Interaction within Dynamic IPS² Networks – A Proposal of an IPS² Lifecycle Management and IPS² Delivery Management Architecture

Thomas M. Dorka, Hoang Bao Dang, Horst Meier, Michael Abramovic

Abstract

In the delivery of Industrial Product-Service Systems (IPS²), a dynamic network of partners is involved. To deliver and use IPS² efficiently, the IPS² has to be managed from several heterogeneous perspectives, namely from a strategic, inter-organizational, operational and information perspective. To manage the information exchange between the partners, an architecture that supports the interaction between the perspectives and their corresponding software systems is needed. This paper introduces the different management perspectives of IPS² and their interplay. For the information exchange, an architectural approach is presented and implemented within a conceptual scenario situated in the micro milling industry. Especially the web service-based interaction between an IPS²-Lifecycle Management System (IPS²-LMS) and an IPS²-Execution System (IPS²-ES) is described in detail.

© 2014 Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/).

Keywords: Industrial Product-Service Systems (IPS²); IPS²-Lifecycle Management; IPS²-Execution System

1. Introduction

Industrial Product-Service Systems (IPS²) are new customer-specific solutions for business-to-business markets and can be viewed as an integrated bundle of key industrial goods as well as corresponding services and software [1]. In order to develop, provide and use such solutions, cross-linked information related to all domains (e.g. mechanics, electronics, service, software) is needed. Furthermore, different partners within the IPS² networks are involved throughout the lifecycle of the IPS². Some of them may be part of the partner network right from the start while others join in at later points [2]. This poses the challenge to exchange and manage IPS²-related information. All partners have to be supplied with relevant information and their systems have to be integrated accordingly, so that information can be managed, gathered and exchanged efficiently within a dynamic IPS² network.

During the IPS² delivery, the IPS² provider has to manage the dynamic IPS² delivery networks and execute strategic capacity as well as operational resource planning. By doing so, the resources of the partners in the network organization are assigned to the different tasks needed for the provision of the IPS². Whenever a partner enters or leaves the network, these changes have to be reflected in the delivery plan. Furthermore, newly arising demands during the operation of IPS² require an immediate reaction by the IPS² provider. These demands mainly consist of two categories: On the one hand, unplanned demands can occur, e.g. caused by malfunctioning components that require the scheduling of a repair process. On the other hand, planned demands are induced by scheduled IPS² solution alterations that may be
caused by either varying customer needs, a modified business model or a changed delivery environment.

Mastering the aforementioned requirements and challenges is a key success factor in order to deliver complex and customer-specific IPS² solutions. It enables the IPS² supplier to provide the customer value that has been defined in the stipulated business model.

This article describes the IPS² management from different perspectives and presents the interplay of some of the key concepts supporting the IPS² supplier in managing the IPS² throughout the lifecycle. More specifically, the IPS² lifecycle management has been developed to manage and integrate domain-specific information, offering a central information platform for different partners of the IPS² network. It provides IPS²-relevant information (e.g. regarding requirements for service delivery) to the central IPS²-Execution System (IPS²-ES), which organizes and plans the resources needed in the service delivery for all IPS². In return, the IPS²-ES informs the lifecycle management system about the allocation of resources (e.g. employees and tools assigned to a specific service delivery). This information enables the lifecycle management system to provide task-specific information to the employee. A web service-based interface enables an effective mutual delivery planning, IPS² network management and an integrated performance measurement method. "An IPS²-Execution System is characterized by the necessary data to perform the process scheduling [10]. It is needed to determine which network partners have to execute which delivery processes [9]. As the planning cannot be performed manually, a software system is needed to support the IPS² provider. This system is called IPS²-Execution System (IPS²-ES) and connects all the partners’ software systems to receive the necessary data to perform the process scheduling [10]. It is defined by [11] as follows:

“An IPS²-Execution System is the essential software system for the IPS² operation phase that supports the IPS² provider in the provision of customer value by adaptive IPS² delivery planning, IPS² network management and an integrated performance measurement method.”

Throughout the lifecycle of IPS², the data of virtual and real PSS from the IPS² provider and also from cooperation partners as well as from the customer need to be managed. This IPS²-related information has to be available for all actors within the IPS² network in order to plan, develop, use or provide IPS².

