DISTRIBUTED ENGINEERING IN CONSTRUCTION: FINDINGS FROM THE IMS GLOBEMEN PROJECT

SUBMITTED: October 2001
REVISED: November 2001
PUBLISHED: December 2001 at http://itcon.org/2001/10/

Abdul S. Kazi, Research Scientist
Technical Research Centre of Finland
email: Sami.Kazi@vtt.fi http://cic.vtt.fi/kazi

Matti Hannus, Chief Research Scientist
Technical Research Centre of Finland
email: Matti.Hannus@vtt.fi http://cic.vtt.fi/hannus

Jarmo Laitinen, Development Manager
YIT Corporation, Finland
email: Jarmo.Laitinen@yit.fi http://www.yit.fi

Olli Nummelin, Development Engineer
YIT Corporation, Finland
email: Olli.Nummelin@yit.fi http://www.yit.fi

SUMMARY*: Inter-enterprise collaboration to deliver a one-of-a-kind product is gaining momentum with the emergence of new information and communications technologies (ICT) to support information exchange and collaborative work amongst distinct geographically dispersed entities. Typically, the modus operandi of such collaborations is that of the virtual enterprise. While relatively new to some industries, this has been a common mode of operation for the construction industry for long. ICT support for the construction industry to support its ways and modes of operation has however been lacking. While organisation specific proprietary tools do exist, those supporting inter-enterprise collaboration are not up to a similar par.*

This paper presents the IMS GLOBEMEN (Global Engineering and Manufacturing in Enterprise Networks) and some of its core findings and developments related to reference architecture for virtual enterprises and primarily distributed engineering in construction. After an overall presentation of the GLOBEMEN project, and some basics of inter-enterprise collaboration, the paper presents the main elements of the Virtual Enterprise Reference Architecture and Methodology (VERAM) through identification of its main components and their constituents. This is followed by an exploration of distributed engineering. In this section the focus is on building construction where some very high level use cases are presented. The section is concluded with a presentation of an ICT architecture at both generic and specific levels for distributed engineering at large and product model based distributed engineering in construction in particular respectively.

KEYWORDS*: inter-enterprise collaboration, virtual enterprise, distributed engineering, construction.*

1. INTRODUCTION

Delivery of unique one-of-a-kind products and services are becoming a norm rather than an exception in many industry sectors. Consequently, project oriented modes of operation are becoming common in many industrial sectors as an attempt to provide both increased flexibility and agility to operations. Consequently, the provision of ICT support for dynamic, geographically and organisationally dispersed project teams, Virtual Enterprises (VE) has been a key area of both research and development.

Currently available commercial ICT systems are primarily focused towards internal use within organisations and their associated supply chains. The requirements and desired functionalities available today do not appropriately address or cater towards the needs of dynamic virtual enterprises, which are organisation independent and where formal lines of control and management may be hard to establish. To address this need, the Globemen project was harnessed. It focuses on ICT support for dynamic virtual enterprises in one-of-a-kind industries. This includes the definition and elaboration of a *Virtual Enterprise Reference Architecture and Methodology* in addition to a systemic analysis of different industrial sectors leading to some domain specific prototypes.

This paper focuses on some of the results to date of the Globemen project. After a general overview of the project, we address some of the key issues related to inter-enterprise communications. This sets the foundation for some salient characteristics of a virtual enterprise. We briefly discuss some core concepts and functions of a typical VE and then continue on to the *Virtual Enterprise Reference Architecture and Methodology (VERAM)* as it currently stands. The paper continues with a focus on one of the main domains considered in Globemen, Distributed Engineering. Here too, the discussion is centred on product model based distributed engineering in construction. This is elaborated through some high-level use cases and complemented with a presentation of the foreseen ICT architecture.

2. GLOBEMEN

GLOBEMEN (Global Engineering and Manufacturing in Enterprise Networks) was initialised in order to define and harmonise ICT support requirements in various one-of-a-kind industries operating in various cultural environments. By combining the views and requirements of various industries the project aims to guide and encourage the industry and IT vendors to develop and adopt improved IT infrastructures. Furthermore, GLOBEMEN aims to demonstrate functionalities, which offer attractive market opportunities to IT vendors for product development to satisfy the needs of various industries worldwide.

2.1. Objectives

The aim of GLOBEMEN is to create IT infrastructures and related models, methods and tools to support globally distributed and dynamically networked operations in one-of-a-kind industries. The focus is on inter-enterprise integration and collaboration in the three main facets of global manufacturing: sales and services, inter-enterprise delivery process management, and distributed engineering (see Fig. 1).

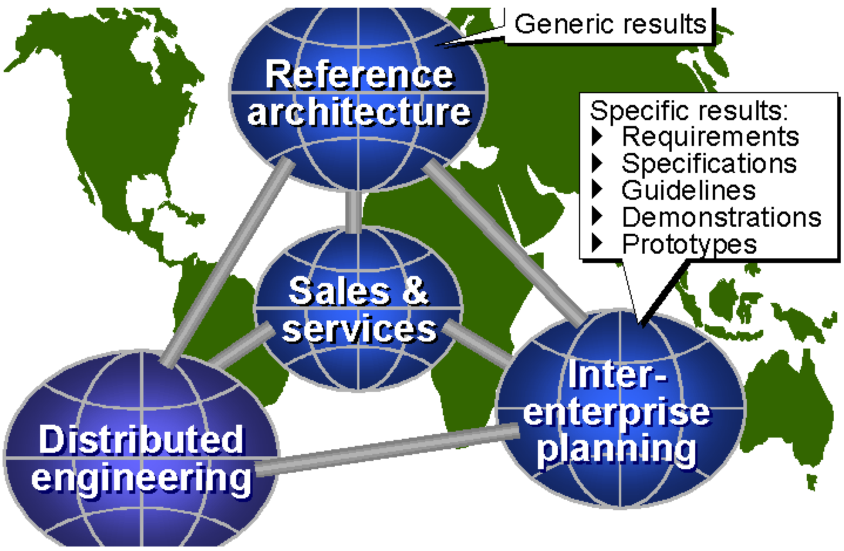

FIG. 1: Main facets of global engineering and manufacturing considered in GLOBEMEN

It has been advocated that despite their apparent differences, various one-of-a-kind industries can be supported by fundamentally similar kinds of ICT tools (Hannus and Kazi, 2000). The project aims to find commonalities between various industrial domains and cultural environments and to suggest an integrating reference architecture. Different application systems can be developed based on the generic architecture. The potential is to magnify the global market for ICT solution providers while offering enhanced support to the industry.

2.2. Description of Work

The development work within each of the addressed facets of global engineering and manufacturing includes: definition of architecture, specification of required methods and tools, preparation of VME (virtual manufacturing enterprise) guidelines to support industrial implementation and deployment, demonstration of proposed architecture and implementation of industrial prototypes. All workpackages, their main tasks and overall inter-dependencies are illustrated in Fig. 2.

