

Migration from Traditional towards Cyber-Physical Production Systems

Ambra Calà, Arndt Lüder
Otto-v.-Guericke University
Universitätsplatz 2. 39106
Magdeburg, Germany
{ambra.cala, arndt.lueder}@ovgu.de

Ana Cachada, Flávia Pires
José Barbosa, Paulo Leitão,
Polytechnic Institute of Bragança
Quinta Sta Apolónia, 5300-253
Bragança, Portugal
{acachada, fpires, jbarbosa, pleitao}@ipb.pt

Michael Gepp
Siemens AG Corporate Technology
Günther-Scharowsky-Str.1
91052, Erlangen, Germany
michael.gepp@siemens.com

Abstract—Nowadays, many organizations intend to convert their existing production systems towards ones that are characterized by adaptability, openness, flexibility and modularity. This requires a redesign of existing information processing systems especially related to control, leading possibly to cyber-physical production systems (CPPS). However, the implementation of new control technologies will have a direct impact on the normal operational status of production while engineers will also face several challenges and obstacles in adopting intelligent automation systems. New step-wise migration strategies are required to holistically support industries in their journey towards CPPS taking into account technical, economic and social aspects. This paper discusses the migration state-of-the-art strategies, analyzing them and providing a first attempt to define a migration approach for innovative production systems.

Keywords: migration strategy, cyber-physical production systems (CPPS), engineering processes.

I. INTRODUCTION

Modern markets are characterized by shorter product life-cycles, increased product variety and shorter time-to-market. Industries need to adapt and reconfigure more frequently their production systems to offer new product variants, while maintaining high-quality standards and minimizing costs. Cyber-Physical Production Systems (CPPS) [1] have the potential to increase the production's efficiency, enabling flexible and re-configurable realization of automation system architectures. However, their industrial implementation is complex and not straightforward.

The majority of actual manufacturing control systems are based on centralized and hierarchical structures which present good characteristics in terms of predictability, robustness and global optimization. On the other hand, they show an increasing cost consumption in case of adaptation execution based on re-engineering and re-design following changing requirements. Distributed control architectures with non-hierarchical modules linked together through different communication systems can instead be suitable for flexible and re-configurable automation systems with highly interconnected and interactive CPPS [1], [2].

The deployment of new automation technologies with decentralized control systems will have a direct impact in industrial environment, considering the current legacy systems

and processes, and needs to be performed in a smooth manner. Therefore, a migration strategy is required to support industries to move from their traditional production systems characterized by rigid centralized control approach towards an agile plug-and-produce system that is dynamically adaptable to changing production environment, open to new features and functions, flexible to different processing tasks and modular to enable quick and economical changes.

Within this paper the current migration processes, considered as a sequence of activities to achieve a migration goal, and strategies, defined as a set of methods and techniques to perform the activities [3], present in literature for different systems are analyzed and compared. The objective is to identify the characteristics of a general migration approach towards Cyber-Physical Production Systems (CPPS) and to propose a model for selecting the migration strategy.

In the following section three main strategies for migration of systems are described and compared. In Section III the state-of-the-art migration processes adopted in previous research projects are presented and a set of engineering processes is reviewed. Section IV describes how the migration for CPPS can look like, deriving the strategy and the characteristics of the general process. The improved migration approach is presented in section V, while the last section concludes the paper with some remarks and outlines the next research steps.

II. MIGRATION STRATEGIES

The existing strategies to transform a legacy system into a target system are analyzed in this section.

Generally, three main migration strategies, coined as Big Bang, Parallel Systems and Phased Introduction, can be found in the literature allowing a more versatile, feature-rich and cost-effective technological transition from a legacy to the target system(s). These strategies differ in the general techniques applied in the migration process.

The following sub-sections describe and compare, thereby, the three migration strategies, detailing for each one of them their advantages and disadvantages.

A. Big Bang Strategy

The Big Bang strategy can be described as a change in a single moment in time, switching off the legacy system, i.e.

the *as-is* situation, and switching on the target system, i.e. the *to-be* situation, on a set date, known as the *Go-Live* date [4], [5].

With this strategy some advantages come, in comparison with the others migration strategies present in the literature, as the amount of time spent for its implementation is very little. The costs are lower since the whole transformation takes place at once, without the need to have intermediate programs and/or duplicated resources. Moreover, the training of the employees is centered in the new system, not wasting time in the training of transition programs [5].