Over the last decade, Product Lifecycle Management (PLM) has become the central management approach of engineering processes and data, and it has been used as a company-wide integration platform. Current PLM solutions are mainly focusing on the management of goods-related data and engineering processes within the development phase. Different research is still done to extend the PLM approach to cover after-sale phases like the closed loop PLM approach [14], [15], [16] but without taking into account the management of service components and the interplay aspects between goods and services. In order to integrate the engineering of goods and services and to manage all IPS²-related data throughout the lifecycle, an overall lifecycle management approach for IPS² has been developed [17] and...
prototypically implemented [18]. Besides merely handling product-and-service-related data and processes, the PLM approach for IPS² (IPS²-LM) also considers the interdependencies of information and communication between all of the partners and customers involved in the IPS² lifecycle.

3. Different Management Perspectives on IPS² with an overarching IT landscape architecture

IPS² can be described as having four key management perspectives (see Fig. 2): At first, the strategic perspective refers to the business model which has been agreed upon by the IPS² provider and the customer. It includes the prevailing aims of the IPS², e.g. the guaranteed availability or provision of a specific function. From the inter-organizational perspective, the delivery of the IPS² is planned. It involves the complex task of managing the resources of all network partners (e.g. workers, tools, etc.). In order to ensure a reliable IPS² delivery, the required resources have to be planned in advance, distributing them at the right time, at the right place and of the right quality. In order to ensure a smooth and efficient planning of an IPS² delivery, which also has to take the aims of the IPS² into account, real-time information management within the IPS² network is essential. Communication between the network partners needs to be managed on an easily accessible common platform, which allows real-time reaction to the highly dynamic changes occurring during the IPS² lifecycle. Finally, the operational perspective refers to the actual delivery of the IPS², as planned by the inter-organizational perspective, governed by the strategic aims, and provided with the needed knowledge from the information perspective. This brief description already demonstrates that all of the IPS² key perspectives are interrelated and closely linked with each other. Due to the dynamic and complex nature of the IPS², agent systems supporting the monitoring and controlling of the IPS² delivery (operational perspective) are essential for an efficient management of the IPS².

However, not all reactions required to ensure a reliable IPS² delivery can be taken onto the operational level. Hence, some regulatory measures have to be taken from the inter-organizational perspective, making an interaction between these perspectives mandatory. Automated communication within the IPS² network is a prerequisite for a smooth interaction between all network partners and real-time monitoring of the IPS². Until now, systems supporting one of the key perspectives of IPS² management have been developed separately. For example, the IPS²-ES supports the planning of the IPS² delivery (inter organizational perspective) while the IPS² LMS represents the information perspective. However, for a fluent and efficient overall management of the IPS², an integration of these systems is needed.

To provide the required interaction between the diverse software systems from different perspectives, a common system architecture has to be established which allows for dynamic changes in the IPS² network while ensuring solid data exchange. In software engineering, there are several types of architectures. Among these are service-oriented [19], plug-in [20] and component-based architectures [21]. While each single software system from the varying domain-specific and partner specific IT landscapes may utilize its own architectural style, the common architecture has to connect all these heterogeneous systems. Therefore, different interfaces have to be established. A list of requirements is given in table 1, which also shows how the different architectural approaches fulfill these requirements.

Table 1. Fulfillment of Requirements by different System Architectures

| Requirement                                             | Architecture |
|---------------------------------------------------------|--------------|
|                                                        | SOA          | Component-based | Plug-in-based |
| R1. Flexibility                                         |              |                |
| R1.1 Adaptable to dynamic IT landscapes                 | +            | -              | +           |
| R1.2 Adaptable to heterogeneous IT environments         | -            | +              | -           |
| R1.3 No system downtime caused by changes               | +            | -              | o           |
| R1.4 Orchestration and management of processes          | +            | -              | -           |
| R1.5 Instant modification of automated business processes| +            | -              | -           |
| R2. Sustainability                                      |              |                |
| R2.1 Software usable for other independent software system | +            | -              | +           |
| R2.2 Backwards compatibility                            | +            | -              | o           |
| R3. Implementation                                      |              |                |
| R3.1 Low implementation effort for interfaces           | +            | -              | -           |
| R3.2 Programming language independent                   | +            | -              | -           |
| R4. Integration                                         |              |                |
| R4.1 System and platform independent                     | +            | -              | o           |
| R4.2 Reliable and scalable                              | +            | -              | +           |
| R4.3 Conforms to enterprise standards                   | +            | +              | +           |
| R4.4 Easily testable                                    | +            | +              | +           |

A component-based architecture is not suitable for the described dynamic IT landscape (R1.1). While the...
architecture conforms to enterprise standards (R4.3) and is easily testable (R4.4), the different components and the interfaces between them would have to be defined at an early stage before the buildup of the IT-landscape (R1.5). All partners would have to be identified beforehand and their software systems would have to be prepared to serve as a component in the architecture (R1.2). Furthermore, a change in the IT landscape or in the provider network as well as in the business processes would result in a modification of the component architecture, which has to be newly defined and constructed (R1.4).