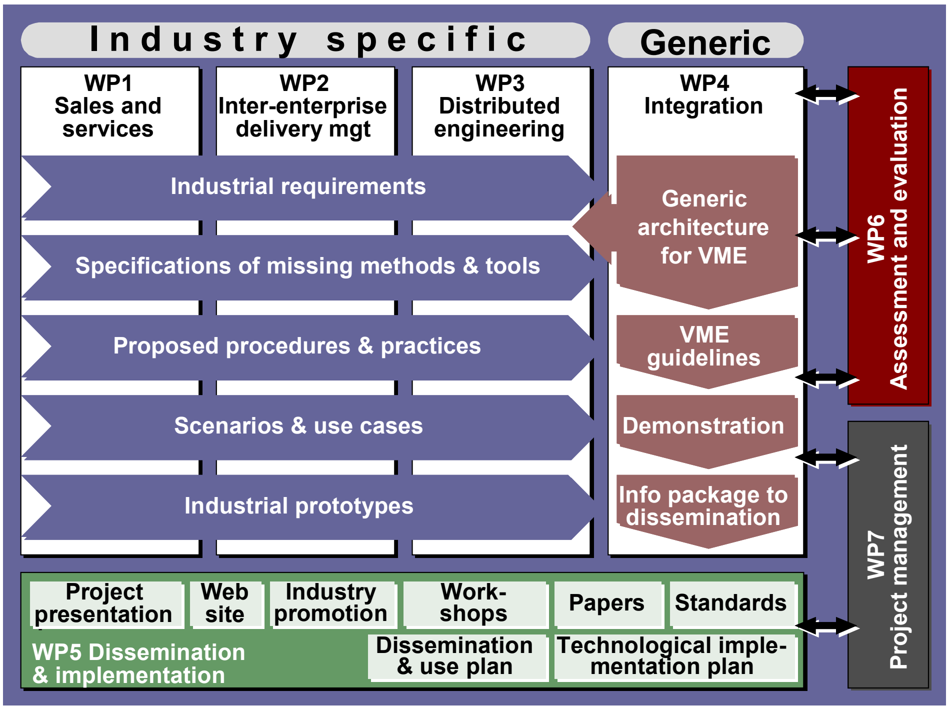

FIG. 2: Main workpackages, tasks, and inter-dependencies

2.3. Industrial Relevance.

The industrial relevance of the undertakings within GLOBEMEN and its contribution to the industry is envisioned as follows:

- Radical decrease (50% or more) in lead-time of all phases in manufacturing is a must in the manufacturing industries. Efficient enterprise networking and distribution of functions is necessary for lead-time reductions of this magnitude.
- Businesses today are becoming more dynamic and multicultural. The relationships between companies in networks are changing with increasing speed.
- Dynamic global networking cannot be efficient without tools allowing true concurrency for all partners in the network. The ultimate aim of Globemen is to equip the initially the project partners and eventually the industry at large with these tools.
- There is expected to be a major direct economic impact to the industrial end-user partners.
- Different application systems can be developed based on the generic architecture developed in the project. This will magnify the global market potential for IT solution providers.

3. INTER-ENTERPRISE COLLABORATION

Current project configurations and associated modus operandi for delivery of one-of-a-kind products and services are changing the way organisations function and collaborate with each other. New trends are emerging and priorities are consequently changing (Table 1). As such, there is a distinct move towards inter-enterprise collaboration. Some differences between intra-enterprise and inter-enterprise setting are shown in (Table 2). Some of the functional requirements for working in such environments have been presented by Kazi and Hannus (2000) elsewhere.

TABLE 1: Changing trends and priorities for inter-enterprise collaboration

From	To
Centralised planning	Transparency of information
Enterprise resource planning	Inter-enterprise coordination
Document management	Object management
In-house operative systems	Inter-enterprise collaborative systems
Supply chain management	Demand change management
Workflow management	Groupwork support
Scheduling	Schedule synchronisation
Management information systems	Decision and negotiation support
Reporting	Forecasting and coordination
Electronic commerce	Elimination of ordering
Access control	Knowledge sharing
Integrated systems	Flexible interfaces

TABLE 2: From intra- to inter-enterprise collaboration

	Intra-enterprise	Inter-enterprise
Contractual coverage	No	Yes
Legal responsibility	No	Yes
Technology	Proprietary	Industry standard
Control of ICT solutions	High	Low/none
Set-up time	Long (~months … years)	Very short (~days)
Training	Yes	No
Application user interface	Yes (~"bells and whistles")	Hidden (~"Save as …")
Integration	Integrated applications and databases	Culture, ontology, standards, data warehouses, etc.
Target users	Own staff	Unknown future partners
Coordination	Resource and workflow management	Deliverable management
Information updating	Synchronous	Asynchronous

The findings from Table 1 and Table 2 clearly point towards the operational paradigm of the virtual enterprise. Its salient characteristics, core concepts and core functions are described in Section 3.1.

3.1. The Virtual Enterprise

The *virtual enterprise* (VE) and its different synonyms (e.g. *Smart Organisation* as referred to by Filos and Banahan, 2001) have by far become one of the most used terms in current research undertakings (Browne, et al, 1994), (Charbuck and Young, 1992), (Rabelo and Camarinha-Matos, 1996), (Walton and Whicker, 1996). While in simple terms, a VE may be identified as a temporary collaboration to exploit a business opportunity (with each business transaction being performed by a collaborating legal entity), many authors have elaborated it further as follows:

- "A Virtual Enterprise is a temporary aggregation of competencies and resources to exploit a business opportunity sharing capabilities, capacities, risk and profit to fulfil specific customer demands. Formation of the VE materialises through configuration of the core competencies available in the network and possibly through inclusion of additional, required competencies provided by non-network participants. Though being comprised by competencies from various partners, the VE appears to the customer as one, unified, and attuned enterprise. Hence its virtual nature.

Accordingly, the business processes are not carried out by a single enterprise, rather every enterprise is just a node in the VE that adds some value to the product chain." (Afsarmanesh et al., 1997)

- "a Virtual Enterprise is a temporary consortium or alliance of companies formed to share costs and skills and exploit fast-changing market opportunities" (NIIIP, 1996)
- "Virtual Corporation is a temporary network of independent companies - suppliers, customers, even rivals - linked by information technology (IT) to share skills, costs and access to one another's markets. It will have neither central office nor organisation chart. It will have no hierarchy, no vertical integration" (Byrne, 1993)
- "the Virtual Enterprise consists of a series of co-operating 'nodes' of core competence which form into a supply chain in order to address a specific opportunity in the market place" (Walton and Whicker, 1996)
- "Virtual enterprises materialize by selecting skills and assets from different firms and synthesizing them into a single business entity." (Camarinha-Matos et al., 1998)

More definitions can be presented by as many authors. All while *verbally* different are centred around a common concept, that of a temporary collaboration of core competencies and capabilities to exploit a business opportunity or need (e.g. delivery of a one-of-a-kind-product. While a detailed exploration is out of the scope of this paper, we continue with a short discussion on some core concepts and functions.

3.1.1. Core Concepts

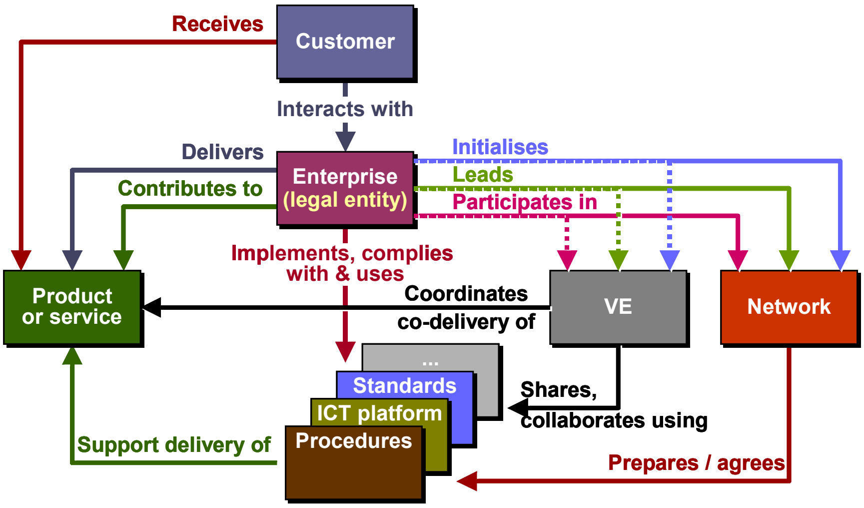

FIG. 3: Core concepts

When seen at a very high level, we can consider customer needs as an input and the product of service satisfying this need as an output. As such, a customer typically interacts with an enterprise (that is a legal entity), which then delivers the product. At the same time, an enterprise may initialise, lead or participate in a network. This network prepares/agrees upon a set of standards, procedures, ICT platforms, etc. to support the delivery of the product. It is to be noted that the network itself does not physically participate in any phase of product/service development or delivery. To deliver the product, it is the enterprise which implements, complies with, and uses, the standards, procedures, etc. provided by the network. In situations where an enterprise on its own cannot deliver the product, a virtual enterprise is established to do the task. Again, the enterprise may initialise, lead, or participate in the virtual enterprise. Members of the virtual enterprise may constitute relevant members of the pre-established network (though this is not necessary). In its capacity, the virtual enterprise coordinates the co-delivery of the desired product/service to which individual participant enterprises of the virtual enterprise contribute. Information sharing and collaboration within the virtual enterprise is typically done through agreed standards, ICT platforms, procedures, etc. provided by networks. A summary of these core concepts is given in Fig. 3.

