UDK 621.91

C. CONSTANTINESCU, A. KLUTH, Stuttgart, Germany

FLEXIBLE CONNECTION OF PRODUCT AND MANUFACTURING WORLDS: CONCEPT, APPROACH AND IMPLEMENTATION

К. КОНСТАНТІНЕСКУ, А. КЛУТ
ГНУЧКИЙ ЗВ'ЯЗОК СФЕР ПРОДУКЦІЇ Й ВИРОБНИЦТВА: КОНЦЕПЦІЯ, ПІДХІД ДО ВИВЧЕННЯ, РЕАЛІЗАЦІЯ
У статті наводяться короткі відомості про сучасні виклики керуванню / гармонізації у сферах розробки продукції й виробництва. Вона являє основи концепції та рішення в просуванні баз даних між неоднорідними й незалежними ICT-застосуваннями, що підтримують конструювання продукції, планування і адаптацію виробництва. Наше рішення із представленим оглядом архітектури "Керування гнучким відтворенням 2.0" наведено в деталях з обґрунтуванням сценарію для взуттєвої індустрії.
Ключові слова: виробниче планування, керування виробничими даними, гнучке відтворення.

В статье приводятся краткие сведения о современных вызовах управлению / гармонизации в сферах разработки продукции и производства. Она представляет основы концепции и решения в продвижении баз данных между неоднородными и независимыми ICT-применениями, поддерживающими конструирование продукции, планирование и адаптацию производства. Наше решение с представленным обзором архитектуры “Управление гибким воспроизводством 2.0” приведено в деталях с обоснованием сценария для обувной индустрии.
Ключевые слова: производственное планирование, управление производственными данными, гибкое воспроизводство.

The paper shortly introduces the current challenges of orchestrating / harmonizing the two worlds of product and factory engineering. This represents the basis for conceiving our approach and solution for the forwarding of data changes between the heterogeneous and autonomous ICT applications supporting the product design and factory planning and adaptation. Our solution, the “Change Propagation Manager 2.0” is presented through an architecture overview, implementation details and a validation scenario for the shoe industry.
Keywords: Factory Planning, Factory Data Management, Change Propagation

1 INTRODUCTION

Today`s enterprises contain of a set of different business units, connected by various information flows as well as resource flows. Every single unit, autonomous to manage its own resources, has the responsibility for delivering services or goods, which are further processed by other units. These units frequently run their own internal information systems in order to manage the data adjusted for the corresponding business processes [1].

These different business units are more and more autonomous and not all information systems store their data in the same way. Each unit uses its own system in order to support its business processes and activities. Moreover, some data is relevant only to a single unit, while other data is relevant to numerous systems and is stored locally. The higher the number of different business units within a collaboration scenario, the higher is the number of heterogeneous information systems that have to exchange data related to the different business processes. Therefore, data changes in one system have different effects on data stored in other systems [1, 2].

Focusing on the connection of product and manufacturing worlds, manufacturing enterprises are forced to deal with ever shorter product life cycles and increasing numbers of variants [3]. The majority of product changes - whether it is a single change, a new variant or a new product - requires an adjustment or redesign of production. With the increasing pressure towards an adaptable business development, the question rises, how to adapt plant structures in a shortest possible time. The most common basis of approaches for creating company structures in a versatile way is a complete awareness of all relevant production planning data of a factory or a manufacturing enterprise [4].

Taking into account the constant change of production systems and their equipment etc. it is evident, that the management of factory data in a useful way, can only take place over the entire life cycle of a plant. That means it is of particular meaning to save, structure and represent all the data from the early planning stages up to dismantling of the factory. Only if the factory planners get access to all needed data and especially the changes about the current configuration of the factory and the products, the planners can make deliberate decisions regarding adaption and optimization of the production [4].

After stating an overview about challenges and approaches in the field of data management, a concept prototype for data change propagation, developed at Fraunhofer IPA and University of Stuttgart – IFF, is given. Afterwards implementation aspects as well as an example scenario are highlighted and the final section concludes the paper regarding future activities to bring data change management on a higher level.