In contrast to the component-based approach, a plug-in architecture would provide more flexibility (R1.1, R1.2). Plug ins can be added to or removed from the system at any time, however, a failure of the base plug in system would result in a breakdown of the whole constructed system (R1.3). Furthermore, most plug-in systems make use of language-specific programming (R3.2), which would imply a high effort supporting the plug-in approach in partner-specific applications.

A service oriented architecture (SOA) does not necessarily have the limitations described above. By using a loose coupling between interfaces, only the required interfaces are established and can be used independently of other systems and interfaces. A change of the system or the interfaces only affects applications or services which are already in use.

Web services have proven to be a simple, yet powerful approach to connect software systems over a network [22]. Web services are platform-independent (R4.1) and can be created and used with virtually any programming language (R3.2) in any application (R1.2). Hence, a service-oriented IT-landscape architecture using web services is an appropriate solution (see Fig. 3).

4. Prototypical Implementation and Validation

The proposed interaction between IPS² lifecycle management and IPS² delivery management has been implemented and verified in a case study, developed within the scope of the Collaborative Research Center Transregio 29 with the topic of “Industrial Product-Service Systems”. The case study is described in detail in [23] and [24].

The above mentioned case study simulates the business to business relationship between two industrial companies and considers different business models. The offered IPS² consists of a micro milling cell integrated with value-added services, e.g. the optimization of the customer-specific production process. The approach proposed in the current article has been validated based on the availability oriented business model. In this business model, an IPS² is delivered to the customer with the guarantee of high performance and is usually integrated as a production unit into a customer production process. Because of dynamic changes and uncertainties in the IPS² use phase, IPS² suppliers have to control and improve their IPS² during its use phase.

Figure 4 shows the conceptual scenario for the interplay within IPS² networks, especially the interaction between the IPS²-LMS, the IPS²-ES and the IPS²-CS. After the development of an IPS², the information about goods and their components are sent to the IPS²-CS. At the same time, data about the IPS² service model and the partner network are transmitted from the IPS²-LMS to the IPS²-ES (1). The model includes information about delivery process requirements, for example necessary competences of the workers and type of

![Diagram](image-url)
needed tools or spare parts. This data is used by the IPS²-ES to create the delivery plan by mapping available resources provided by the partners in the IPS² networks to delivery processes. The delivery plan is then forwarded to the IPS²-CS, which monitors the IPS² (2). Whenever the IPS²-CS observes that an unplanned demand arises, it requests the scheduling of a new delivery process for the IPS² from the IPS² ES (2).

The IPS² ES in turn updates the delivery plan and forwards this information back to the IPS²-CS. Parallel to this communication scheme, the delivery plan is always forwarded to the IPS²-LMS (3). This information will be connected to other data of the related IPS², for example information about required steps in a process’s execution or machine history (4). Prior to and during the execution of the delivery process, the service technician can interact with the IPS² LMS to receive the required information he is entitled to (5). After the execution, feedback from the service technician is reported to the IPS²-LMS. The sequence of the communication for steps (1) to (3) is depicted in Fig. 5.

![Fig. 5. Sequence diagram for the interaction of IPS²-LMS, IPS²-ES and IPS²-CS](image)

The following example demonstrates the developed concept with the focus on the IPS²-LMS and the IPS²-ES. An IPS² is developed for the customer Omicron who ensures technical as well as organizational availability of a micro milling machine. An IPS² model is developed which contains the delivery processes of maintenance, repair and delivery of cutting tools. The maintenance process is needed to ensure technical availability, the delivery of cutting tools is required for organizational availability and the repair process enables the IPS² provider to restore the availability as a reaction to an unexpected defect. The processes including their periodicity and their requirements, e.g. technician qualification, needed tools or spare part types, are pushed to the IPS²-ES using a Representational State Transfer (ReST, [25])-based web service called “StoreDeliveryProcesses”. Information about newly added or removed partners in the provider network will be pushed through the ReST-based web service “AddNetworkPartner” or “RemoveNetworkPartner” respectively. This leads to communication step (1). The IPS²-ES starts the strategic capacity planning and generates the delivery plan. The plan includes the scheduled maintenance and delivery processes of cutting tools complying with their periodicity. For each process, the plan determines which resources are involved in the delivery. When a new plan is available, the IPS²-ES informs all involved network partners and the IPS²-LMS is triggered via its Simple Object Access Protocol (SOAP, [26])-based web service “NewPlan” so that it can retrieve the generated plan via the IPS²-ES ReST-based web service “GetNewPlan” (3).

The IPS² LMS connects the needed information for each of the planned delivery processes. This will include a process description, historical delivery data on the generic delivery process and the related product share (4) for each maintenance process. Each human resource involved in the delivery processes can access the IPS²-LMS via a web-based cloud service using a specific delivery process ID to retrieve above mentioned information (5). Furthermore, the resource can post its feedback directly via the cloud service so it can be used to optimize the delivery process (see Fig. 6).

![Fig. 6. Screenshot of information retrieval via a cloud service](image)

This exchange of information is consistent while the availability of the IPS² is still guaranteed. However, when the IPS²-CS detects a damaged component or a shortage of cutting tools, it informs the IPS² ES that it needs to schedule a new delivery process, either a repair process or an additional delivery of cutting tools (2). After running an operational resource planning algorithm, the newly created delivery plan is again transmitted to the network partners and the IPS²-LMS (3), which again adds information (4) for retrieval by service technicians (5).

4. Conclusions and Outlook

The current paper provides an overview of different management perspectives of IPS² and presents an SOA-based architecture for the interplay and integration of different domain specific systems within the IPS² network. In order to
implement the SOA-based architecture, different web-service-based interfaces are developed and validated in a scenario. This approach makes it possible to manage dynamic networks and deal with changing partners. Furthermore, it facilitates the integration of several heterogeneous and independent software systems, not only newly developed IPS² systems but also existing systems of different domains. Hence, the integration of different management perspectives is accomplished.

To be able to use our approach, an IPS² provider has to have implemented the IPS² Lifecycle Management and IPS² Delivery Management as a foundation. This means that amongst his duties are the management of the whole IPS² lifecycle related information and the management of the IPS² dynamic network.

The implemented approach is using direct communication between the different software systems. While this approach is suitable for a prototypical implementation, this approach also implicates restrictions with respect to the extensibility and easy substitution of systems. For example, the different technologies ReST and SOAP used for web service communication have to be supported by all involved IT systems, depending on which interfaces have to be used. For future work, an enterprise service bus environment using, for example, the Oracle Fusion Middleware can be established to improve the above-mentioned SOA architecture. Such middleware enables the reuse of existing services and development of new services for the change of business processes during the IPS² lifecycle using Business Process Execution Language (BPEL). It also allows for a mapping of technologies like ReST and SOAP. By using this approach, the integration of new IT systems becomes less tedious.

Acknowledgements

We express our sincere thanks to the German Research Foundation (DFG) for financing this research within the Collaborative Research Project SFB/TR29 on Industrial Product-Service Systems – dynamic interdependency of products and services in the production area.

References

[1] Meier H, Roy R, Seliger G. Industrial Product-Service Systems—IPS². CIRP Annals - Manufacturing Technology 2010;59(2):607–27.
[2] Meier H, Uhlmann E, Krug CM, Völker O, Geisert C, Stelzer C. Dynamic IPS² networks and operations based on software agents. Industrial Product-Service Systems IPS² 2010;3(3):165–73.
[3] Aurić JC, Clement MH (eds.). Produkt-Service Systeme: Gestaltung und Realisierung. [Product-Service Systems: Design and Implementation]. Berlin: Springer; 2010.
[4] Durugbo C, Bankole OO, Erkoyuncu JA, Tiwari A, Alcock JR, Roy R et al. Product-Service Systems across Industry Sectors: Future Research Needs and Challenges. In: Sakao T, Larsson T, Lindahl M, editors. Industrial Product-Service Systems (IPS²): Proceedings of the 2nd CIRP IPS² Conference. Linköping: Linköping University; 2010. p. 535–542.
[5] Most OK. Clarifying the concept of product–service system. Journal of Cleaner Production 2002;10(1):237–45.
[6] Meier H, Steven M, Funke B, Boßlau M, Keine gen. Schulte J. Complexity and Flexibility of IPS² across various Planning Levels: Functional Thinking for Value Creation. In: Hesselbach J, Herrmann C, editors. Functional Thinking for Value Creation: Proceedings of the 3rd CIRP International Conference on Industrial Product Service Systems. Berlin, Heidelberg: Springer; 2011. p. 315–319.
[7] Meier H, Völker O. Service Provision for Industrial Product-Service Systems. In: Roy R, Shehab E, Hockley Chris, Khan S, editors. Enduring and Cost-Effective Engineering Support Solutions: Cranfield University Press; 2012. p. 57–64.
[8] Völker O. Erbringungsorganisation hybrider Leistungsbündel. [IPS² Delivery Organization]. Aachen: Shaker, 2012.
[9] Funke B. Adaptive Planungsmethode zur Terminierung der Erbringungsprozesse hybrider Leistungsbündel. [Adaptive Planning Method for the Scheduling of IPS² Delivery Processes]. Dissertation. Bochum: Shaker, 2012.
[10] Meier H, Morlock F, Dorka T. Functional Specification for IPS²-Execution Systems. In: Meier H, editor. Product-Service Integration for Sustainable Solutions. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 507–519.
[11] Meier H, Dorka T, Morlock F. Architecture and Conceptual Design for IPS²-Execution Systems. In: do Carmo Cunha PF, editor. 46th CIRP Conference on Manufacturing Systems 2013. Amsterdam: Elsevier; 2013. p. 365–370.
[12] Uhlmann E, Raue N. IPS² Control System for the Integrated Support of Service Processes. In: Shimomura Y, Kimita K, editors. The philosopher's stone for sustainability: Proceedings of the 4th CIRP International Conference on Industrial Product-Service Systems, Tokyo, Japan, November 8th-9th, 2012. Berlin, New York: Springer; 2012. p. 215–320.
[13] Meier H, Uhlmann E, Raue N, Dorka T. Agile Scheduling and Control for Industrial Product-Service Systems. In: Roberto Teti, editor. 8th CIRP International Conference on Intelligent Computation in Manufacturing Engineering: Elsevier; 2012.
[14] Jun H, Kiritis D. Xirouchakis P. Research issues on closed-loop PLS. Computers in Industry 2007;58(8-9):855–68.
[15] Kiritis D. Closed-loop PLS for intelligent Products in the Era of the Internet of Things. Emerging Industry Needs for Frameworks and Technologies for Exchanging and Sharing Product Lifecycle Knowledge 2011;43(5):479–501.
[16] Hribernik KA, Stietencron M, Hans C, Thoben K. Intelligent Products to Support Closed-Loop Reverse Logistics. In: Hesselbach J, Herrmann C, editors. Globalized Solutions for Sustainability in Manufacturing. Berlin, Heidelberg: Springer Berlin Heidelberg; 2011. p. 486–491.
[17] Michael Abramovici, Jan Michele, Manuel Neubach. Erweiterung des PLM-Ansatzes für hybride Leistungsbündel. [Enhancement of the PLM Approach for IPS²]. Available from: http://www.zwf-online.de/ZW101335.
[18] Abramovici M, Dang HB, Wolf M. IPS²-KOP: IPS² Knowledge-Based Service-Oriented Lifecycle Management Platform. In: Meier H, editor. Product-Service Integration for Sustainable Solutions. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 485–493.
[19] Mackenzie M, Laskey K, McCabe F, Brown P, Metz R. Reference Model for Service Oriented Architecture 1.0: OASIS Standard; 2006.
[20] Wolfinger R, Dhungana D, Priller H, Mössenböck H. A Component Plug-In Architecture for the .NET Platform. In: Hutchison D, Kanade T, Kittler J, Kleinberg JM, Mattem F, Mitchell JC et al., editors. Modular Programming Languages. Berlin, Heidelberg: Springer Berlin Heidelberg; 2006. p. 287–305.
[21] Blair G, Coupaye T, Stefani J. Component-based architecture: the Fractal initiative. Ann. Telecomunum. 2009;64(1-2):1–4.
[22] Booth D, Haas H, McCabe F, Newcombe E, Champion M, Ferris C et al. Web Services Architecture; 2004.
[23] Meier H, Steven M, Boßlau M, Alevifard S. Modeling of Flexibility assessment and monitoring framework for product-service systems. In: Coms C, editor. Proceedings of the 5th International Conference "Management of Technology - Step to Sustainable Production". 2013.
[24] Abramovici M, Dang H, Quezada A, Schindler T. A sustainability assessment and monitoring framework for product-service systems. In: Cosic P, editor. Proceedings of the 5th International Conference "Management of Technology - Step to Sustainable Production"; 2013.
[25] Fielding RT. Architectural styles and the design of network-based software architectures. Doctoral Dissertation. Irvine, California; 2000.
[26] World Wide Web Consortium (W3C). Simple Object Access Protocol (SOAP) Version 1.2. 2nd ed.(1.2); 2007; Available from: http://www.w3.org/TR/soap12/; [October 07, 2013].