However, this strategy has a huge risk for the enterprise given the difficulty of re-creating all the conditions of a live production environment [4]. Considering all the interdependencies, a failure in one element of the system can cause problems in other modules [5], [6]. In this case, a small failure may be very difficult to recover or even fatal. In addition, the available time to train the employees is very little [5].

The Big Bang strategy is therefore suitable for the migration of production systems requiring a complete organizational/technological change, for example, in a scenario where a new product model is introduced in the system (e.g., the introduction of a new model in the automotive industry).

B. Parallel Systems Strategy

In this strategy, both legacy and target systems run at the same time, i.e. in parallel, for a certain period of time [4], [7]. This time corresponds to the migration execution time, in which the legacy system is designated as Master and the target system as Slave. The target system becomes the Master system only after is tested and validated and, then, the legacy system becomes the Slave system or is switched-off [4]. If the legacy system continues running as Slave, additional costs need to be considered in the migration process [4].

Given the fact that both systems will run together, all transitions will be carried out in both systems, meaning that a synchronization is required. This synchronization can also bring additional costs to the migration process [4], [7]. Additionally, since the systems run together until the target one is validated, there is a low likelihood of problems, which means that this migration approach involves a small risk [7].

Having both systems, the old and the new running at the same time, bring certain advantages to the users. The comparisons between the legacy and the target systems can be performed in real-time and it is possible to improve the target system during the migration process. One of the most important advantages is that in case of failure it is possible to rollback to the legacy system [5], [4].

However, this strategy involves a huge number of resources and the duplication of the functional systems resulting in very high implementation costs [5].

Therefore, the parallel strategy is adequate for migration of critical (software) systems and small production lines that cannot survive with a major system failure [5].

C. Phased Introduction Strategy

The Phased Introduction strategy allows executing the migration through a gradual transition, following a well-planned sequence [7]. Initially, this migration requires an intensive study of interdependencies and processes' priorities in order to know the correct sequence of the migration phases.

The implementation of this migration starts by introducing the target system block-by-block, taking into account the previous study, turning it on and shutting down the legacy system. This process is repeated until the target system is completely implemented, replacing all the legacy systems by the target system (e.g., in the entire factory) [4], [7].

Since this process is executed by replacing step by step smaller blocks, it is possible to get feedback between each phase, promoting a continuous improvement of the migration process [5]. This strategy also carries advantages as low level of complexity, which means a lesser risk and consequently lesser resources are required. The high implementation time makes possible to the employees to have more time to adapt to implemented changes [5].

Similarly to the previous strategies, this one also presents some disadvantages, namely very high implementation time [5] and high implementation costs [4].

An important aspect of this migration strategy is the definition application areas, followed by a definition of the secondary types of the migration strategies. In fact, for each phase, the previously described strategies, i.e. Big Bang, Parallel and also the Phased strategy, can be used independently. This represents a recursivity in the implementation of this strategy, meaning that it is possible to repeat recursively the choice of the migration strategies, namely Big Bang, Parallel and Phased strategies, according to the granularity of the factory level, as illustrated in Figure 1.

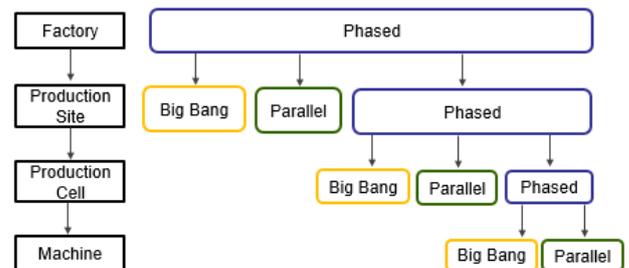


Figure 1. Recursivity in the implementation of the Phased strategy.

In this case, and selecting the Phased strategy for the migration at the factory level, the gradual migration at the production sites can be implemented by considering Big Bang, Parallel and/or Phased strategies for the different sites. If a phased strategy is selected for one production site, its implementation at production cell level can recursively use the same approach, being implemented by using Big Bang, Parallel and/or Phased strategies for the different cells.

D. Migration Strategies Comparison

The selection of the best migration strategy to be adopted depends on the environment and the addressed technical, economical and social conditions. The comparison of the different strategies considers the assessment of several features, such as risks, migration design time, migration execution time, downtime and costs (effort), as summarized in Table I.

