VOL. 2, NO. 5, May 2012 ISSN 2222-9833

ARPN Journal of Systems and Software


http://www.scientific-journals.org

# Middle-of-life PLM Solutions for Reconfigurable Networked Mechatronic Products

[1] Tommi Parkkila, [2] Jukka Kääriäinen, [3] Hannu Tanner and [4] Jukka Riekki
[1] VTT – Technical Research Centre of Finland, P.O. Box 1100, FI-90571 Oulu, Finland
[4] Department of Computer Science and Engineering, University of Oulu, P.O. Box 4500, FI-90014 University of Oulu, Finland
Email: {[1] tommi.parkkila@vtt.fi, [3] hannu.tanner@vtt.fi, [4] jukka.riekki@ee.oulu.fi}

## ABSTRACT

This paper argues for the importance of product configuration data management during the operational life phase of modern, software intensive, reconfigurable and networked mechatronic products. The state-of-practice of the configuration management of mechatronic products is presented first. The challenges in configuration management that the machine industry is facing are then identified empirically, by conducting semi-structured interviews within seven companies. The main challenges identified are the lack of evolving product unit-specific, as-maintained data and its synchronization with the configuration changes executed on the product in the field. Finally, product lifecycle management (PLM) systems requirements are specified based on the challenges, and commercial systems are evaluated against these requirements. The results indicate that the potential of remote connectivity between PLM systems and products in the field has been recognized by both the machine industry and PLM system vendors, although systems, technologies, and work processes are not yet mature.

**Keywords:** *Product lifecycle management, Configuration management, Reconfigurable computing, Mechatronics, Networked systems*

## 1. INTRODUCTION

The increasing use of embedded computing devices and software in modern mechatronic products has introduced new possibilities for intelligent functionality, remote diagnostics and maintenance. Even though this creates opportunities to meet new functional needs, safety regulations and maintainability requirements that have emerged from customers and governments, new requirements are set for the information management of these products throughout their lifecycle as well.

Product lifecycle can be divided roughly into three phases (Figure 1): Beginning-of-Life (BOL), Middle-of-Life (MOL), and End-of-Life (EOL). At the BOL phase products are designed, created and manufactured. MOL is the phase where the manufactured products are delivered to end users; products are used, maintained, and serviced by customers, an aftermarket service or a third-party service provider. At the EOL life phase products are retired; they are disassembled, reused, and disposed of. [1]

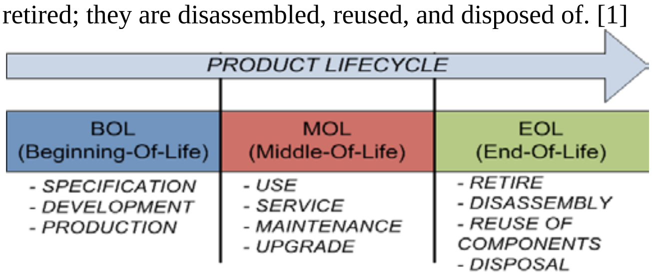


**Figure 1:** Product lifecycle

This paper focuses on presenting the state-of-practice of product configuration management in the machine building industry during the MOL phase; on identifying the need for such management empirically, by interviewing 17 experts from seven companies; and on evaluating commercial PLM system solutions against the identified needs. The results indicate a need for more integrated product data and configuration management systems through product development, manufacturing and aftermarket services, i.e. the BOL and MOL phases.

The next chapter familiarizes the reader with the industrial context, configuration management specialties in mechatronic products, the concept of closed-loop PLM, and the types of MOL PLM systems. The third chapter introduces the research process and the fourth chapter shows the result of the industry interviews and MOL PLM systems evaluations. The research results are then discussed and conclusions are drawn.

## 2. BACKGROUND AND INDUSTRIAL CONTEXT

### 2.1 Mechatronic Systems

The French standard NF E 01-010 [2] states that mechatronics is a multidisciplinary field of engineering, which integrates mechanics, electronics, control theory, and computer science within product design and manufacturing. Mechatronics products in the machine industry, such as excavators, loaders, mining machines, and air compressors are relatively low-volume products, where high number of product variants, together with long—up to 40-year lifecycles—has led to scattered production volumes [3]. System engineering has approached this issue with modular product architectures [4], where product variants are built through interchangeable modules and their configurations. At its best, the designed and built modules can be utilized among different products when the module becomes a standard component of a certain level for product platforms. Developing modules and platforms helps when managing product design, development and manufacturing, and helps in customizing a variety of complex products from simpler building components. [5] The design, manufacturing and maintenance of modularized systems are more efficient and can be subcontracted from a third

ARPN Journal of Systems and Software

http://www.scientific-journals.org

party [6]. Modular product architecture and design, and platform design methods are studied in detail by e.g. Jose &Tollenaere[5] and Dahmus et al. [4].

### 2.2 Configuration management of mechatronic systems

Because of their complex and multidisciplinary nature, the role of a well-utilized configuration management (CM) process is essential with mechatronic products [7]. CM can be defined as a set of tools and guidelines that apply technical and administrative control over the lifecycle of complex products. The purpose of CM is to manage and control the design, development, manufacturing and maintenance of products, and their variants, changes, corrections, extensions, and adaptations. [8] [9] CM supports efficient product development, decreases time to market, and helps organizations to have the information they need to ensure that their products perform as intended. Well-structured CM gives also detailed guidelines on how the products can be maintained, and how they can be customized according to customers' wishes. [10] [9] [11]

In the machine industry, configuration management has its roots in managing hardware-related design artifacts (CAD files) with specific product data management (PDM) systems. With mechatronic products, there is also a need for management guidelines and tools for both electronic design artifacts and software artifacts from requirements specifications to source codes, application parameters and test documents. All these three artifact groups – CAD files, electronic design files, and software artifacts – have their own data management systems, which makes the configuration management tasks challenging, especially regarding product modifications i.e. change management tasks at the product's MOL phase. Mechatronic systems require a whole system-wide product data management system. Product lifecycle management systems (PLMs) are seen as promising approach to fulfilling this requirement. [12]

Based on Brown [13], mechatronic products are becoming even more software oriented in the future. He found that the amount of software, and its importance in products, have increased significantly in over 50% of the companies during the past five years, and the same trend is continuing into the future as well. According to Brown, software enables manufacturers to integrate more intelligent functionalities into their products, and the more software-based functions exist, the more flexible and adaptable a product can be. This flexibility improves the possibilities of customizing products according to customer wishes, increases hardware reuse, and improves more agile product updates and upgrades in the field. These features are essential especially for low-volume mechatronic products.

Current digital hardware technology enables flexibility at the hardware level as well. Here, flexibility is achieved through configurable hardware technologies, such as field-programmable gate arrays (FPGAs) [14] [15]. Similar development can be seen in analog hardware technology as well, where reconfigurable mixed-signal platforms, field-programmable analog arrays (FPAAs), are pushing hardware flexibility to a new level [16]. The possibilities of reconfigurable hardware have been noticed in the machine industry sector as well. Saha[3] points out the potential of customization possibilities of configurable hardware technology in mobile control system electronics. The same hardware solution can serve more applications and products when flexibility is extended into the hardware logic level. This decreases the expenses of manufacturing and testing new computer boards and customizing software for new dedicated hardware implementations.

Saha[3] also clarifies the improvements that the reconfigurable hardware technology brings for the maintenance of control electronics of long lifecycle mechatronic products. So far, the main challenges have been ensuring the continuum of electronic hardware as long as possible and ensuring software compatibility with new hardware when a hardware upgrade is required. Reconfigurable hardware technology allows software and reconfigurable hardware design artifacts to remain stable and in synchronization, when physical semiconductor hardware is required to be upgraded during the long operational life. This is because reconfigurable hardware design, or the more specific hardware assignment layer of the design, can work as a configurable hardware abstraction layer for upper logical hardware design and software artifacts.

While enhancing the flexibility of a product, configurable hardware also increases the complexity of product configuration management. A new changeable layer needs to be managed alongside fixed hardware, firmware, and software [8]. The functionality of a reconfigurable hardware chip e.g. FPGA, depends on the hardware design file artifact, a bitstream file, developed through standard hardware description language (HDL), which is saved into the FPGA configuration memory. Therefore, configurable hardware development has similar late binding features as software has in electronic products in general [10]. Moreover, reconfigurable hardware designs can be updated to a remote product working in the field, similarly to software and configuration parameters as well [17] [18].

### 2.3 The concept of closed-loop PLM

The increasing networking capabilities of embedded computing systems bring remote machines within reach of services on Internet and dedicated industrial communication networks. The Internet of Things [19] [20] and Ubiquitous Computing [21] concepts, and increasing performance on networked embedded industrial systems have enabled remote application management, which is utilized in many products on the market already. For machine industry, where the products' operational life is relatively long, remote maintenance and management during the product's MOL phase (Figure 1) is an attractive and cost-saving possibility for aftermarket services. [3] Additionally, product-related information from the field about the executed services and configuration changes is valuable for a product's early development phases as well [22]. Kiritsis[1] and Terzi et al. [22] calls for product lifecycle management (PLM) systems to be used as the main information integrator throughout all the phases of

ARPN Journal of Systems and Software

http://www.scientific-journals.org

the product lifecycle. So far, this has not received much attention from industry due to the lack of efficient tools to gather product data into PLM systems [1].

Kiritsis[23] [1] has approached this challenge by establishing a concept of *closed-loop PLM*, where the product information is gathered from the field through product-embedded information devices (PEIDs) and integrated into PLM systems through a product knowledge and management system (PDKM). However, the concept does not give a clear view on closing the loop; how the collected product information is utilized in further configuration management of these devices. For example, how configuration parameter changes or software updates executed in the field can be managed and tracked from an individual product unit to the corresponding data structure in a PLM system. In addition, mechatronic systems, especially the control systems of mobile machines are typically highly safety critical, which raises safety and security into the highest priority issues.

### 2.4 Information systems for MOL PLM

The management of product lifecycle data (i.e. Product Lifecycle Management, PLM) is typically spread into several information management systems that are usually not interconnected. For instance, PDM (Product Data Management) systems are used to primarily manage product data produced at the development phase of the product, whereas ERP (Enterprise Resource Planning) systems focus on the management of product data at the manufacturing phase. According to Terzi et al. [22] PLM acts as a service support system for MOL product operations. These systems typically provide product data in conjunction with service management capabilities focusing on servicing existing assets and on the "*as-maintained*" product information [7]. The systems that are used in this phase of the lifecycle are called, for instance, CMMS (computerized maintenance management systems) [22] or service management solutions [7]. The increasing demands for after-sales services have invoked PDM tool vendors to upgrade their tool suites to support the MOL phase of the products [11] (e.g. the maintenance module of Teamcenter[24]). This same trend can be seen in ERP systems (e.g. the PLM module of SAP) [24]. Therefore, PLM systems for the MOL phase for managing the data of evolving products can be classified in three types as listed in Table 1. The table also presents examples of commercial products representing the classified PLM system types.

**Table 1:** MOL PLM system types

| ***Type:*** | *Maintenance/ service management systems, CMMS* | *MOL PLM system modules in PDM/PLM systems* | *MOL PLM system modules in ERP systems* |
|---|---|---|---|
| ***Commercial Products:*** | Idhammar[1], AfterSales[2], MPulse[3], C-Care[4] | Teamcenter[5], Aton[6], Windchill[7] | SAP PLM module[8] |

[1] http://www.idhammarsystems.com/
[2] http://www.absent.fi/en/After_Sales/
[3] http://www.mpulsesoftware.com/

From a technology point of view, PLM system vendors are adopting cloud technology. For instance, C-Care is a cloud based system available directly from the Internet (web-based) without any special software installations. From ERP providers SAP also provides ERP services from the cloud. Kim [25] presents an interesting discussion about the advantages and concerns related to cloud technology that companies should consider when moving towards cloud services.

## 3. RESEARCH PROCESS

This research is part of a wider research effort studying the dynamic configuration management of reconfigurable networked industrial systems. The research presented in this paper studies the need for the MOL product lifecycle management of mechatronic products in the context of machine industry. The aim of the study is to identify the existing PLM solutions for the MOL phase, to identify the requirements for such systems, and to evaluate how existing solutions meet the identified requirements. This information will be then used in next research step for the generation of requirements specification for the dynamic configuration management of mechatronic products in the MOL phase.

The wider research effort also includes evaluation of existing machine-to-machine (M2M) communication technologies against requirements that have arisen through the industry interviews and literature studies. M2M technology requirements and evaluations, as well as the identified requirements for the control electronics of mechatronic products, are outside the scope of this paper. Figure 2 presents the overall research process. The focus of this paper is surrounded with a dotted line.

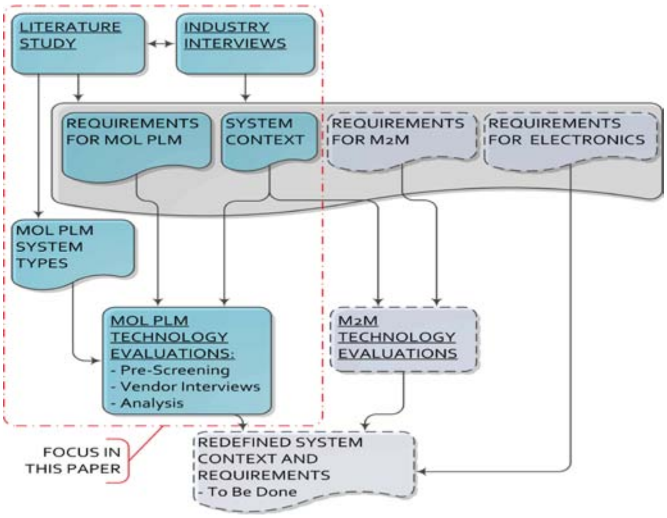


**Figure 2:** The overall research process and the focus of this paper.

[4] http://www.cieltum.com/fi/in-english/products.html
[5] http://www.plm.automation.siemens.com/en_us/products/teamcenter/index.shtml?stc=fiiia400111
[6] http://www.modultek.com/fi/ratkaisut/installed-base-management.html
[7] http://www.ptc.com/solutions/windchill-10/
[8] http://www.sap.com/solutions/business-suite/plm/index.epx

ARPN Journal of Systems and Software


http://www.scientific-journals.org

The research started with a literature study resulting in a list of MOL PLM system types and a background study about the MOL phase configuration management in a machine building industry context. Then 17 experts from seven organizations in the machine industry from Finland and the USA were interviewed (semi-structuredinterviews).The interviewees were asked about their organization in general,their product structures and features, the configuration management state-of-practice applied to products, the remote connectivity of products and the aftermarket services that the products have during their MOL phase.

**Table 2:** Summary of the companies interviewed

| *Company name* | *# of Persons Interviewed* | *Category of Products* | *Role of Company* | *Applied Computing Technologies* | *System Architecture* | *Remote Connectivity* |
|---|---|---|---|---|---|---|
| **C1** | 8 | Mechatronic Machines | OEM | Embedded PC , Microprocessor, PLC, CHW | Distributed | Yes |
| **C2** | 2 | Mechatronic Machines | OEM | Embedded PC, Microprocessor, PLC | Distributed | Yes |
| **C3** | 1 | Mechatronic Machines | OEM | MCU | Single | No |
| **C4** | 1 | Machine Condition Monitoring Systems | OEM | Embedded PC, Microprocessor | Single | Yes |
| **C5** | 2 | Machine Condition Monitoring Systems | OEM | Embedded PC, Microprocessor, CHW | Single | Yes |
| **C6** | 1 | Control Electronics | SUB | Embedded PC, Microprocessor | Both | Yes |
| **C7** | 2 | Control Electronics | SUB | Embedded PC, Microprocessor, PLC, CHW | Both | Yes |

The system context description and requirements for the MOL PLM were created based on the results of the interviews and literature studies. These introduce the problem field and challenges which MOL PLM solutions should address. During the MOL PLM evaluation phase existing commercial MOL PLM systems were pre-screened based on material found from vendors' web-pages with a view to finding systems that represent different MOL PLM system types (Table 1). Based on pre-screening, three vendors were selected for vendor interviews. Interviews were semi-structured using context description and preliminary requirements as a frame for interviews and analysis. All these interviews were tape-recorded and transcribed.

# 4. RESULTS

## 4.1 Industry Interviews

Table 2 summarizes the machine industry companies interviewed. Three of the interviewed companies (*C1-C3*) are original equipment manufacturers (OEMs) of mechatronic machines, and two (*C4, C5*) are vendors of machine condition monitoring products, which can be used as independent monitoring units or as sub-units e.g. diagnostic modules of control systems. The final two (*C6, C7*) are control electronics subcontractors (SUB) for OEMs. The next paragraphs summarize the industry interviews.

*Question 1: What control electronics are applied to the products?*

All the interviewed companies, except *C3*, are using dedicated microprocessor-based embedded controllers or embedded PC platforms, such as PC-104, in their products (Table 2, 5$^{th}$ column). Company *C3* uses microcontroller-based single controller architecture. Programmable logic controllers (PLCs) are widely used as well, especially in distributed control systems, where they work as slave controllers for executing sensing and real-time control next to actuators. Configurable hardware (CHW) CPLDs and FPGAs are also utilized in controllers, mainly as reconfigurable IO-interfaces, but also for real-time signal processing.

*Question 2: How are product design data, hardware and software, and configuration data managed at the BOL phase?*

At the BOL phase of the product lifecycle, all the interviewed experts stated that their companies utilize PDM information systems in managing product hardware-related data; CAD-designs, Bill-Of-Materials (BOMs), etc. Uniquely identified hardware components, and their design artifacts and relations build a fine-grained structured product data tree, which is stored into a PDM system. The built product data tree works as a foundation, a *neutral product data structure*, for the actual products to be sold, built, and delivered. Many of the interviewed experts stated that software design artifacts and source codes are managed with separated software versioning and revision control tools, e.g. SVN, or more sophisticated ALM (Application Lifecycle Management) systems. An expert from company C1 specified: "The *software of a controller is linked to the neutral product data structure after it is released and compiled into a binary file*." The binary file is combined with its default boot-time configuration parameter file into a single software installation package, which is then referenced as a single configuration item number and corresponding file path in

VOL. 2, NO. 5, May 2012 ISSN 2222-9833

ARPN Journal of Systems and Software


http://www.scientific-journals.org

the neutral product data structure in the PDM system. According to an expert from company C1: "*The PDM has a software identification number, which references all corresponding software files and parameters that are stored into a network drive*."

Two of the mechatronic machine OEMs (*C1* and *C2*) manage their products' complexity by applying modular distributed control system architecture (Table 2, 6th column). Product modularity expands into controller level hardware and software architectures as well. The product controllers are composed from two to almost ten separated sub-modules, PCB-boards with a specific interface or computation functions. For example, experts from companies C4 and C5 stated that their products have separate PCB-board modules for every communication interface applied. Controller software is divided into several specific layers, as a firmware, a hardware abstraction layer (HAL), an operating system (OS), and an application layer. Company *C7* also utilizes a virtual machine layer between applications and OS-layers. The virtual machine enables fine-grained software modularity by allowing function-based modularity in the application layer.An expert from company C7 said that the virtual machine approach enables: "*Compound functions to be broken down into manageable components with their own part numbers*." All the interviewed experts noted that companies modularize software through options, which are manageable with boot-time configuration parameters. For instance, an expert from company C1 said: "*Basic machines have about 10 features, which can be started and stopped through software options*." These options are linked to the uniquely identified features of a product that the manufacturer can sell as aftermarket updates without a need to recompile and reprogram the software.

*Question 3: How is product-specific configuration made and where it is stored?*

Modularity in hardware and software designs and implementations allows the utilization of subcontractors. Experts from companies C1-C3 clarified that they use subcontractors for implementing the hardware modules and special software modules of the control system electronics or the control system as a whole. "*80% of products are basic machines, which have five pieces of company X modules*," explained one of the interviewed experts. "*We define how the modules should work, and company X programs them, then we test them.*" Additionally, modularity in products enables their easier customization for customers. Experts from companies C1, C2, C6 and C7 stated that they use specific configuration generators as a supportive tool for product sales, where the product is built virtually to meet customer needs. According to an industry expert from company C6, they have a: "*Configuration generator for mass-customizable products, whereby a salesman can customize… it has a rulebase. With the help of such a product, the selling skills of the best salesman in a company can be utilized by other salesmen*," he added. Based on the features that a customer has ordered, a configuration generator builds a product unit specific data structure from the neutral product data structure, and activates the features in controller software by enabling the corresponding sets of boot-time parameters in the configuration file. This *as-built* product data structure and its configuration file is stored in the PDM system and/or ERP system for managing the production phase and for planning resource and component needs during manufacturing. User manuals, spare part lists, and service instructions for aftermarket are generated based on the as-built product data structure as well.

Before the manufactured product is delivered to the end user, a software installation package *(software and configuration file*) is programmed into the controller or controllers and the activated configuration parameters are tuned for the end application. One of the interviewed companies stores these parameter changes to the specific product unit as-built data structure in the corresponding PDM or ERP system, but the rest let the updates affect only the local configuration file in the products. "*After parameter tuning, parameters are stored into a network drive with a reference to the product identification number… software image is also stored in the same place*," explained an expert from company C1.

*Question 4: How do product aftermarket services utilize product-specific data structures?*

The data and configuration management processes in the BOL phase of a product lifecycle are well established within the interviewed companies and as-built product-specific data structures exist. This is especially the case with the management of product hardware items, but when there is a need to update product-specific data structures after the product manufacturing phase, work processes and tools seems to be missing from the most of the interviewed companies. "*The system is able to store service information on installed products, but there is a lack of organized documentation of the service process*," specified an expert from company C2. Only company C7 has its own specific data management system for maintaining product-specific, *as-maintained* data structures.

Companies, except for the company C3, do remote monitoring and diagnostics for products in the field (Table 2, far-right column), but the diagnostic information is primarily used for aftermarket services to see how products are currently working in the field. An expert from company C1 stated: "*5% of computer-controlled machines are remotely operated. At the moment parameter changes are not done through Wi-Fi connection in the field, there are no technical limits, maybe it is a political question or changes are needed in the work processes*," he continued.Because of the missing product-specific, as-maintained data structure and required information links of the services executed in the field, product management at the MOL phase of the product lifecycle is challenging. *"As-built product data instances stored into a network-drive are not updated, changes in product data structure are only done for the local data structure stored in products*," explained an expert from company C1. He emphasized, "*Information on configuration changes executed in the field are not delivered to the factory*."

**ARPN Journal of Systems and Software**

http://www.scientific-journals.org

*Question 5: What are the challenges in managing the configuration changes of a product in the field?*

The interviewed experts from companies C1, C2, C3 and C7 noted that shortages in configuration data replication cause several disadvantages for product aftermarket services and product warranty programs as well. It is problematic especially, when a service issue with a product in the field needs to be solved at a higher level of the manufacturer's aftermarket sector and maybe in co-operation with R&D specialists. Decisions about the required service actions or next configuration changes are challenging to make when there is no available history data about the executed configuration changes. Therefore, prior to any analysis of the issue, "*The customer or local service need to send information about the current parameter setups*," one of the interviewed experts from company C1 stated. Moreover, "*Aftermarket providers would need information on the status of the product before the detected fault or problem… they need the log files of parameter changes in conjunction with diagnostic data*." Thus, up-to-date configuration data has to be gathered from the field, which might be expensive and time-consuming extra work, especially when there are no communication systems between products and data management systems. According to an expert from company C2: "*Beyond customer service calls... a warranty program is the only way to get information about installed products*."

The application software update process was also felt to be problematic in practice, mostly for the same reason. New software releases are taken into use automatically in new products, but for a deployed product an upgrade is done only when it is absolutely mandatory, e.g. in the case of a safety-related software update. "*Production starts to use new software versions immediately; however aftermarket service providers define for each product whether the update is necessary or not*," explained an expert from company C1. Updating is problematic especially when software is delivered as an installation package with a configuration file. Experts from companies C1, C2, and C7 noted that if the configuration file downloaded from the product-specific data structure is not up-to-date, the update may overwrite the tuned parameter values with old or factory-default values and the control system can become unstable.

Updating firmware software and upgrading configurable hardware design (e.g. FPGA configuration) are prohibited in the field at the moment. Experts from companies C1 and C7 stated that both of these artifacts are fixed to controller hardware item numbers, and updating/upgrading the artifact causes a change of the configuration item number of the controller. This leads to a mismatch in static configuration management data. Management of software and configuration files gets even more difficult when the target for the service or change request is the product of a subcontractor. An expert from company C1 suggested a solution, which could have a significant effect on the performance of their aftersales work: "*A simple model would be good, where every firmware change executed for third-party component is logged into a file.*" The interviewed experts identified a collaboration network of manufacturers and subcontractors for product data management, in both the BOL and MOL phase, as an important research topic in the future.

The interviewed companies were willing to see more comprehensive utilization of wireless communication technologies in product management. According to an expert from company C3: "*There is a huge market potential for getting configuration management information exchanged with remotely located devices through short-range communication, with a mobile user interface.*" Remote connectivity is challenging when the working environment of a product is underground or if it has strict regulations for remote connectivity, as for example in nuclear plants. "*Problems arise when end users wants closed systems where is no straight information link from a global network, like power plants. There needs to be a way to manage the data exchange between management information systems and products through other medias also, for example utilizing hand-held terminals and short-range wirelesses,*" explained an expert from company C2. Therefore, the remote communication and information data exchange between products and information management systems should be able to exploit alternative communication systems and technologies.

As a summary of the results, one can say the interviewed experts identified the need for the traceability of configuration management actions executed on products in the field. They require that the configuration change history of products should be kept up-to-date at both ends of the configuration management system – in products in the field and in PDM/PLM systems. In addition the integration of product-specific diagnostic data history and configuration management data was seen as valuable for both aftermarket and R&D professionals. Configuration management actions should be able to be executed through remote and local connections, and there should be mechanisms and processes for third-party subcontractors to log information to product unit-specific data structure about executed services and configuration changes.

### 4.2 System context and requirements

The system context for the MOL PLM was specified based on the findings of the interviews. Requirements for the MOL PLM systems were also specified to tackle the identified challenges. Figure 3 presents the system context, with requirements also placed in this context. The figure contains actors operating in the machine-building industry context. The MOL PLM system stores and manages evolving (*as-maintained*) product unit-specific data throughout a product's operational life in the field. Evolving product unit-specific data structures are created automatically from a product neutral data structure in the BOL PLM system. The data structure can be modified by the manufacturer's R&D team, production and aftermarket services, but can also be viewed and modified by subcontractors in a collaborative environment. A subcontractor can maintain a subpart of the mechatronic

ARPN Journal of Systems and Software

http://www.scientific-journals.org

product data, for which the subcontractor is responsible. Products in the field can have direct online access from the field to the product-specific data structure, and also mechanisms for synchronizing configuration and other product-specific data through a mobile terminal, or through other products nearby, utilizing a short-range radio technology.

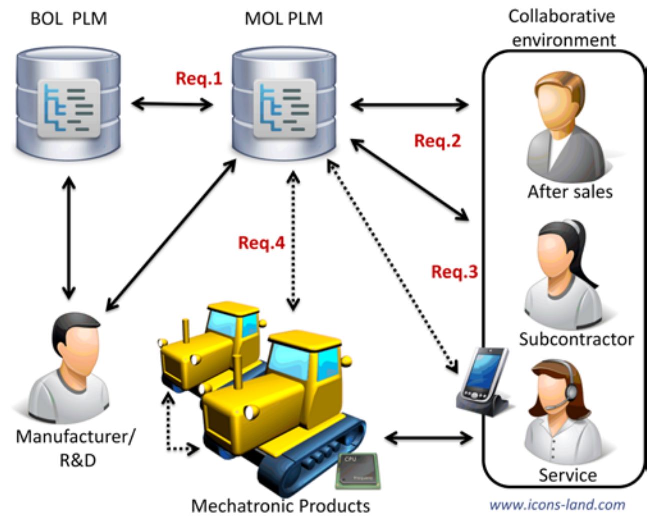


**Figure 3:**System context

The requirements listed in the Table 3summarize the challenges that have arisen for MOL phase PLM systems in the configuration system context. These requirements were used as a guideline for commercial MOL PLM system evaluations and the vendor interviews.

**Table 3:**Requirements for MOL PLM systems

| |
|---|
| *Req.1. Data Management* |
| The evolving product unit-specific data structure and related data can be created and transferred automatically from the BOL PLM system to the MOL PLM system. The MOL PLM system also stores the configuration change history of the product unit. |
| *Req.2. Multi-organizational access to the MOL PLM database* |
| Multiple organizations can access the product unit data in a collaborative environment. |
| *Req.3. Connection to the MOL PLM system with a mobile device* |
| 2-way secure communication via mobile device: Data related to a particular product unit or structure can be accessed and edited via a mobile device. |
| *Req.4. Wireless connection between smart devices and the MOL PLM* |
| 2-way secure wireless communication between the MOL PLM system and a smart product: Product configuration data can be kept in sync. |

## 4.3 MOL PLM evaluation results

Based on pre-screening, three MOL PLM systems were selected for vendor interviews. The target was to select systems that would potentially be used by the participating industrial companies, that is, systems that are mature enough for industrial use and that were mentioned during the industrial interviews as potential systems. In order to gain insight on what PLM systems can currently offer, the intention was to select systems that would represent different approaches to the MOL PLM. The systems selected for further study are presented in Table 4.

**Table 4:** MOL PLM systems selected for vendor interviews

| | *Type* | *Description* |
|---|---|---|
| ***System 1*** | *MOL PLM system (part of a PDM / PLM system)* | *Mature PDM/PLM system with a long history. Extended its capabilities towards MOL functionality (introduction of a MOL PLM module).* |
| ***System 2*** | *Maintenance/ service management system (CMMS)* | *Modular CMMS system especially for MOL PLM purposes. Vendor has also a strong competence for mobile application development.* |
| ***System 3*** | *Maintenance/ service management system (CMMS)* | *Novel CMMS system provided as a cloud service. System is especially targeted for the MOL PLM.* |

The three MOL PLM systems were evaluated against the MOL PLM requirements (see Table 3). The evaluation was done based on information received from the vendor interviews as well as information available on their web pages. System 1 represents a solution that has long history in the management of BOL product data. The system has been extended towards the MOL phase. Systems 2 and 3 represent CMMS systems. System 3 is quite new solution relying on cloud technology.

Regarding Req.1 (Data Management), in general, the systems seem to have a strong support for the required MOL PLM data structures. The data structures may contain all the relevant item data as well as related documentation. However, it seems that in some cases the problem is the missing as-maintained structure (i.e. service structure). For instance, a System 2 vendor said, "*Some companies do not even have a service structure in place…*" Furthermore, a System 3 vendor stated that service structures, "*Can be very different and unique in each company*." Therefore, system vendors also provide consultancy services to help the transition from BOL structures to MOL structures. The SW part of the system is typically stored as a SW package (binaries) into the MOL PLM system. A detailed structure of the SW (source code) is stored into the SW CM system and the SW package can reference the SW baseline stored in the software configuration management (SCM) system. Change histories related to the product units, changes and activities are collected and stored. Data structures can be created within the system or imported using BOL data, typically in XML format. All evaluated systems appear to provide strong support for usage over multiple organizations as well (Req.2). User privileges can be

ARPN Journal of Systems and Software

http://www.scientific-journals.org

managed and users can be provided with access to the information and structures of individual product units.

All interviews indicated that the systems seem to provide at least some level of support for connection to the MOL PLM system with a mobile device (Req.3). Web-based systems are potentially more usable, as most mobile devices currently have built-in browsers that are optimized for mobile use. However, typically the systems are currently not widely used via mobile devices, which may cause problems due to limited data-processing capabilities as well as screen sizes. The System 1 vendor indicated that currently mobile solutions can be used for instance to, "*Participate in workflows, browse structure and items, create/participate engineering changes, browse 3D models, etc.*" Similarly the System 2 vendor has developed a mobile solution that can be used for various MOL PLM activities. Based on interviews, mobility was seen as a trend for the future. All vendors stated that the needs for mobile connectivity will increase and vendors will expand their mobile MOL PLM solutions in the future.

Wireless connection between smart devices and the MOL PLM (Req.4) seems to have limited support currently. While there are some implementations and experiments with e.g., storing diagnostics data from the product to the MOL PLM system (e.g. System 2), the possibilities of two-way wireless connections are still being investigated. This was seen as having huge potential for the future, as long as issues with security and safety can be managed.

## 5. DISCUSSION

MOL PLM activities can be supported by different types of systems. Traditionally CMMS systems have been used to coordinate and plan maintenance work and product unit changes. Nowadays, PLM/PDM and ERP providers have started to expand their solutions towards the MOL PLM [24]. It is interesting to see how they succeed in getting their own share from MOL PLM markets that have been dominated by CMMS providers. Recently, the use of cloud technology has also enabled smaller companies to deploy complex information management systems. In these systems the third-party cloud service providers manage the computing resources. Therefore, the users do not need to make a large upfront investment on computing resources, IT staff, network, and database [25]. It has been forecast that in the future the amount of cloud services will increase [25].

According to Svensson[26] different operations (e.g. sales, design, manufacturing and after sales) have their own needs for product structure and product information. After-sales services operate with product structures that are transformed from BOL-phase product structures [22] (as-maintained structure). Svensson[26] calls these structures as *maintenance structures*. The results of our study show that in some cases these maintenance structures might be missing, or that transitioning from BOL structures to MOL structures is challenging and results in problems to the maintenance of product-related services and structure information. Therefore, MOL PLM system vendors provide consultancy services to handle this transition since an accurate product structure with related service documents is an essential pre-requisite for effective after-sales operations. Terzi et al. [22] argue that currently there is an information gap between the BOL and MOL phases. They believe that data transition should happen, not only from BOL to MOL, but in both directions, also enabling producers and designers to benefit from information collected from the users and after-sales providers. This unification of information management through all phases of the product lifecycle (from BOL to MOL to EOL) currently seems to remain more of a theory than a practice in industry.

The industry and PLM interviews indicated that SW is typically treated as a whole or as a large chunk of a mechatronic product's structures in MOL PLM systems. More fine-grained management of the software happens in dedicated systems called software configuration management systems (SCM). A SW package stored into an MOL PLM system has reference to the SW baseline (traceability of files and versions) stored into an SCM system (cf. Crnkovic et al., [9]; Krikhaar et al., [10]). However, HW-SW interdependencies (especially interdependencies between SW, configurable hardware designs, e.g. FPGA, and hardware electronics) and parameterization in mechatronic products require more fine-grained management of interdependencies between HW and SW components and their related product features. The unified configuration management of SW and HW portions of the product seems to remain an issue in some cases. This is surprising since SCM-PDM/PLM system integrations exist and the topic of the unified management of multidisciplinary products was studied widely a decade ago (see e.g. Crnkovic et al. [9], PerssonDahlqvist et al. [27], Estublier et al. [28] and Conradi &Westfechtel[29]). Recently Krikhaar et al. [10] have published interesting research that aims to understand more deeply the evolution of integrated hardware/software systems and discusses configuration management on the hardware/software boundary. They argue that even though SCM-PDM integrations exist, their deployment might be prevented by the lack of required functionality, or the industry is just slow in adopting them.

Based on the results, it seems that one of the future trends for PLM systems could be enabling mobile, collaborative use. Distributed, multi-organizational development practices and tools have already been studied in the industry, and the benefits and success factors reported (e.g. [30] [31]). As more and more modern machinery and equipment are being enhanced with software features, the Internet is also becoming ubiquitous. According to Brown [32] and Kiritsis[23] PLM systems can benefit from mobile technology e.g. by improving service response time since up-to-date product information is readily available in the field when it is needed. Moreover, information on the executed services and configuration changes in the field can be transmitted to PLM systems through mobile technology for keeping product-specific data up-to-date and improving aftermarket processes. The need for more comprehensive

ARPN Journal of Systems and Software




utilization of mobile communication technology for the configuration management of products was also identified by the companies interviewed. While the potential of portable devices—e.g. tablet computers—has been noticed by the system vendors, all the possibilities have not been yet identified. Moreover, direct connectivity between PLM systems and smart products in the field would appear to provide a new kind of functionality: remote diagnostics, firmware and software upgrades as well as more advanced communication open up new opportunities. While the potential of such features is highly promising, the risks involved—e.g. security issues or losing track of configurations—are also significant meaning that further research and empirical studies are still needed.

## 6. CONCLUSION

This paper identifies four main requirements (see Table 3) for product data management systems to support the configuration management of reconfigurable networked mechatronic products in the field. The requirements were identified through literature studies and industry interviews, in which 17 persons from seven organizations (see Table 2) were interviewed. Interviews demonstrated that configuration management is well utilized in the product development and manufacturing phases. Configuration management challenges appear when products are delivered to customers and there is a need to do configuration updates to products in the field. Product unit-specific, as-maintained data structures are either missing or they are not kept up-to-date according to configuration changes in the field. Configuration parameters and application software changes in the field are problematic, and firmware, software and configurable hardware design updates in the field are prohibited due to the lack of flexibility of applied hardware based configuration management practices. The interviewed companies pointed out the need for the traceability of configuration management actions executed to products in the field, the configuration change history of products should be kept up-to-date in products in the field and in PDM/PLM systems.

Three commercial MOL PLM systems were evaluated against the created configuration management system context (Figure 3) and identified requirements (Table 3). The evaluated systems have strong support for MOL PLM data structures, although software is typically stored in large compiled binary packages. A more fine-grained management of embedded software was seen as a desirable feature in MOL PLM solutions. Product data structures can be created within the system or they can be imported via specific data interfaces. The systems maintain product change history and they all have strong support for multi-organizational access management. All evaluated systems seem to provide some level of mobile connectivity, but it is not widely supported. However, the system vendors identified huge potential for PLM system features supporting mobile communication technologies and straight product remote connectivity. This potential was also recognized by the interviewed companies.

## ACKNOWLEDGEMENTS

This work was carried out as a part of ProMoNet–the Product Life-Time Configuration Management of Networked Industrial Systems joint research project. We would like to acknowledge the industry partners and TEKES – Finnish Funding Agency for Technology and Innovations for supporting this ProMoNet project in the Digital Product Process research program. We wish to give special thanks to all interviewed machine industry professionals and PLM system vendors.

## REFERENCES

[1] D. Kiritsis, “Product Lifecycle Management and Embedded Information Devices,” in *Handbook of Automation*, Springer, 2009, pp. 749-765.

[2] *Standard NF E 01-010, Mechatronics - Vocabulary*, France: Association Française de Normalisation (AFNOR), 2008.

[3] H. Saha, Some Methods to Improve Life Cycle Manageability of Low-Volume Mobile Processing Platforms, Tampere: Tampere University of Technology, 2010.

[4] J. Dahmus, J. Gonzalez-Zugasti and K. Otto, “Modular product architecture,” *Design Studies*, vol. 22, no. 5, pp. 409-424, 2001.

[5] A. Jose and M. Tollenaere, “Modular and platform methods for product family design: literature analysis,” *Journal of Intelligent Manufacturing*, vol. 16, no. 3, pp. 371-390, 2005.

[6] A. Sangiovanni-Vincentelli, L. Carloni, F. De Bernardinis and M. Sgroi, “Benefits and challenges for platform-based design,” in *Proceedings of the 41st annual Design Automation Conference*, San Diego, 2004.

[7] Aberdeen Group, “The Configuration Management Benchmark Report: Formalizing and extending CM to drive quality,” Aberdeen Group, Boston, MA, USA, 2007.

[8] J. Buckley, Implementing Configuration Management: Hardware, Software, and Firmware, Los Alamitos, CA, USA: IEEE Computer Society Press, 1996.

[9] I. Crnkovic, U. Asklund and A. Dahlqvist, Implementing and Integrating Product Data Management and Software Configuration Management, London: Artech House, 2003.

[10] R. Krikhaar, W. Mosterman, N. Veerman and C. Verhoef, “Enabling System Evolution Through Configuration Management on the Hardware/Software Boundary,” *Systems Engineering* , vol. 12, no. 3, pp. 233-264, 2009.

[11] A. Sääksvuori and A. Immonen, Product lifecycle management, Berlin: Springer, 2008.

[12] J. Brown, “Complexity and Opportunity in the Mechatronic Era,” Tech-Clarity Inc., Media, PA, USA, 2011.

[13] J. Brown, “Addressing the Innovation – Complexity Conundrum,” Tech-Clarity Inc., Media, PA, USA, 2012.

[14] S. Hauck and A. DeHon, Reconfigurable Computing: The Theary and Practice of FPGA-Based Computation (Systems on Silicon), Burlington, MA, USA: Morgan Kaufmann, 2007.

[15] K. Compton and S. Hauck, “Reconfigurable computing: a survey of systems and software,” *ACM Computing Surveys (CSUR)*, vol. 34, no. 2, pp. 171-210, 2002.

[16] R. Selow, H. Lopes and C. Lima, “A comparison of FPGA

**ARPN Journal of Systems and Software**


http://www.scientific-journals.org

and FPAA technologies for a signal processing application," in *International Conference on Field Programmable Logic and Applications*, Prague, Czech Rebublic, 2009.

[17] J. Castillo, P. Huerta and J. Martinez, "Secure IP downloading for SRAM FPGAs," *Microprocessors and Microsystems,* vol. 31, no. 2, pp. 77-86, 2007.

[18] S. Drimer and M. Kuhn, "A Protocol for Secure Remote Updates of FPGA Configurations," *Lecture Notes in Computer Science 2009,* vol. 5453, pp. 50-61, 7 March 2009.

[19] J. Conti, "The Internet of things," *Communications Engineer,* vol. 4, no. 6, pp. 20-25, 2006.

[20] N. Gershenfeld, R. Krikorian and D. Cohen, "The Internet of things," *Scientific American,* pp. 76-81, 1 October 2004.

[21] M. Weiser, "Some computer science issues in ubiquitous computing," *Communications of the ACM - Special issue on computer augmented environments: back to real world,* vol. 36, no. 7, pp. 75-84, 1993.

[22] S. Terzi, A. Bouras, D. Dutta, M. Garetti and D. Kiritsis, "Product lifecycle management – from its history to its new role," *Int. J. Product Lifecycle Management,* vol. 4, no. 4, pp. 360-389, 2010.

[23] D. Kiritsis, "Closed-loop PLM for intelligent products in the era of the Internet of things," *Computer-Aided Design,* vol. 43, no. 5, pp. 479-501, 2011.

[24] H. Cao, P. Folan, J. Mascolo and J. Browne, "RFID in product lifecycle management: a case in the automotive industry," *International Journal of Computer Integrated Manufacturing,* vol. 22, no. 7, 2009.

[25] W. Kim, "Cloud Computing: Status and Prognosis," *Journal of Object Technology,* vol. 8, no. 1, pp. 65-72, 2009.

[26] D. Svensson, Towards Product Structure Management in Heterogeneous Environments, Göteborg: Chalmers University of Technology, 2003.

[27] A. Persson-Dahlqvist, I. Crnkovic and M. Larsson, "Managing Complex Systems - Challenges for PDM and SCM," in *Proceedings of ICSE 2001*, Toronto, Canada, 2001.

[28] J. Estublier, J. Favre and P. Morat, "Toward SCM/PDM Integration?," in *Proceedings of SCM8*, Bruxelles, Belgium, 1998.

[29] R. Conradi and B. Westfechtel, "Software Configuration Management and Engineering Data Management: Differences and Similarities," in *Proceedings of SCM8*, Bruxelles, Belgium, 1998.

[30] A. Välimäki, Pattern Language for Project Management in Global Software Development, Tampere: Tampere University of Technology, 2011.

[31] J. M. Bhat, M. Gupta and N. Murthy, "Overcoming Requirements Engineering Challenges: Lessons from Offshore Outsourcing," *IEEE Software,* vol. 23, no. 5, pp. 38-44, 2006.

[32] J. Brown, "Service Lifecycle Management Goes Mobile," Tech-Clarity, Inc., 2004.