3.1.2. Core Functions

In the light of the core concepts presented in Section 3.1.1 and the high level discussion that a customer request is an input and the desired product/service the output, the following main functions for product/service delivery through a virtual enterprise can be clearly established:

- Operate enterprise
- Operate enterprise network
- Operate virtual enterprise
- Operate product collaboration

The main inputs, outputs, controls, and mechanisms for these core functions are shown in IDEF0 format in Fig. 4.

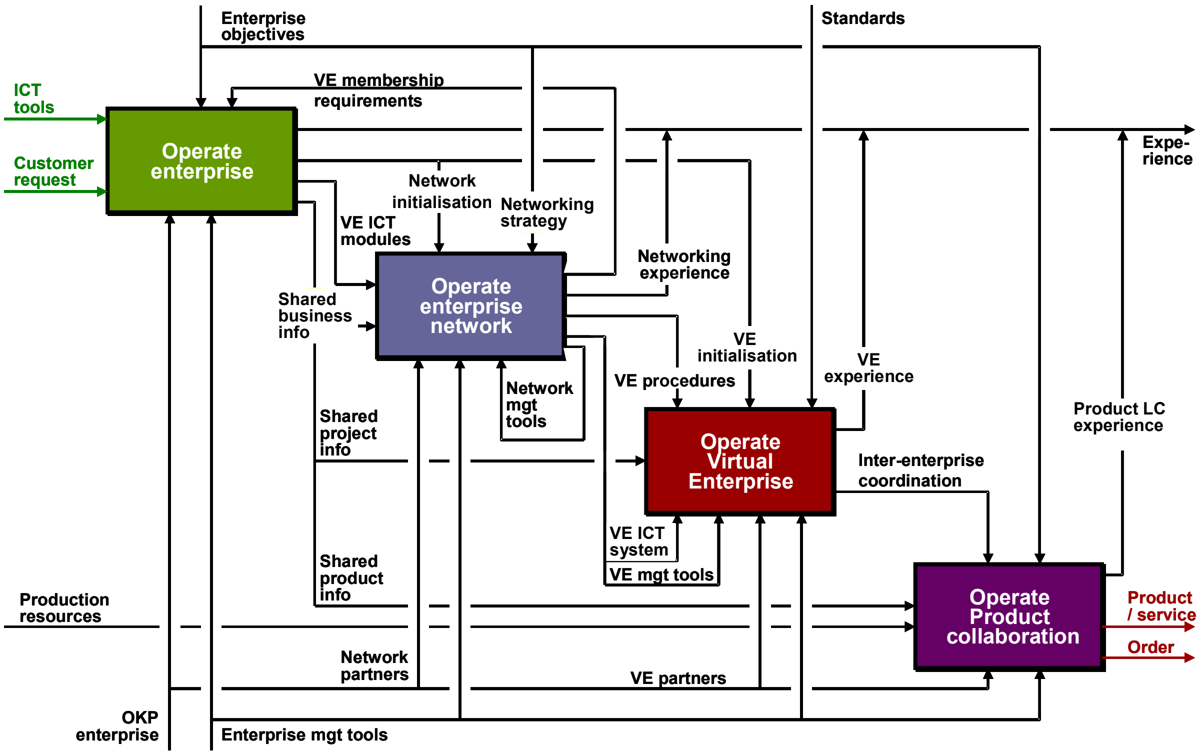

FIG. 4: Core functions

This high level IDEF0 diagram (Fig. 4.) was later validated by the industrial partners and specialised at several levels: domain level (e.g. distributed engineering), and industry level (e.g. construction).

3.2. Inter-enterprise communication mechanisms

When enterprises collaborate with each other under the VE umbrella, there is a need to have mechanisms through which information may be exchanged and shared (Fig. 5.). The past, present, and future of these mechanisms is presented below.

Traditionally, while using ICT, collaboration has been primarily through simple document exchange between individuals using email. This point-to-point form of communication has both led to data/information redundancy, and a lack of control to the physical source of the information and its owner. Furthermore, both information provider and information receiver needed the same software applications or viewers to be able to read/work on the same piece of information.

Of late, inter-enterprise communications in VE settings has been through a information repository where VE specific data/information is held. This mechanism whilst alleviating the problem of data/information redundancy and enabling information centralisation, is still individual user centric. Consequently the legal entity, the enterprise itself is not appropriately catered for. In simple terms, this implies that any information released by an individual would imply legal endorsement by the enterprise to which he/she belongs. Another problem left unresolved is lack of support for an organisation to release *partial* information whilst maintaining the whole *internally*.

As the authors see it, and have reported elsewhere (Kazi and Hannus, 2000), in the future, individuals would communicate only with their own central repositories, which in turn would communicate with a VE specific central repository. In simple terms, this would entail enterprise specific systems/repositories transferring and receiving *information packages* on a periodic or per-request basis to/from the VE specific repository. This ensures compliance with legal considerations, as in this form of communication the information released would be from the enterprise and not an individual. As such, an enterprise would simply need to *plug-and-communicate* to the VE repository. Both enterprise and VE specific systems would communicate through *interfaces* based on established standards and thereby eliminating/minimising the need for *compliance* between the two systems other than through the interface.

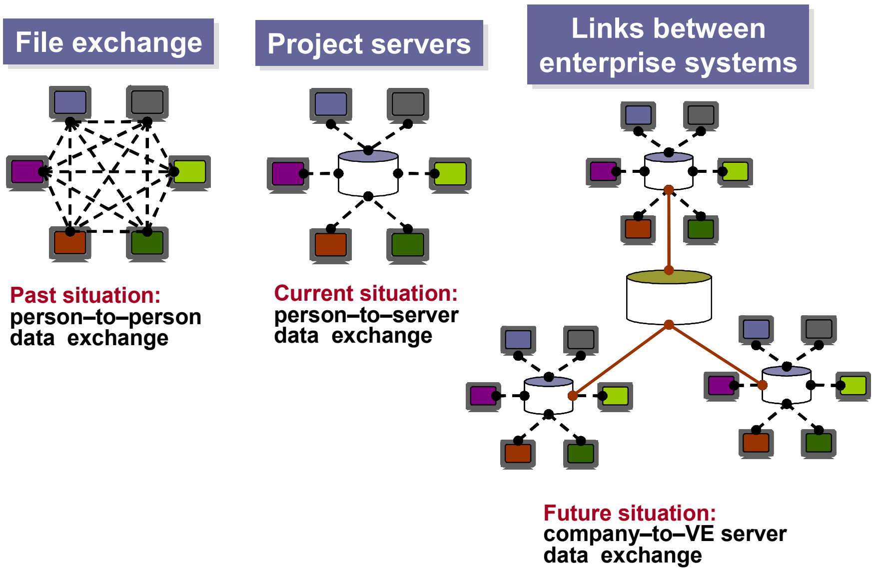

FIG. 5: Inter-enterprise Communication Mechanisms

The focus of the GLOBEMEN project is on current inter-enterprise communication forms (central VE repository), with some in-roads towards the future (enterprise systems *plug-and-communicate* with a central VE specific system).

4. VIRTUAL ENTERPRISE REFERENCE ARCHITECTURE AND METHODLOGY

Using requirements analysis from the different domains covered in GLOBEMEN, in addition to experiences related to the set-up, formation, operation, reconfiguration, and decommissioning of virtual enterprises, the consortium developed *VERAM* (Virtual Enterprise Architecture and Methodology).

The following sub-sections provide a brief overview of VERAM and then focus on a presentation of the defined infrastructure modules/layered architecture for inter-enterprise collaboration. This is to be later demonstrated for a specific case later on in this paper.

4.1. VERAM

Using requirements analysis from the different domains covered in GLOBEMEN, in addition to experiences related to the set-up, formation, operation, reconfiguration, and decommissioning of virtual enterprises, the consortium developed *VERAM* (Virtual Enterprise Architecture and Methodology). VERAM is an architectural framework that positions elements that support modelling, formation/set up, management and ICT support of VEs, such as reference models, and supporting tools and infrastructures. Interrelations among these elements are furthermore indicated. VERAM is about those tools, applications, models, etc., that can be used during the formation and operation of VEs and networks. It additionally includes a methodology, which describes how an organisation should use the various components of the architectural framework during virtual enterprise engineering. While a more elaborate presentation of VERAM is provided elsewhere (Zwegers, et al 2001), VERAM is illustrated in Fig. 6, and a short high-level description of the main constituents is provided in this paper.

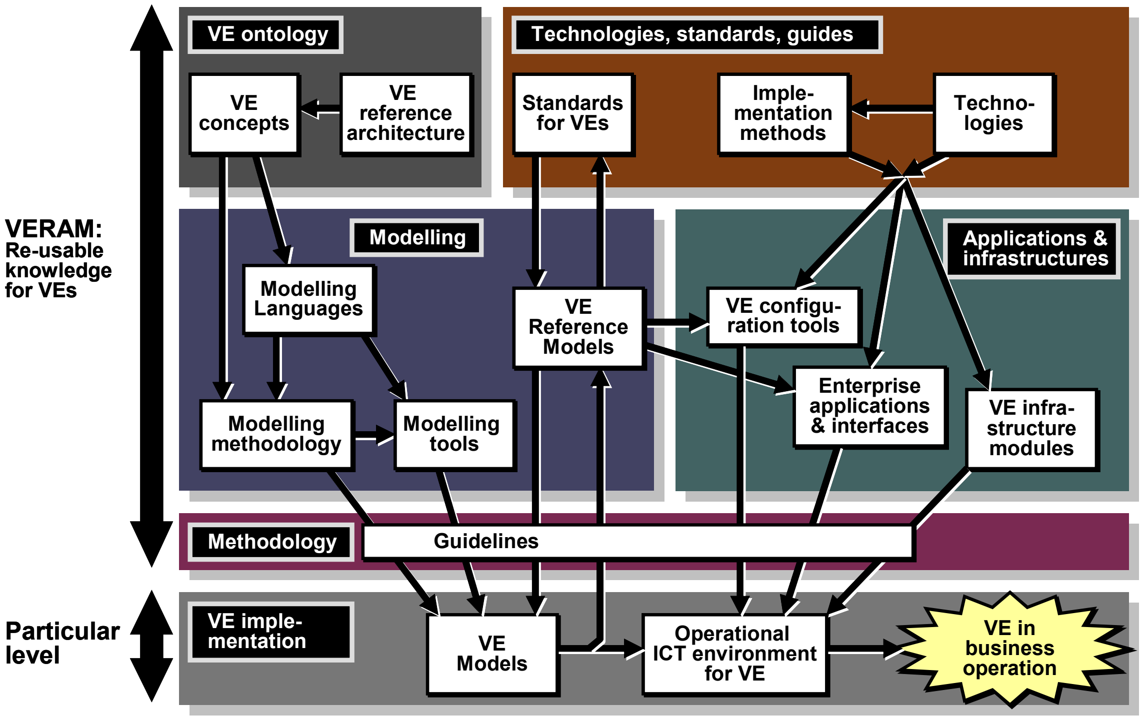

FIG. 6: VERAM (Virtual Enterprise Reference Architecture and Methodology)

The main constituents of VERAM are summarised below:

- *VE ontology*: This box, presenting VE concepts and a VE reference architecture, focuses on a description of shared concepts related to VEs for the purpose of enabling shared understanding and communication. It may be seen as a conceptual information model that describes "the things that exist" in a VE, namely: concepts, properties, facts, rules and relationships.
- *Technologies, standards, guides*: This box presents standards for VEs, implementation methods and tools, and technologies. These contain and define the factors that affect the way a VE is put into operation. It contains a rather broad set, ranging from technologies to legal aspects. Some of these factors could potentially lead to different contingencies in the VE formation, while other factors could lead to different VE implementations.
- *Modelling*: The different elements of this box allow enterprises to analyse and re-design the business processes of a VE. Both during the formation and even reconfiguration of a VE, enterprises may acquire knowledge of current business processes by means of modelling. This knowledge is needed in order to analyse the existing processes and communicate about them. It should be noted that the

focus is on modelling to support application-application and service-service interaction in an inter-enterprise setting.

- *Applications and infrastructures*: This box defines the different components that perform or support the business processes as identified in the *modelling* box. As such, they provide the (technological) realisation of those business processes, enabled by the technology as defined in the *technologies, standards, and guides* box. The *applications and* infrastructures box focuses on the execution or support of the formation or operation of virtual enterprises and networks.
- *Methodology*: This box focuses on the concretisation of the findings of the previous boxes in the form of a set of implementation guidelines as applicable to a VE. The guidelines cover the entire project life-cycle showing what happens when, how, where, by whom, using what means, etc. A collection of identified best practices is also provided.
- *VE implementation*: This box is at a particular level, (see Fig. 6) for the formation, instantiation, operation, reconfiguration, and decommissioning of a particular VE. Focus here is on the specific VE model and operational ICT environment for the VE in question.

4.2. Layered ICT Architecture for Inter-enterprise Collaboration

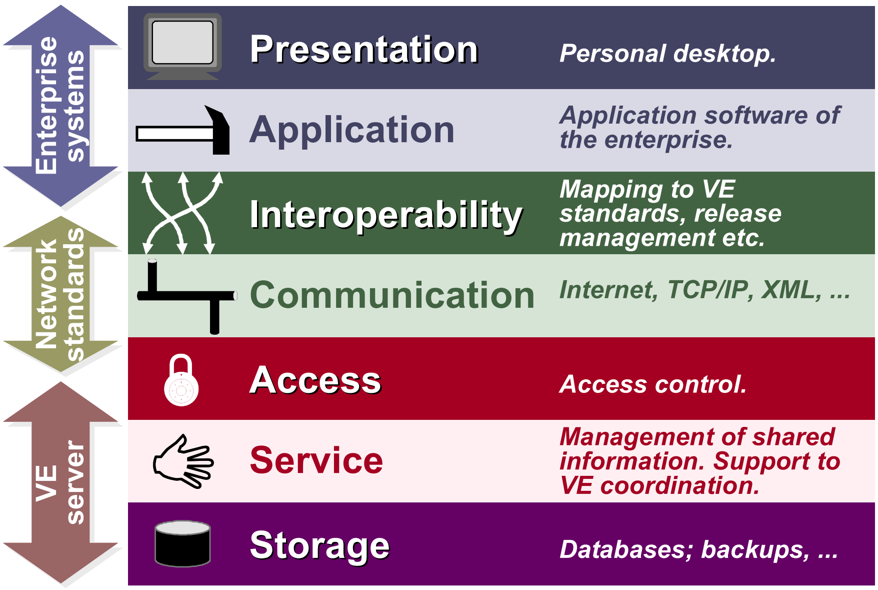

FIG. 7: Infrastructure modules/layered ICT architecture for inter-enterprise collaboration

A modular approach was used under *infrastructure modules* (see Fig. 6) to help identify a layered architecture for inter-enterprise collaboration in a global engineering and manufacturing setting. This architecture (see Fig. 7) attempts to in a diagrammatic fashion portray how different enterprises may exchange and use information between their respective organisation specific proprietary systems and a central server. The central server maintains and enables access to common data sources (released information and documents) and services.

VE Server

This constitutes the “central” server domain. It contains three distinct layers:

- Storage: this layer is the one which describes databases, backups, etc. In short this is the VE server repository. We can also describe this layer as the *persistence layer*. A document management system would for example fall in this layer.
- Service: this layer describes the services which are made available through the VE server. Typically these services will operate over the data/information contained in the storage layer. We may also describe this layer as being the *business logic layer*, as it contains the rules and definitions to carry

out certain tasks on the content held within the *storage layer*. Some simple examples would include: management of shared information, calendaring, access to documents, etc.

- Access: this layer describes the different forms and mechanisms available for access control to the VE server. Once access is granted, a user would b able to access the services available in the service layer. Furthermore, this layer in conjunction with the service layer could offer audit trailing capabilities. While primarily concerned with access control, this layer could also present some interaction interfaces to the outside. It is to be noted that the end-user or external applications would interact with the VE Server through this layer. This layer can also be described as a *security layer*. Some examples could include https, simple access control using some scripts written in asp, jsp, etc.

Network Standards

This constitutes the path and mechanism through which an end-user would through his/her organisation specific applications/environments interacts with the *VE Server*. It consists of one distinct layer:

- Communication: this layer describes the various protocols, methods, and formats through which an end-user or application could potentially communicate with the VE Server. In simple terms, this layer acts as the *interface* through which the Enterprise Systems and VE Server communicate. Relevant examples include: internet, TCP/IP, etc.
-

Enterprise Systems

This constitutes the working environment and systems of the end-user through his/her organisation specific applications and platforms. It consists of 3 layers:

- Application: this layer describes different organisation specific applications. It may be seen as a composite of one or more of the *persistence*, *business logic*, and/or *access layers* for a particular application. Some examples for this layer could include the organisational document management system, email server, etc.
- Presentation: this layer describes the different *user interfaces* and *mechanisms* through which a user can physically interact with an application. It is to be noted that while some applications would provide such an interface, for others one may need to be built in the form of for example some simple web pages.
- Interoperability: this layer describes the *mechanisms* and *standards* through which application specific information is restructured or mapped to comply with mutually agreed upon standards for data and information exchange. Since many applications have different internal data/information semantics and representation norms, this layer is required to ensure and foster compatibility with other applications/services. Typically these other applications would read in this information and then re-map it to the application specific data/information semantics and representation norms. Examples for this layer could include: IFC file conversions, conformance to some particular schema, etc.
-

It should be noted that certain elements may span across more than one layer. This will be demonstrated later in the context of a distributed engineering in construction prototype environment.

5. DISTRIBUTED ENGINEERING

Project oriented modes of operation under the modus operandi of virtual enterprise have been a long established practice in the construction industry. One of a kind product such as buildings, encompassing the involvement of multiple organisations during the product life cycle. At the same time, most of the currently available commercial ICT tools are geared towards internal use with organisations and their associated supply chains. As such, a need exists for ICT support for dynamic and at times geographically and organisationally dispersed project teams.

It is to be noted that in both one-of-a-kind and small batch production, the control of the information flow becomes critical if the design, engineering, and manufacturing processes are performed in a globally distributed environment. This is the key guiding statement for work related to distributed engineering in construction in the GLOBEMEN project.

In forthcoming sections, a discussion of some of the salient characteristics of distributed engineering in construction is discussed followed by some use cases, mapping of the different modules onto the layered architecture illustrated in Fig. 7, and finally a short walkthrough the prototype that is still under development.

5.1. Distributed Engineering in Construction

A good starting point to understand distributed engineering in construction can be taken through the example of a typical building construction project. An end-user makes a request to build a building. As is common knowledge, this involves the collaboration of different organisations offering distinct competencies to deliver the building product as illustrated in Fig. 8.

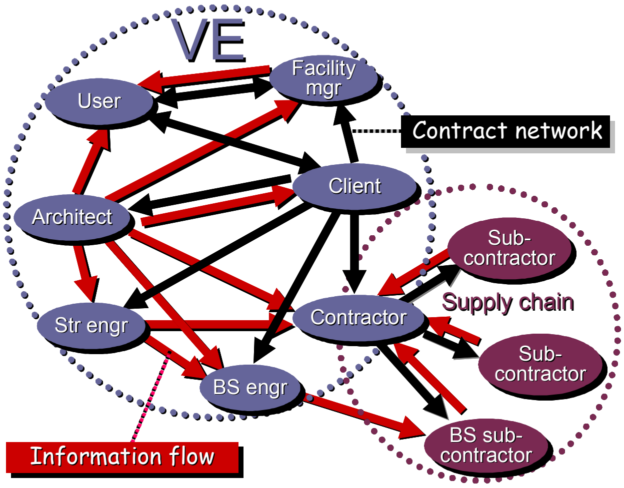

FIG. 8: Information flows and contract networks in construction

As is evident from this illustration, a virtual enterprise type collaboration is visible in addition to a traditional supply chain one. Furthermore, it may be noted above, that while information flows seem to stem from the architect, contractual flows are around the client. While several variants of the above exist, rarely, if at all, information flows and contractual flows are in parallel (Kazi, et al. 2001).

Salient characteristics of distributed engineering in construction (Hannus and Kazi, 2000) are:

- temporary relationships
- some participants are not known in advance
- complementary competence is provided by distinct companies
- absence of a dominant actor
- disparity between contractual relationships and information flows
- participation of some actors in other distributed engineering settings concurrently
-

These characteristics were noted by YIT construction (an end-user in the GLOBMEN project), and methods to work in this dynamic environment identified.

5.2. Using Product Models to Enable Distributed Engineering in Construction

Developments in the domain of product data technology (PDT) have paved the way to the formulation, adoption, and exploitation of standards such as the Standard for the Exchange of Product Model Data, STEP (ISO 10303) and the Industry Foundation Classes (IFC). Applications and toolkits that enable and handle the potential of this technology are now emerging.

Use of PDT in a distributed engineering context has several advantages. Product models are only schema dependant and not application dependant. In practical terms, this translates to the fact that once a common schema has been agreed upon, a significant amount of inter-operability is achieved. An architect could for example export his drawings in the form of a product model from a drawing toolkit that could then be taken by a structural engineer and opened in his CAD programme for further analysis and design. While the application software of the architect and designer may not necessarily understand each other or even know if the other exists, they have the ability to read, and export product models (Kazi, et al. 2001).

Product model based distributed engineering in construction is seen as the means for enablement of information sharing, exchange, and interoperability between different applications and stakeholders.

5.3. Building Construction Life Cycle: From As-Is to To-Be

Five distinct building construction life cycle stages were identified that needed improvement to enable product model based distributed engineering in construction in a dynamic global engineering and manufacturing environment. These stages are: briefing, design, product planning, construction, and use and maintenance. Table 3 presents an As-Is to a To-Be analysis of these stages.

TABLE 3: Building construction life cycle: from as-is to to-be

	As-Is (Before (GLOBEMEN)	To-Be (In GLOEMEN)
Briefing	- Sketches are done by architect manually - No systematic analysis - Key figures manually calculated on a case by case basis	- All sketches are based on a product model - Visualisation (interface) is automatically available for end-user in he form of graphs, 3D models, key figures, etc.
Design	- Only advanced architect and designers can produce product model information	- Product model is utilised by all stakeholders including structural designers, HVAC engineers, etc.
Product Planning	- No systematic feedback from contractor to architect/designer	- All partners have access to contractor's experience (available in the form of structural types, etc.)
Construction	- Information management is restricted to simple document management - Only some partners are able to use product models in their internal works and associated applications	- Shared product model is available for and used by all partners
Use and Maintenance	- No as built model is created	- Contractor and suppliers add as built information to product model - Product model information is exploitable for a long period

5.4. Modelling Methodology

A systemic methodology was employed in GLOBEMEN to come to application interfaces from simple user requirements. A combination of both top-down and bottom-up approaches was employed. Furthermore, abstraction and analysis was done at three distinct levels: Generic (one-of-a-kind-product [OKP] reference model, Partial (requirements), and Particular (implemented system).

Inspired in part by GERAM (Generalised Enterprise Reference Architecture and Methodology) and the modelling approach used in the OSMOS (Open System for Inter-enterprise Information Management in Dynamic Virtual Enterprises) project (http://osmos.vtt.fi), an extensive modelling exercise was conducted. The basic modelling approach used for all addressed levels of abstraction (generic, partial, and particular) a combination of different modelling methodologies was used as follows:

- *Step O*: GERAM modelling framework used to define the basic modelling views and concepts
- *Step 1*: IDEF0 used to define target processes for analysis
- *Step 2*: Processes mapped from IDEF0 to use cases (UML notation) and further elaborated
- *Step 3*: Potential system modules defined through use case analysis
- *Step 4*: Interactions between system modules identified and modelled using sequence diagrams
- *Step 5*: Functions and interactions between system modules specified in the form of interfaces
-

The modelling approach is summarised in Fig. 9.

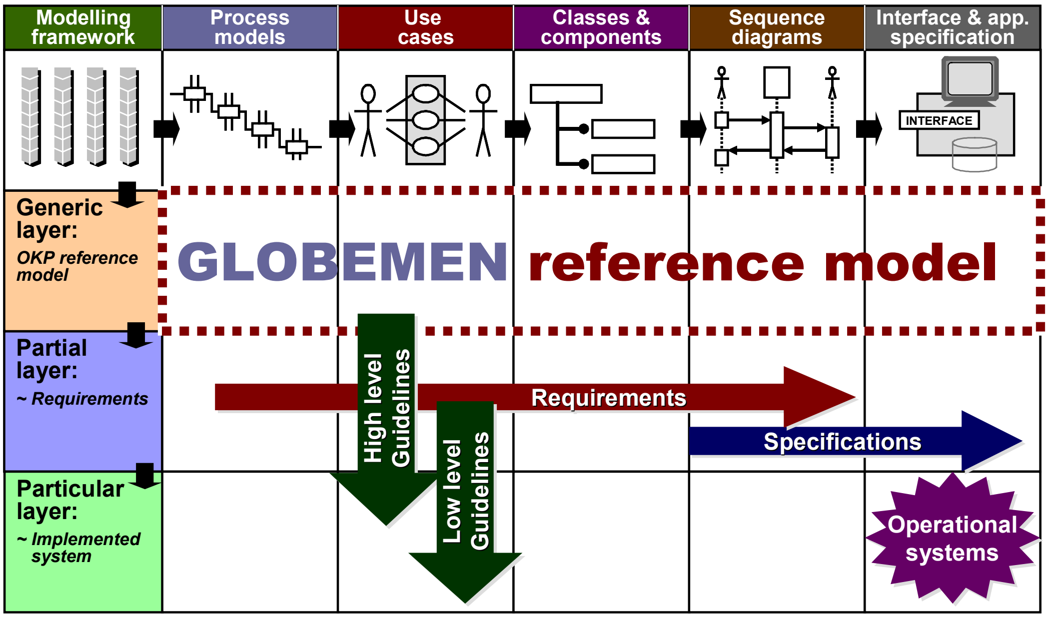

FIG. 9: Modelling methodology employed in GLOBEMEN

Details of the findings of each step of the process are out of the scope of this paper. A presentation of some of the main use cases for distributed engineering in construction is however made followed by the prototype architecture and a simple walkthrough thereon.

5.5. Main Use Cases

Targeting product model based distributed engineering in construction, YIT identified four high level use cases:

- Data exchange
- Model management
- Best practice management
- Linking documents with models
-

5.5.1. Data exchange

Description: An architect would typically create and then upload a project specific product model to the product model server. Structural engineers and contractors could then download the product model and do their planning and design on its basis. After completing their work, the designers and contractors would upload their *partial* models to the product model server. These partial models are then merged into the initial product model created by the architect. The precast concrete element manufacturer would then continue production planning based on the updated product model available for download from the product model server. Manufacturing schedule information would be added to the product model which would then be uploaded and updated on the product model server. This product model is now available to all stakeholders.

Actors: architect, structural engineer, precast concrete element manufacturer, contractor, product model server

Use Cases:

- Send IFC models as files
- Get model
- Integrate partial models with architectural model
- Add schedule data to model
- Get scheduled model
- Add precast element's manufacturing schedule

5.5.2. Model Management

Description: The focus of this use case is on the set-up and configuration of the product model server for a VE setting. A system manager would establish different roles with different levels of access rights for different groups of users. Furthermore, user profiling is supported so each individual user may have his/her own profile in terms of an interface, special requirements, classifications, etc. This use case also enables basic product model server functionality in terms of product model upload, merge, partial extraction capabilities. The configured environment enables partners to work concurrently.

Actors: system manager, designers, precast concrete element manufacturer, contractor, product model server

Use Cases:

- Initialise system with model schema
- Define actor roles
- Set access rights
- Upload model
- Merge model
- Extract (partial) model
- Delete object/model
-

5.5.3. Best Practice Management

Description: The product model server, described in the previous use cases, is initialised on the basis of YIT's best practices knowledge library. This library includes hundreds of structural types and associated parametric details. Designers utilise this best practices library, which is both linked and downloadable from the product model server. Manual feedback from designers and suppliers is included into the process. The library is improved continuously on the basis of the feedback.

Actors: designers, precast concrete element manufacturer, contractor, product model server

Use Cases:

- Store preferred structural solutions
- Get preferred structural solutions
- Upload IFC model
-

5.5.4. Linking Documents with Models

Description: This use case focuses on the interlinking of objects in the product model server to documents stored in a project server. A user could select any kind of electronic document, like drawings, specifications, delivery notes etc. and select any object that belongs to the product model. A cross-referential link is then created between the document and the object of relevance held in a product model. The physical link/reference is stored in the product model. Links are updateable and removable.

Actors: designers, precast concrete element manufacturer, contractor, product model server, project server

Use Cases:

- Select document
- Select object
- Create link
- Remove link
-

5.5.5. Distributed engineering in construction use cases: Global Context

A summary of how YIT's use cases relate to and fit into the overall scope of GLOBEMEN (European contribution) is illustrated in Fig. 10. This was derived on the basis of specialisations from top-down guidelines and templates, followed by generalisation of bottom-up specifications.

A use case distinction at three distinct levels of abstraction is noticeable.

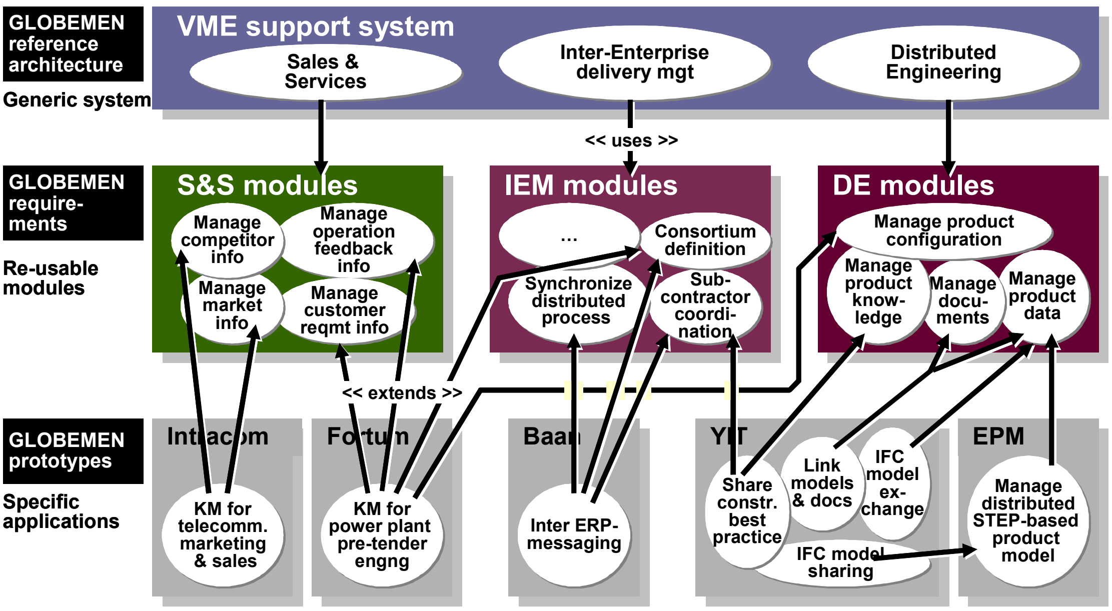

FIG. 10: GLOBEMEN use cases at three distinct levels of abstraction

5.6. Generic ICT Architecture for Distributed Engineering

It has been noted and reported that within dynamic distributed engineering in construction environments, it is necessary to have a usable and commonly accessible infrastructure that can be quickly set-up and configured (Kazi, et al. 2001). Furthermore, it is noted that since each participant entity would have its own legacy system, there is a need to both differentiate and maintain what is *internal* and *common* (released) information. It is advocated in GLOBEMEN that organisation specific legacy systems could exchange information with each other in a transparent manner through provision of and extraction from a shared distributed engineering environment.

The environment would provide repositories for storage of shared and released information in addition to the provision of some common VE specific services.

Initially a generic form of a possible distributed engineering environment was developed (Fig. 11). This consisted of two main components:

- *Distributed Engineering Environment*: This constitutes the *common* data and information repository, and sharing environment for entities participating in the distributed engineering setting. Typical services provided include distributed groupwork and sharing of information based upon available (or agreed upon) standards. Furthermore, depending on the needs of the participants and of the product or service to be delivered, different product model management, document management, and process management were incorporated,
- *Distributed Engineering Interface*: This constitutes the main interface and *front-end* through which organisation specific legacy systems would interact with the distributed engineering environment. As such, this interface acts as a communication bridge between the two systems and furthermore, as a control agent for the transfer and management of released information from organisation specific systems to the *shared* distributed engineering environment.

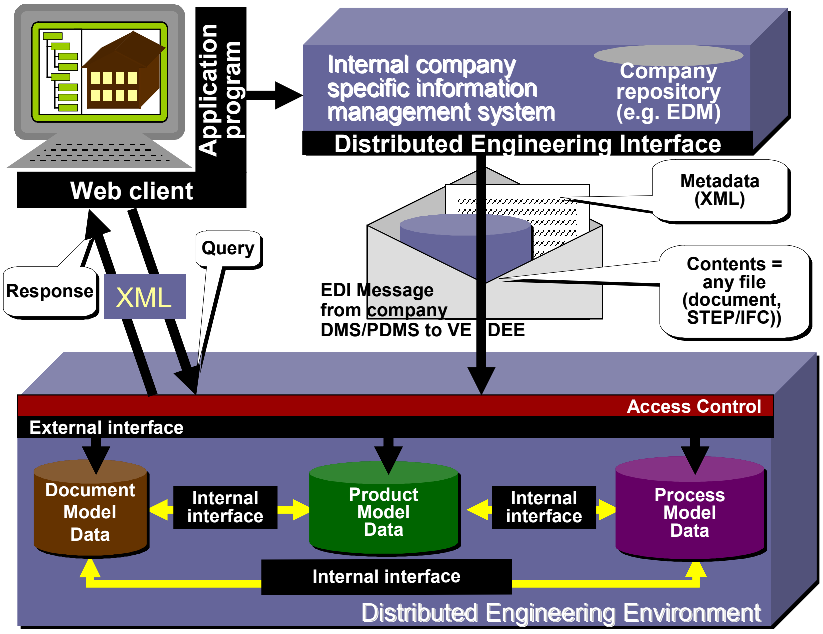

FIG. 11: Distributed engineering environment

5.7. Distributed Engineering in Construction Prototype

The *generic* ICT architecture for distributed engineering (Fig. 11) was made more specific and fine-tuned to the needs and requirements of construction, as specified by YIT's use cases for product model based distributed engineering in construction. This involved the definition of the *common* distributed engineering environment (or VE platform), and its interface to the YIT specific environment.

5.7.1. Technologies to be Exploited/Developed

An exploration was made of the available technologies to construct the overall environment and to deliver the functional needs specified. While details are out of the scope of this paper, and have been in part presented elsewhere (Kazi and Hannus, 2000), a summary of the selected technology and associated functionality to be exploited or developed is summarised in Table 4.

TABLE 4: Technologies to be exploited for product model based distributed engineering in construction

Technology	Functionality to be Exploited/Developed within GLOBEMEN
Document management, EDM, PDM	Integration of product model management with current document management technology
Product modelling (STEP/IFC)	Development of product model repository with model merging and partial model extraction capabilities
Product data (GDL)	Utilisation of GDL (Graphical Description Language) technology as a representative and configurator of the partial model of a specified product
Virtual Reality	Development and utilisation of a 3D interface to access product model objects and related documents
XML	Development of an XML based product model server allowing data access over the Internet

5.7.2. Layered ICT Architecture for Distributed Engineering in Construction

A modular and layered approach was used to define the ICT architecture for product model based distributed engineering in construction. This basis was drawn from the layered ICT architecture shown in Fig. 7. It is to be noted that the *common* VE platform is the one with which company specific platforms would interact and exchange *released* information with. YIT's platform in association with the VE platform is illustrated in Fig. 12, and the contents of each module/layer described in Table 5.

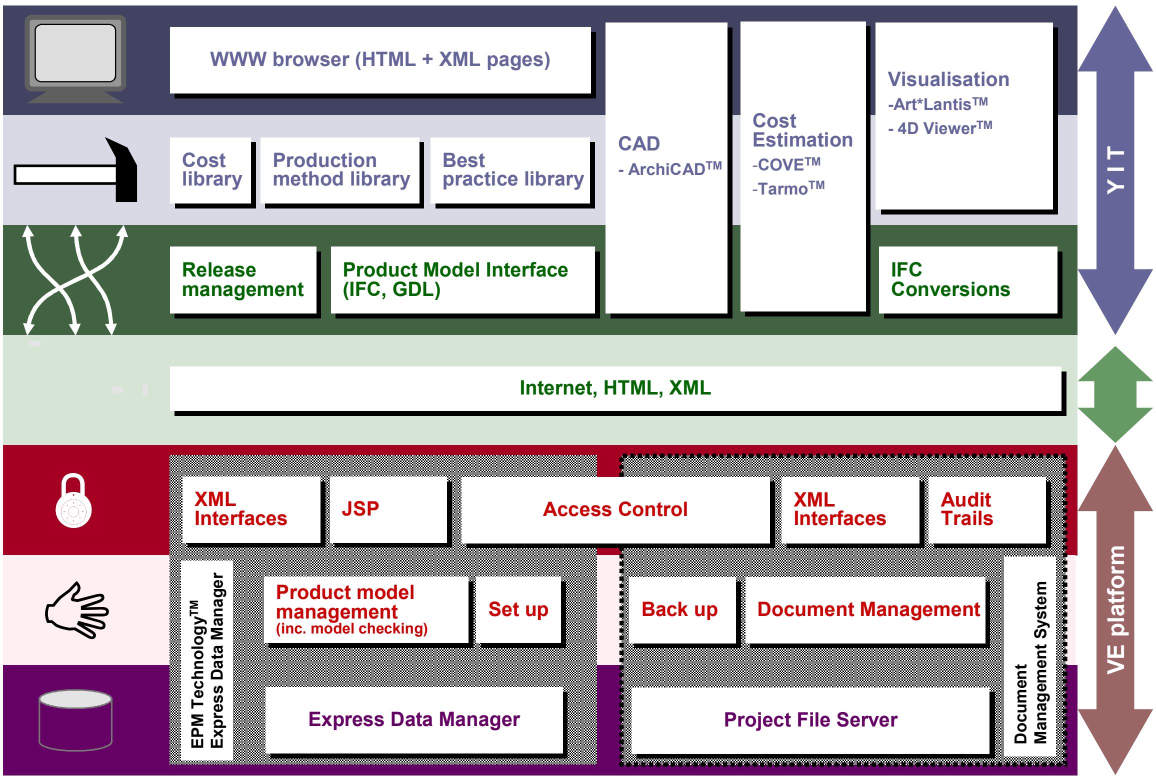

FIG. 12: Layered ICT architecture for distributed engineering in construction (YIT perspective)

TABLE 5: Description of layers in ICT architecture for distributed engineering in construction

Layer	Description
Presentation	In the foreseen implementation, this layer is served in part by several commercial products that present their own interfaces for the end-user to interact with. Alternatively, a simple web browser is used.
Application	This layer is the main container for organisation specific applications. In the implementation for YIT, it consists of different CAD tools, cost estimation tools, 4D rendering and visualisation tools, etc. Furthermore, this layer is host to their different libraries and associated management applications.
Interoperability	In essence, this layer enables the mapping of application specific syntactics and semantics to those of an agreed upon standard (e.g. IFC, from the IAI, GDL, etc.). Some applications, e.g. ArchiCAD™ possess a capability to do this on their own, while for others, this needs to be done through converters. This layer also controls and manages *released* information (this functionality is currently lacking in commercial offerings.
Communication	This layer enables the communication between (the ICT systems of) a specific VE participant like YIT, and the *common* VE platform. The Internet, data communication and exchange protocols, etc. form this later. Within GLOBEMEN, the focus is on the Internet and XML.
Access	This layered provides control access to the VE platform. Different user specific profiles are furthermore provided enabling content viewing in different forms (when a query is made). In a similar fashion, information filtering is provided. XML interfaces are made available for further view and content formatting.
Service	The service layer provides basic product data model management functionality such as, model merging, partial model extraction, etc. A simple document/file management service is also provided. Provision exists for backing up information and packaging it to participants upon decommissioning of the platform at the end of a project.
Storage	This layer forms the container where different types of data and information are stored. A simple file server is used for the storage of documents (e.g. CAD drawings, reports, etc.), while the Express Data Manager from EPM technology is used for product model storage.

6. CONCLUSIONS

ICT support to enable the seamless exchange of data/information in dynamic, geographically and culturally diverse, and one-of-a-kind product delivery in global engineering and manufacturing settings is a key requirement. Exploration of this need is done in GLOBEMEN through the addressing of three particular domains: sales and services, inter-enterprise delivery process management, and distributed engineering. A systematic methodological approach employing both top-down and bottom-up approaches is used to develop a generic virtual enterprise reference architecture that can then be specialised to a particular domain and even industry (e.g. construction).

The focus of this paper has been on distributed engineering in construction. It has been advocated that inter-enterprise collaboration and information exchange may b achieved through reliance on product data technology. Concurrent use of a common product model is seen as the means forward. With added capabilities to merge models, extract partial models, it is envisioned that true collaboration and exchange of *released* information is possible in construction. Two modes of realisation were presented: person to VE server exchange, and organisation specific application to VE server exchange. A generic ICT architecture for distributed engineering was presented as a means to this realisation. In simple terms, organisation specific systems would exchange and share *released* information through interfaces connecting each to a *common* VE platform.

Further developments in GLOBEMEN will focus on the development of industry specific prototypes that will then be inter-connected through a *common* VE platform. As such, the complete life cycle covering sales and services, inter-enterprise delivery process management, and distributed engineering will be covered. A demonstration of each prototype will be separately presented in addition to a common integrated demonstration based on a real-life scenario. More information and updates will be made available at the public GLOBEMEN website, http://globemen.vtt.fi

7. ACKNOWLEDGEMENTS

GLOBEMEN, Global Engineering and Manufacturing in Enterprise Networks, is an RTD project that has been initiated to define and harmonise ICT support requirements in various one-of-a-kind industries operating in various cultural environments. This is an IMS project involving 19 partners from different IMS regions. The European Commission within framework V of the IST programme funds EU participation in GLOBEMEN: IST-1999-60002.

8. REFERENCES

Afsarmanesh H., et al. (1997). Management of Distributed Information in Virtual Enterprises – The Prodnet Approach. *Proceedings of the 4th international conference on concurrent engineering*, Nottingham, UK, 241-255.

Browne J., Sackett, P.J. and Wortmann J.C. (1994). *The System of Manufacturing: A Perspective Study*, Report to the DG XII of the CEC.

Byrne J.A. (1993). The Virtual Corporation, *Business Week*, February 8, 37-41.

Camarinha-Matos L. M., Afsarmanesh H., Garita C., and Lima C. (1998) Towards an Architecture for Virtual Enterprises, *Journal of Intelligent Manufacturing*, Vol. 9, No. 2, 189-199.

Charbuck D. and Young J.S. (1992). The Virtual Workplace, *Forbes*, Vol. 150, No. 12, 184-190.

Filos, E. and Banahan, E. (2001). Towards The Smart Organization: An Emerging Organizational Paradigm And The Contribution Of The European RTD Programs, *Journal of Intelligent Manufacturing*, Vol.12, No. 2, 101-119

Hannus M. and Kazi A.S. (2000). Requirements for Distributed Engineering. *Proceedings of ECPPM 2000: roduct and Process Modelling in Building and Construction*, Lisbon, Portugal, 41-78.

Kazi A.S. and Hannus M. (2000) Functional Requirements for Inter-enterprise Intranet Services. *Global Engineering, Manufacturing and Enterprise Networks* (Mo, J.P.T. and Nemes, L., editors), Kluwer Academic Publishers, 54-60.

Kazi, A.S., Hannu, M. and Laitnen, J. (2001). ICT Support Requirements for Distributed Engineering in Construction. *E-work and E-commerce: Novel solutions and practices for a global network economy* (Stanford-Smith, B. and Chiozza, E., editors), IOS Press, 909-915.

NIIP (1996). The NIIP Reference Architecture (http://www.niiip.org/public/public-forum.html)

Rabelo R. and Camarinha-Matos L.M. (1996). Towards Agile Scheduling in Extended Enterprise, *Proceedings of BASYS'96: Balanced Automation Systems II- Implementation Challenges for Anthropocentric Manufacturing* (Camarinha-Matos L.M. and Afsarmanesh H., editors), Chapman & Hall, 413-422.

Walton J. and Whicker L. (1996) Virtual Enterprise: Myth and Reality, *Journal of Control*, Vol. 22, No. 8, 22-25.

Zwegers, A. et al (2001), An Architectural Framework for Virtual Enterprise Engineering. *E-work and E-commerce: Novel solutions and practices for a global network economy* (Stanford-Smith, B. and Chiozza, E., editors), IOS Press, 1117-1123.

OSMOS (2000), Open System for Inter-enterprise Information Management in Dynamic Virtual Enterprises (http://osmos.vtt.fi)