Table I
MIGRATION STRATEGIES COMPARISON [4].

| | Big Bang | Parallel | Phased |
|--------------------------|-----------------|-----------------|---------------|
| Risk | High | Low | Medium |
| Migration Design Time | High | Low | Medium |
| Migration Execution Time | Low | Medium | High |
| Down Time | High | Low | Medium |
| Cost (Effort) | Low | High | Medium |

Briefly, it is possible to conclude that the Big Bang strategy has a low cost of implementation but involves a higher risk, migration design time and downtime. In opposite, the Parallel strategy has a low risk, migration design time and down time, but has a high cost. The Phased strategy is a kind of compromise between these two approaches, presenting a high migration execution time.

III. STATE-OF-THE-ART MIGRATION PROCESSES: AN APPLICATION PERSPECTIVE

Migration processes that support industries in adopting new technologies have been already considered in previous research projects, however they are not numerous and not designed for planning and supporting the smooth migration towards cyber-physical production systems.

Existing migration processes mainly concern the migration of Information Systems. Bisbal et al. [8] survey the processes of legacy information system migration and suggest five phases for a generic migration process, named as the “Butterfly Method”: Justification, Legacy System Understanding, Target System Development, Migration, and Testing. Within the initial phase risks and benefits associated with legacy system evolution are investigated, determining the economic benefits of the evolution and the technical feasibility. The goal of the second phase is to identify the legacy system components, understand their static and dynamic behavior, and recreate documentation. The main activity in the third phase is the elicitation of requirements and specifications of the target system and the selection of the most adequate architecture and standards of the target system. The Migration phase consists in the physical transformation of the complete legacy system to the target system, while tests are carried out throughout the evolution process to ensure that the target system delivers the specified functionalities of the evolution.

Moreover, several processes and techniques have been defined for migrating from legacy applications to services and cloud computing. Lewis et al. [9] propose a process for migrating legacy IT systems to Service-Oriented Architecture (SOA)

environment using their Service Migration and Reuse Technique (SMART). The SMART Migration Planning process has six iterative activities which are: Establish Context, Define Candidate Services, Describe Existing Capability, Describe Target SOA Environment, Analyze the Gap, and Develop Strategy. The first goal of the process is to understand the migration context in terms of business, stakeholder goals, candidate services, and legacy systems, in order to determine if the migration of the considered legacy systems is feasible or not. Initially, few of the candidate service are selected and more specified. In parallel, the legacy system components are described, as well as the target SOA environment. Then the gap between candidate services, target SOA environment and legacy system components is analyzed to estimate the effort, risks and costs related to the migration. The final activity develops a strategy from legacy components to services.

Cetin et al. [10] propose the MASHUP process to migrate legacy system to service-oriented computing. It is based on the mashup technology and consists of six steps: Model, Analyze, Map and Identify, Design, Define, and Implement and Deploy. The first step is modeling the business requirements to understand the functional requirements of the target system. The second step is analyzing the existing legacy systems to identify the Quality of Service attributes and the important data for Domain Specific Kits (DSKs). The third step is mapping the functional and non-functional requirements to DSK components and identifying Business Requirements to Services and Service to Service requirements. The fourth step is designing a concrete “Mashup” Server with DSKs. Next step is defining the Service Level Agreements and final step is implementing and deploying the system.

Zillmann et al. present in [11] the SOAMIG process, a generic and iterative transformation-based migration process mainly focused on code and architecture migration. SOAMIG consists of four phases: Preparation, Conceptualization, Migration, and Transition. In the first phases the legacy system is standardized and prepared for the conversion activities. Secondly, the technical feasibility of migration and tool adaptation are assessed in order to define a migration strategy and migrate the entire system. During the Conceptualization and Migration phases the SOAMIG core disciplines are performed iteratively. The core disciplines are: Business Modeling, Legacy Analysis, Target Architecture, Strategy Selection, Realization, Testing, and Cut Over. Finally, the performed transition of the system is evaluated.

Baserra et al. [12] present a step-by-step decision process aimed at supporting legacy application migration to the cloud. The main objective of the Cloudstep process is the identification and analysis of key factors that might influence the cloud selection and relative migration task. The process is characterized by nine activities: Define Organization Profile, Evaluate Organizational Constraints, Define Application Profile, Define Cloud Provider Profile, Evaluate Technical and/or Financial Constraints, Address Application Constraints, Change Cloud Provider, Define Migration Strategy and Perform Migration. The starting point is to collect relevant legal or administrative

information that might influence the cloud migration decision. Moreover, the organizational constraints are evaluated in order to preliminarily identify the critical factors for cloud adoption within the organization. This phase is followed by two parallel activities: the identification of usage and technical characteristics of the application targeted for migration, and the definition of each candidate cloud provider characteristics. Afterwards the organization profile, the application profile and the profile of the candidate cloud provider are evaluated considering seven types of constraints (financial, organizational, security, communication, performance, availability, and suitability). If there are no constraints the next step consists in the definition of the migration strategy for the legacy application. Otherwise, the next activity addresses the constraints in the context of the application, the cloud provider or the organization. Then the actual migration to the cloud is performed.

Another process that has been taken also into account is the stepwise migration process from legacy DCS/SCADA system to SOA components proposed by Colombo et al. [13] within the project IMC-AESOP. The process is characterized by four main steps: Initiation, Configuration, Data processing and Control execution. It starts defining services representing the legacy components. Then the parts of legacy DCS/SCADA systems, which do not require short response times, are migrated in a structured way. Within the last step, the functionality provided by controllers are migrated considering real time requirements.

Fuentes-Fernández et al. [14] proposed a model-driven process for the modernization of component-based systems defined in the MOMOCS project. The modernization methodology is called XIRUP and includes an iterative process that consists of four-phases: Preliminary Evaluation, Understanding, Building, and Migration. Firstly, a cost-benefit analysis is conducted to decide if the system is going to be modernized or not. Afterwards, the information regarding both legacy system and target platform are gathered, including the identification of components, transformations and constraints to build the modernized system. Then, the transformations from legacy systems components to the new system components are established. Finally, the deployment of new components is addressed.

Even though only a set of recent migration approaches have been considered here, some conclusions or hypotheses on how a migration process towards CPPS should look like can be derived. Regardless of domain and target of the migration, the described processes present some similarities. All the processes considered a stepwise approach in which, first the legacy system and the target system are analyzed, then the target system is developed, finally the migration is defined and performed all in accordance to the general migration process described in Section II. During the first phases of the processes the requirements are defined. Processes like SOAMIG and IMC-AESOP focus mainly on the technical constraints and characteristics of the migration, while SMART, MASHUP and XIRUP pay attention also on business requirements and involved stakeholders, and Cloudstep includes legal, administrative and organizational constraints.

In addition, mostly of the described processes analyze the migration iteratively but only XIRUP takes into account possible new features after the successful validation of the migrated components.

Migration processes usually consist of a rigid transition process to a precise goal (e.g., service-oriented architecture, cloud computing) rather than considering a continuous improvement from a system of systems perspective towards the innovative technology concept. Moreover, these processes usually define the migration strategy to the target system not taking into consideration the possible alternatives of migration and their evaluation under technological, economical and organizational aspects. Thus, a new migration process, suitable for CPPS, needs to be defined.

IV. REQUIREMENTS FOR MIGRATION PROCESS TOWARDS CPPS

The migration towards CPPS is costly and full of uncertainties. Several factors need to be considered together and different options of migration need to be analyzed and evaluated to identify the best migration strategy for the considered systems. To support industry in adopting new technologies and move towards CPPS, a set of required characteristics has been derived from literature:

- *Generalized*: the process is valid for production systems, but not limited to technical parts, allowing for other domains to follow the same general migration process;
- *Step-wise*: the process considers a step-by-step implementation to enable a smooth migration [15], [16];
- *Incremental*: the process aims innovation incrementally, feature-by-feature and component-by-component [17];
- *Iterative*: the process is designed to be iterative. If the goal is not achieved during the first iteration users are suggested to go back and repeat the set of activities;
- *Flexible and "open-ended"*: the general workflow is a set of process fragments and users can flexible combine these fragments to achieve their specific goals [15] and to recombine them if the goals slightly change;
- *Include alternatives/options*: the process considers several possible solutions from which one has to be chosen;
- *Integrate economical, technological, legal and social aspects*: the process comprehends an integrative and interdisciplinary work mode between all technical functions but also between social, economic, legal and technological functions that need to be considered simultaneously, not consecutively [18].
- *Distinguish tactical and strategical goals*: where the tactical goals are the intermediate short-term goals to reach the strategical goal, i.e. the long-term vision of the migrating system [19].

Contrarily to the existing processes described in the previous section, the migration process should consider all these characteristics in order to enable the migration.

The migration process should be step-wise and iterative, like the current processes, to enable a smooth migration of the system. However, especially the last four characteristics are

fundamental when considering a continuous innovation and improvement of the system. In such uncertain environment, it is very important to always keep a holistic view on the migration, comprehending the long-term vision and the short-term goals as well as the possible migration options, risks and opportunities related to the new technology solution and its impact on different levels, such as economical, legal and social. For this reason, the process should allow also a kind of flexibility to modify the migration approach according to changing conditions of the market environment or even the system goal.

Following these requirements a corresponding migration process is derived and described within the following section.

V. MIGRATION APPROACH FOR CPPS: THE PERFORM PERSPECTIVE

Within the EU HORIZON 2020 project PERFoRM (Production harmonizEd Reconfiguration of Flexible Robots and Machinery) [20], a migration strategy to implement CPPS in industry is going to be developed. This section presents the current research results achieved. The migration process is described followed by a general migration strategy to transform traditional production systems into a CPPS.

The migration process proposed here is inspired by existing industry-driven engineering processes and standardized processes, especially the VDI 3695 [21] and PDCA [22], that, even though developed for different purposes, present some of the required characteristics for migration towards a CPPS, as defined previously. Based on these and in the evaluation of existing engineering processes, a general migration process has been derived to improve the production system.

Considering the stepwise, incremental and iterative characteristics, two different kind of goals are taken into account: the long-term vision and the short-term goals [19]. The long-term vision represents the target production system, e.g. the CPPS, that the manufacturer intends to achieve in the long run, following the company's strategy, and will be reached throughout a number of intermediate short-term goals. The objective of the migration process is to define the right path towards the long-term vision, designing the step-by-step short-term goals. The migration path consists in migration steps that represent different solutions to achieve the goal. These options need to be investigated and evaluated to identify the right migration path considering different decision aspects. Therefore, the migration process will be repeated for each migration step in the direction of the planned goal and after each process cycle the next goal is defined.

The general process consists of five phases: Preparation, Options investigation, Design, Implementation, and Deployment.

The process starts with the *Preparation* phase. The purpose of this phase is to analyze the existing system and define the target system. In this first phase the context of the system is defined, namely what is the motivation of the migration and what are the actors and systems involved in the migration process. In addition, the long-term vision, specifically the target system in which the current system is going to migrate,

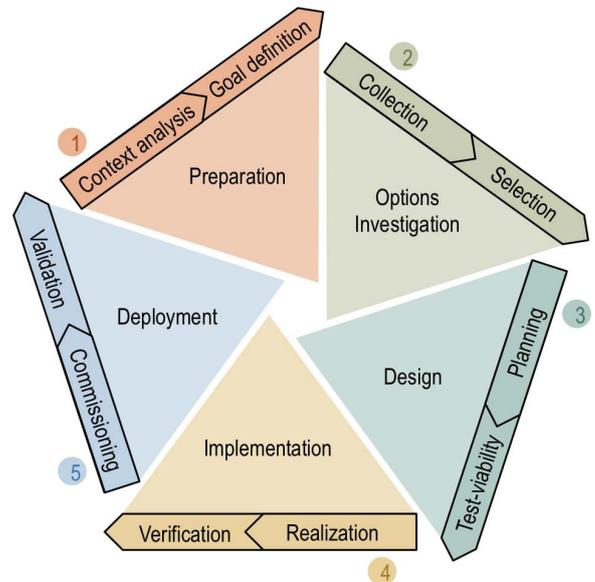


Figure 2. The Migration Process for CPPS.

is defined. Based on the factory vision and mission, selection criteria, or impact aspects, are derived in order to evaluate different migration solutions and to guide the selection of the optimal one. Consequently, a set of solution options for the migration is investigated. In the *Options Investigation* phase, possible solutions are collected and assessed. This is the most crucial phase of the process since there may be many options to implement, in a stepwise approach, the target system. The selection of the optimal migration solution (migration step) to achieve each short-term goal depends on the relevant impact aspects of the factory, defined in the Preparation phase.

This solution option is then detailed within the *Design* phase. Here, the tasks necessary to implement and integrate the selected solution within the system are planned accordingly to the migration strategy, i.e. Phased Introduction, