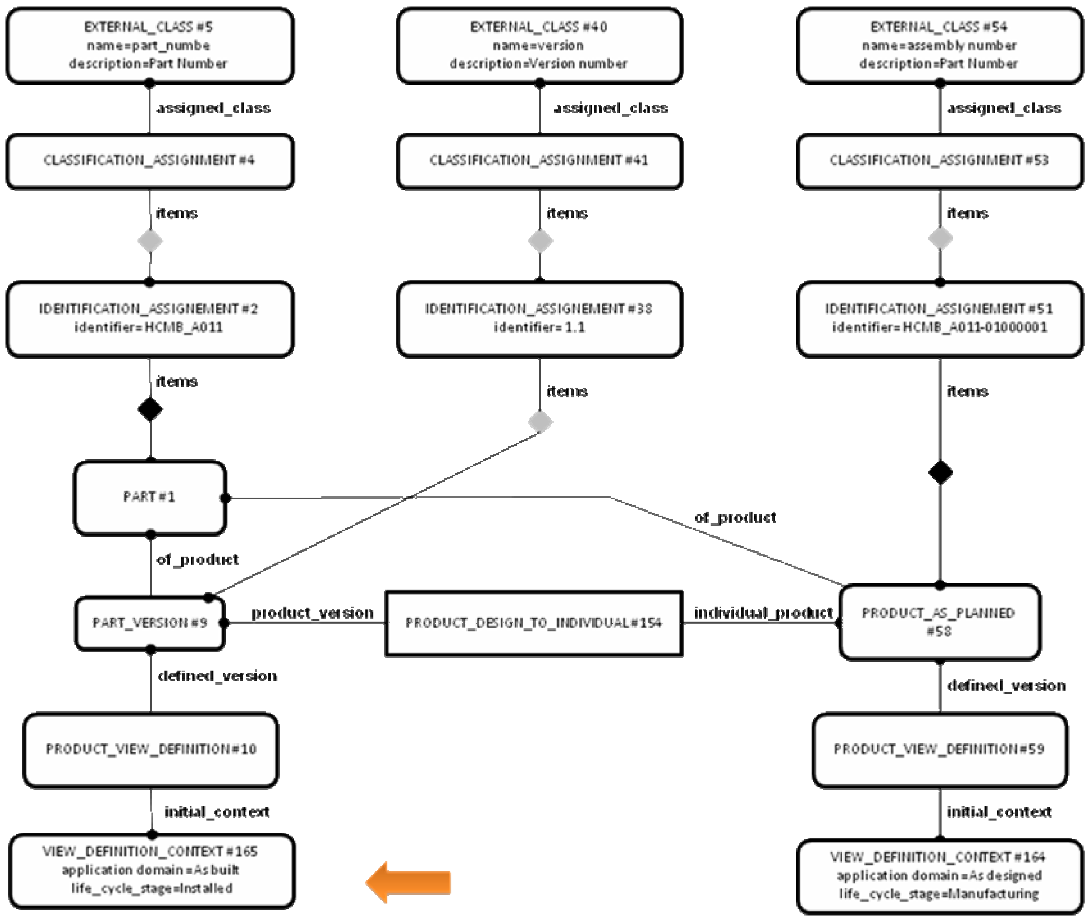


Representation of a Product as Realized

The Product_as_realize represents, in our case, a product manufactured. It is identified by a serial number which is represented by assigning an identifier classified as "Serial_identification_code". To further identify the product the organization to which it belongs and/or date/time may be associated. The Product_view_definition entity represents the identification of a particular view on a version of the product base relevant for the requirements of particular life-cycle stages and application domains. Once again, all Product_version entities should be linked to a Product_view_definition.

If the product is a result of the product as planned then it maybe related to a version of the Product_as_planned by the relation entity Product_planned_to_realized.

11.9.4. Relationship of a Product as Planned to a Version of the Design



Relationship of a Product as Planned to a Version of the Design

12. Conclusions

STEP and PLCS

Recent changes in business are influencing the companies' view on how to approach partnerships in projects with considerable dimensions. The traditional point-to-point data exchanges are no longer a viable option in a market where agility and access to information are key factors for success. Companies need to get the newest and most complete set of information to make the best decisions.

In complex projects involving partnerships between different companies, information exchange is mandatory. The nature of these exchanges is directly dependent on the type of collaboration. Communication may only be done between a company and its suppliers or, on the other hand, all entities need to communicate and shared information between each other. As an answer to this need for information shared the *Extended Enterprise* concept was created. The *Extended* or *Virtual Enterprise* advocates the need to share information using a shared repository to which all parties involved in the partnership have access. This solution provides a neutral information set where the master data for the collaboration is managed. Like this, partners can agree that the valid information for the partnership is that in the repository regardless of what is managed on their own internal systems. The result is a common, consolidated and agreed information set available in a shared repository available to every partner.

In order to implement such shared repository a common data representation format is needed. The industry came up in its own solution for this problem by introducing the STEP standard and, more recently, the PLCS application protocol. The STEP has been around for many years providing a neutral data representation for data exchange. It has been widely used over the years mainly for exchange of CAD drawings. The standard though, was limited to the design phase, not supporting any of the other phases of a product life cycle. To bridge this gap in the STEP standard, an application protocol was developed that provides full coverage of the life cycle. The PLCS provides a mechanism to maintain information needed to support complex assets over its life cycle. It can then be seen as a desirable extension to the ISO 10303, which addresses the requirements of product support from concept to disposal.

PLCS offers data independence from processes, IT systems, and even data formats. This allows information to be de-coupled from internal business processes making it available for every partner in a neutral format or allowing its re-use outside the partnership.

The key concept of PLCS is the creation of a "single source of truth" – concerning assured product and support information - for use across the enterprise. This single source or shared repository allows partners to share the same view on the current information. This means each enterprise involved will have access to the same information set which is at the same time the latest version.

The consolidation of data from different sources into a single environment enables collaborators to always know where the valid piece of information resides which is not obvious when not using a shared repository.

Other benefits of PLCS are automation and standardization. Automation not only decreases error occurrence of manual entry of data but also improves concurrency since frequent data updates can be made with minimal effort. The standardization of the data exchange through PLCS is an invitation for data re-use and facilitates upgrades or replacements without affecting other systems.

Unfortunately PLCS is not easy to implement. The STEP standard itself is quite complex and the AP239 demands an extra effort to be applied in a system. Implementers need to be well acquainted with the standard even before considering going ahead with the project. So, there is this overhead needed.

In addition, the implementation itself is not particularly easy. The first step is to develop the DEXs. There are DEXs available although they are generic, meaning that some “tailoring” has to be down to adapt them to a particular business context. In this phase the data model also needs to be defined. As in every project this is a crucial stage of a system development a demands extra consideration since developers are dealing with “new technology”.

In conclusion, a project for the implementation of PLCS must be carefully studied. It demands great commitment of the company since it is complex and may take a while to produce visible results. Consequently, the balance between the investment made and the return expected must be carefully analyzed. In a very simplistic view, PLCS implementation may only be worth when many companies are involved in a long-term partnership. This kind of projects usually involves exchange of large volumes of data, through a long period of time and so, the advantages of PLCS may be important.

PLCS at CERN

The LHC project at CERN is a very good example of how a company may profit from the implementation of PLCS. This project is a partnership with hundreds of companies through all the construct of the LHC. Of course, some partnerships lasted longer than others and some have more importance. The Cryodiple assembly provided a good example of how PLCS could be useful. It contained all ingredients to justify appealing to the use of the standard.

As in any other PLM, PLCS can be used with EDMS/MTF. The most viable option would be to keep the current implementation of the EDMS/MTF service unaltered, but providing a PLCS module as an extension. The application of the standard could then be gradual, not disturbing the normal operation of EDMS/MTF.

Since the LHC project is reaching its operation phase, most partnerships are reaching their deadline. It is then difficult to justify the implementation of PLCS in EDMS/MTF at this point.

Although the implementation could still be important, since there a still collaborations going on and some may still be started, no “big projects” are foreseen in the near future.

The future of PLCS is still quite uncertain. Although it has been around for some years now it has not been widely used. In fact, most of the critics derive from its slow development, especially when compared with e-commerce which standards are being developed much more quickly.

However, there are already some known successful implementations of PLCS. The standard has been particularly thriving in the air and defense business. The Norwegian Frigate, BAE Systems, UK RAF, US DoD, Boeing and Rolls-Royce are known to have their own implementations. Still, Eurostep’s Share-a-Space is the only famous commercial implantation of PLCS, which may be a symptom of the lack of adoption of the standard by the industry.

References

[1] – CERN, http://cern.ch

[2] – EDMS, http://edms.cern.ch

[3] – MTF, http://mtf.cern.ch

[4] – The CERN EDMS for engineers, EDMS doc. 401800

[5] – Agile PLM, http://www.agile.com/plm/index.asp

[6] – D7i, http://www.datastream.net/English/Default.aspx

[7] – Peter Engström, Structure Management throughout the Product Lifecycle using a PDM System

[8] – Product Data Management (PDM), John Stark

[9] – Data modeling in a PDM system to support detector design and production specifications, Joakim Sundkvist

[10] – Product Breakdown Structure Creation, ADD AB

[11] – Pre-Study and Definition of Data Collection for Manufacturing and Test Data in the CERN EDMS, Magnus Nerman

[12] – White Paper: Sharing Product Data Through Life in the Extended Enterprise – the Share-A-Space solution, Dr Mathias Johansson, Eurostep Commercial Solutions AB, Sweden

[13] – What is Extended Enterprise, http://jpfarrell.blogspot.com/2008/04/extended-enterprise.html, A Management Consultant@Large

[14] – Supply Chain, http://en.wikipedia.org/wiki/Supply_chain, Wikipedia

[15] – ISO, ISO 10303:1994 - Industrial Automation Systems and Integration - Product Data Representation and Exchange. (The ISO web site is at http://www.iso.ch/cate/cat.html - search on 10303 to find a list of parts of the standard).

[16] – ISO 10303, http://en.wikipedia.org/wiki/ISO_10303, Wikipedia

[17] – STEP AP203 data exchange study, Alan L. Clark *et all*

[18] – Manufacturing Interoperability: Problems, Costs, Mitigation, James E. Fowler, National Institute of Standards and Technology

[19] – STEP for Data Management, Exchange and Sharing, John Flowler

[20] – STEP Standard, http://www.steptools.com/library/standard/, STEP Tools

[21] – ISO, ISO 10303-239, Product Life Cycle Support
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38310

[22] – PLCS FAQ, OASIS

[23] – Re-Engineering the Aftermarket with PLCS, Eurostep and John Stark Associates

[24] – PLCS Resources, www.plcs-resources.org, Eurostep Group

[25] – Eurostep, http://www.eurostep.com/

[26] – Share-A-Space, http://www.share-a-space.com/index.asp

[27] – Eurostep's Share-A-space Product Lifecycle Collaboration through Information Integration, CIMdata

[28] – Engineering Project Support in a Distributed Design Process using WWW and Internet, David Widegren

[29] – The CERN EDMS – not just for building the LHC, David Widegren, EDMS doc. 846988

[30] – PLM at CERN .- A True Challenge, David Widegren

[31] – EAB2 Getting Started, Marcin Sobieszek, EDMS doc. 837935

[32] –MICADO Software Documentation, Bertrand Rousseau, EDMS doc. 827122

[33] – Federated Through-Life Support, Enabling Online Integration of Systems within the PLM Domain, Jonas Rosén

Appendix I

The following list contains the STEP Application Protocols (AP) as of June 2004.

Part 201 Explicit Drafting
Part 202 Associative Drafting
Part 203 Configuration Controlled Design
Part 204 Mechanical Design Using Boundary Representation
Part 205 Mechanical Design Using Surface Representation
Part 206 Mechanical Design Using Wireframe Representation
Part 207 Sheet Metal Dies and Blocks
Part 208 Life Cycle Product Change Process
Part 209 Design Through Analysis of Composite and Metallic Structures
Part 210 Electronic Printed Circuit Assembly, Design and Manufacturing
Part 211 Electronics Test Diagnostics and Remanufacture
Part 212 Electrotechnical Plants
Part 213 Numerical Control Process Plans for Machined Parts
Part 214 Core Data for Automotive Mechanical Design Processes
Part 215 Ship Arrangement
Part 216 Ship Molded Forms
Part 217 Ship Piping
Part 218 Ship Structures
Part 219 Dimensional Inspection Process Planning for CMMs
Part 220 Printed Circuit Assembly Manufacturing Planning
Part 221 Functional Data and Schematic Representation for Process Plans
Part 222 Design Engineering to Manufacturing for Composite Structures
Part 223 Exchange of Design and Manufacturing DPD for Composites
Part 224 Mechanical Product Definition for Process Planning
Part 225 Structural Building Elements Using Explicit Shape Rep
Part 226 Shipbuilding Mechanical Systems
Part 227 Plant Spatial Configuration
Part 228 Building Services
Part 229 Design and Manufacturing Information for Forged Parts
Part 230 Building Structure frame steelwork

Part 231 Process Engineering Data
Part 232 Technical Data Packaging
Part 233 Systems Engineering Data Representation
Part 234 Ship Operational logs, records and messages
Part 235 Materials Information for products
Part 236 Furniture product and project
Part 237 Computational Fluid Dynamics
Part 238 Integrated CNC Machining
Part 239 Product Life Cycle Support
Part 240 Process Planning

Appendix II

Extracted from Quality Assurance Procedures in a Large Engineering Project Aided by EDMS: a Study, by Peter Nitter