2 DATA CHANGE MANAGEMENT

2.1 System Heterogeneity

It is more and more important to increase the collaboration and communication between actors within a value-chain. Computer-based information systems are used to support the value-chain activities by allowing automated communication between single actors [2].

Most enterprises have a diverse environment of heterogeneous and autonomous information systems, which can be viewed as producers of/consumers for changed data. If the same data is relevant for several information systems, data changes coming from one system affect data stored in other consumer systems [5].

It is often infeasible or too expensive to manage a single, integrated enterprise information system that feeds all business units with their required data. Therefore, enterprise change management has to be supported by a generic approach to solve data change management. Such a generic approach has to provide the possibility to manage certain data dependencies and help to transform data stored in a source information system into data stored in the depending information systems [1].

Many enterprises have highly diversified IT infrastructures, and both applications as well as data management systems are evolving constantly [6].

One major requirement is to accommodate heterogeneity and to preserve the autonomy of each component. As a solution federated information systems offer access to diverse databases and systems in an integrated way [7].

2.2 Terms and Definitions

In the following section some important terms and definitions in the field of data management will be described and explained according to [1, 6].

In this approach, any software system providing access to data is called an *information system.*

A *data model* describes different data structures as classes and the different relationships. Also the specified attributes of the classes and objects are described. A *data schema* specifies the data structures for an information system. There can be more than one data schemas within an information system, and a single schema can be used in one or more information systems. In this approach a *system* is an application. A system can act as a producer and/or consumer of data. A *dependency* is a directed relationship between a data producer (*source system)* and a data consumer (*destination system)*.

Change propagation is the process of forwarding a data change from a source system to all dependent systems, which need the data. This propagation process includes transformations as well as filtering of the data changes. A t*ransformation* is here defined as mapping given input data into output data according to a certain specification. The specification defines how the input data has to be adapted in order to represent valid destination data.

Filtering is an operation that accepts or rejects a transformation according to a constraint, which is an expression (a set of conditions) over the contents of the input data of a transformation.

2.3 Used Standard for Data Exchange

Currently the four main challenges in data management are identified as: capture, storage, retrieval and exchange of data [8]. For the exchange of data, certain standards are necessary.

Extensible Markup Language (XML) is a simple and flexible text format, derived from the Standard Generalised Markup Language (SGML) which is described in the ISO 8879 Standard [9]. Originally, it has been designed for facing the challenges of large-scale electronic publishing. Furthermore XML is playing a more and more important role in the exchange of a wide variety of data for example on the Web [10].

XML is able to deliver portable data, and thus it is considered as a key web application technology. It defines a basic syntax in order to mark up data with simple and human-readable tags. For transforming data between different applications the XML format can be flexible customised. The XML specification syntax defines for example the delimitation of elements by tags, what a tag looks like, the acceptable names for elements and where attributes are placed. XML does not have a fixed set of tags and elements that are supposed to work for everybody in all areas of interest for all time. Developers and writers can define the elements according to their needs [11]. The main XML elements are the XML document, containing XML code, the XML schema, describing the structure of a document and its tags and the XML stylesheets, containing formatting instructions for an XML file. A created XML file can be presented in various ways by the application of different XML stylesheets. As standard text files, XML can be read by any application and therefore the XML format is independent of hardware and software [8].

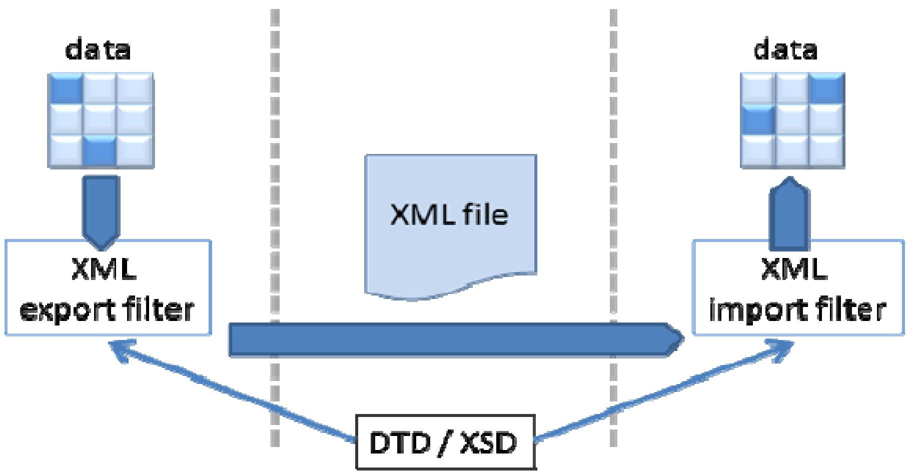

Figure 1 – Data Exchange via XML files, according to [12]

An XML document comprises a reference to a Document Type Definition (DTD), which declares its elements, attributes and grammatical rules. Furthermore it describes which attributes may be applied, their order of appearance, as well as the permissible parent/child relationships [12].

A present schema language, described by the W3C (World Wide Web Consortium) as the successor of DTDs, is the so called XML Schema Definition (XSD).It allows an information exchange in a standardised structure. It defines custom markup tags, which can comprise attributes for describing the content enclosed by these tags. By using an application called *parser*, information could be extracted from the tagged data in the XML document [12].

The exclusive strengths of using XML as a data format contains the simple syntax support for nesting, easy to debug and the independence of language and platform [11].

The use of XML in order to exchange data, provides several benefits, like the usage of human, not computer, language, for what it is therefore human readable and user friendly. It is fully compatible with Java and completely transferrable [13].

3 CHANGE PROPAGATION MANAGER 2.0

3.1 System Overview

The Change Propagation Manager 2.0 (CPM 2.0) bases on the philosophy of a Product Lifecycle Management by integrating Product Data Management (PDM) with Factory Data Management (FDM). This flexible integration allows a communication between product and factory data, supporting the synchronization of the product and process data. Moreover CPM 2.0 supports the propagation of product data from the product design applications to the FDM system and vice versa (XML data exchange). This system guarantees that all exchanged data is compliant with the defined Data Model. In this way a decoupling between manufacturing side tools and product design tools will be achieved without losing the integration between the two sides. The tools export their data in a common XML-based file format. The specific XML-formats from the different applications have to be transformed into the common XML, before they are being passed to the CPM 2.0.

This solution achieves the decoupling of the tools, both between each other and from the data management system, implementing the central role of the Data Model as the common speaking language. All the tools, being connected to this centre, become really independent one to each other and communicate only through CPM 2.0. The focus is not on providing a global view for querying data sources but more on providing a solution for flexible integration and a flexible method for propagating data between autonomous and heterogeneous application systems from product and manufacturing world.

The developed prototype is a part of the European research project “Design of customer driven shoes and multi-site factory” (DOROTHY – EU FP7-NMP) on innovative concepts and technologies to enable highly flexible mass customization systems in the manufacturing shoe industry.

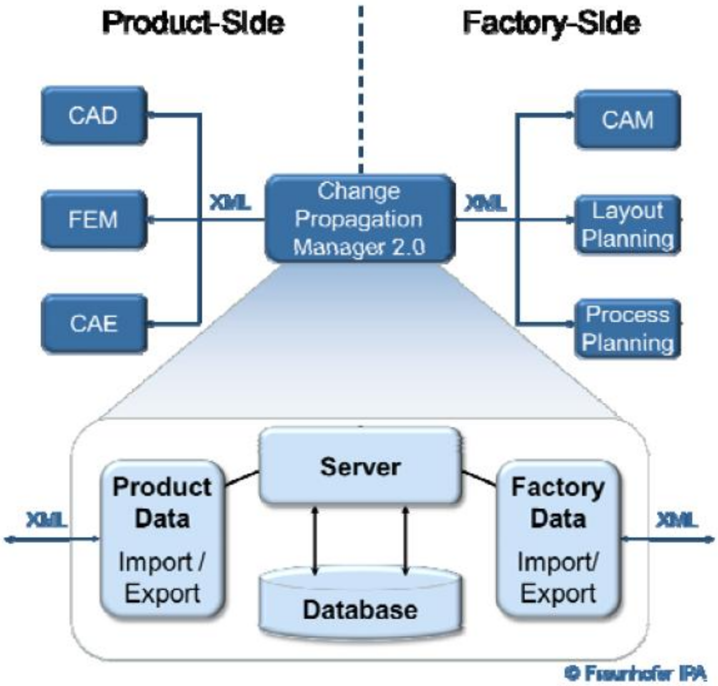

Figure 2 – Change Propagation Manager 2.0 Landscape [4].

3.2 Implementation

The Change Propagation Manager 2.0 is a system aiming to fulfil two main functionalities:

• Flexible integration of Factory Data Management and Product Data Management within a PLM Software supporting the synchronization of product and process data.

• Coupling of the product side application to the PLM System as well as of the factory side application to the PLM System, supporting the propagation of product and process data from/to product side application to Data Management System and of the product, process and resources data from/to the factory application to the Data Management System.

The propagation of data and the corresponding changes is implemented by using service-oriented and grid based architecture. This form of architecture allows the system, depending on the planning phase, to create connections between the different single software applications. Therefore for example at the end of the product-design phase the CAD data can be used for the Product Data Management (PDM) System as well as the Factory Data Management (FDM) for other planning and engineering activities. On the other hand, the adjustments referring to subsequent changes such as of equipment or tool data will be written back to systems of earlier stages of development, such as the aforementioned CAD system. For this purpose the individual applications of both the product and the factory world can connect with the server of the CPM 2.0 via online services and Client-/Server applications in order to upload their respective data in the form of XML files. The digital information are collected, transformed and automatically checked against a central database (Siemens PLM System Teamcenter Manufacturing). This allows the identification of possible changes of the data. These changes will also be described in an XML file and forwarded to the server from the PLM System. Afterwards through a Notification Service the involved software applications will be notified, that there have been done significant data changes needed for their planning activities. The changed data can be downloaded from the server or transmitted directly into the corresponding program via specified client interfaces. This creates a common and consistent database. The core of the CPM 2.0 system is a centralized data model as a template for creating product and process data.

3.3 Processing Model and Example Scenario

The following scenario describes the usage of introduced technologies by illustrating a simple data propagation example (Figure 3). This processing model refers to a manufacturing enterprise from the shoe industry that receives a new order for producing a customized new shoe. At least four systems of the enterprise are involved within this scenario:

• *Shoe Design*: Defining the basic shoe components and design for configuration of new shoes.

• *Shoe Customization*: Generating the customized order for new shoes to be produced.

• *Factory Layout Planning*: Planning and adapting the resources/layout concerning the new product.

• *Process Planning*: Planning and adapting the processes concerning the requirements of the layout and the new order.

The scenario flow highlights the main concepts for data change management: the transformation, filtering and routing of data changes. The given scenario consists of the following steps to represent an overall realistic processing model:

1) Upload of the shoe-catalogue from Shoe Design Module to CPM 2.0 via Client-/Server application.

2) Distribution of the "catalogue.xml" to other tools like Shoe Customization Module.

3) Upload of product-structure from Shoe Customization to CPM 2.0 via Client-/Server application.

4) Loading the product-structure from CPM 2.0 directly into Teamcenter Manufacturing (PLM System); CPM 2.0 generates XML (transformation) with the productstructure for the other tools like Process Planning.

5) Upload the resource-structure from Factory Layout Planning to CPM 2.0 via Client-/Server application.

6) Loading the resource-structure from CPM 2.0 directly into Teamcenter Manufacturing; CPM 2.0 generates XML (transformation) with the resource-structure for other planning tools like Process Planning.

7) Process Planning tool imports product-structure and resources-structure directly via integrated interface.

8) Process Planning activities (Processes sequence; linkage of products, processes and resources; visualization of the material and process flow).

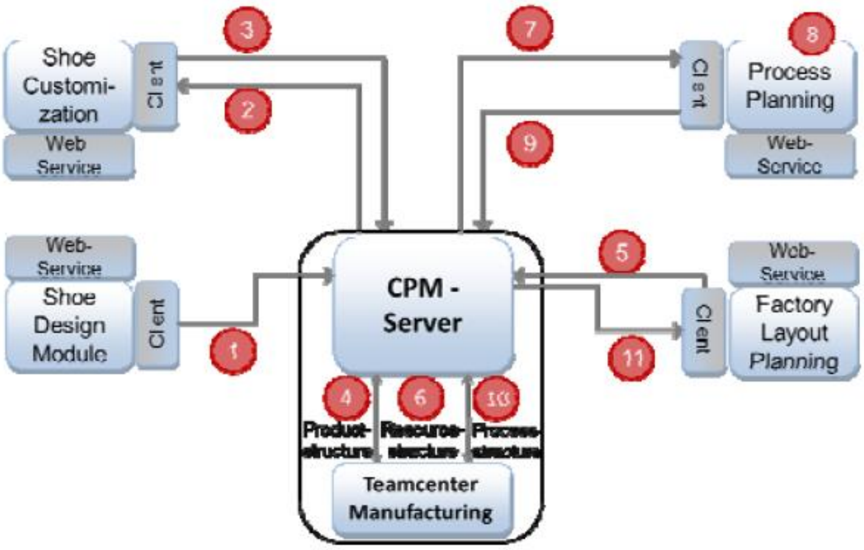

Figure 3 – Data Propagation Scenario

9) Export (via integrated interface) and upload of the process data, including linkage between products, processes and resources from Process Planning to CPM 2.0 via Client-/Server application.

10) Loading the process data (Linkage between products, processes and resources) from CPM-Server directly into Teamcenter Manufacturing; CPM 2.0 generates XML with the operations-structure and production plan for Layout Planning.

11) Distribution of the transformed data to the involved engineering tools (e.g. Layout Planning) in order to fulfil their planning activities.

Summing up, the Change Propagation Manager 2.0 enhances a flexible integration between product and manufacturing worlds, able to manage synchronized product and process data, collecting and propagating the changes in one single software-based platform.

3.4 Advantages and disadvantages

The main advantage of the presented approach is the flexible connection between heterogeneous planning modules / software applications. By using a corporate Data Model and a common XSD schema the different structures, objects as well as attributes etc. can be described. This leads to a common understanding and communication between the autonomous software tools. The defined Data Model within the presented overall approach follows the specified Data Model of the used PLM System. This ensures that the storing of exchanged data into the PLM System can be realized in a more easy and effective way.

Another advantage of the presented data propagation system is the use of standards for interfacing and data exchange between the different applications, which have to be connected. This offers advantages for the communication regarding openness and simplicity [14].

The presented approach of the Change Propagation Manager 2.0 is not restricted to a certain industry scenario. The approach and the existing system can be adapted and used for several industry sectors, e.g. automotive, machine-building and aerospace industry.

Some of the involved tools within the presented scenario imply real production measured data as input or specific customer data as output. However the presented system does not foresee an ERP development or integration. A solution for that issue could be to foresee an interface to a generic ERP, to understand if the involved tools need a real ERP or eventually to plug an ERP behind these interfaces.

4 CONCLUSION AND FUTURE WORK

Present IT infrastructures require a flexible and loosely coupled methodology for propagating data changes between different enterprise information systems while conserving their data management autonomy [2]. Within this paper, the shown approach suggests a software component, called Change Propagation Manager 2.0 (CPM 2.0), that deals with dependencies between data stored in potentially different schemas and models and with the propagation of needed data to different systems. The Change Propagation Manager 2.0 achieves a transformation of an incoming XML message into an XML output message based on a transformation specification that has been defined for certain data dependencies [1] and workflows. For the future work, more standards for data exchange have to be defined and developed in order to enable new concepts and architectures for integrating heterogeneous software applications.

Concerning the described example scenario, there should be further work on integrating and using workflow management systems as well as adapting such concepts and approaches from manufacturing sector to other industrial sectors.

Apart from all the technical aspects, the significance of integrating knowledge within such change management approaches will increase more and more in the future. Therefore not only the dealing and the propagation of data, but capturing, transferring, managing and integrating knowledge through change management systems between different applications and users should be more taken into consideration.

5 ACKNOWLEDGEMENTS

The authors would like to thank Mr Axel Bruns, the Chief Integration Officer of our Grid Engineering for Manufacturing Laboratory (GEMLab) in Stuttgart for supporting us in developing and implementing our ideas and concepts on European level as well as into our GEMLab.

REFERENCES: 1. *Constantinescu, C., Heinkel, U., Rantzau, R., Mitschang, B.,* 2002, A System for Data Change Propagation in Heterogeneous Information Systems. **2.** *Makris, S., V. Xanthakis, D. Mourtzis and G. Chryssolouris,* 2008, On the information modeling for the electronic operation of supply chains: A maritime case study, Robotics and Computer-Integrated Manufacturing, (Vol.24, No.1, 2008), pp. 140-149. **3.** *Alexopoulos, K., Makris, S., Xanthakis, V., Chryssolouris, G.,* 2010, A Web-Services oriented workflow management system for integrated production engineering, Proceeding of the 43rd CIRP International Conference on Manufacturing Systems, Vienna, Austria (May 2010), pg. 351. **4.** *Westkämper, E., Constantinescu, C., Landherr, M., Kluth, A., Eichelberger, H.,* 2010, Fabriklebenszyklus-Management; Unterstützung des FLM durch PLM, it-Production: Product Lifecycle Management (PLM) 2010/11. **5.** *Constantinescu, C., Heinkel, U., Meinecke, H.,* 2002, A Data Change Propagation System for Enterprise Application Integration. **6.** *Rantzau, R., Constantinescu, C., Heinkel, U., Meinecke, H.,* 2002, Champagne: Data Change Propagation for Heterogeneous Information Systems. **7.** *Constantinescu, C., Heinkel, U., Rantzau, R., Mitschang, B.,* 2001, SIES - An Approach for a Federated Information System in Manufacturing. **8.** *Watson, R. T.,* 2009, XML - Managing Data Exchange, available at: http://en.wikibooks.org/wiki/ XML_Managing_Data_E Exchange, 2010-12-15. **9.** International Organization for Standardization – ISO 8879, 1986 Information processing – Standard Generalized Markup Language (SGML). **10.** World Wide Web Consortium (W3C), Extensible Markup Language (XML), available at: http://www.w3.org/XML/, 2010-12-15. **11**. *Harold, E. R.; Means, W. S.* 2004, XML in a Nutshell, O´Reilly Media Inc., 2004. **12.** *Bodendorf, F.,* 2005, Daten- und Wissensmanagement, Springer [13] IBM: Advantages of XML; In : IBM iSeries Information Center-Version 5 Release 3 (V5R3), available at: http://publib.boulder.ibm.com /infocenter/iseries/v5r3/index.jsp?topic=%2Frzakl%2Frzaklintroadvantages. htm, 2010-12-15. **14.** *Chryssolouris, G., Makris, S., Xanthakis, V., Konstantinis, V.,* 2003, An XML based implementation of the Value Added Chain in Manufacturing: A Ship repair Case Study, In : CIRP Journal of Manufacturing Systems, (Vol. 32, No.6, 2003).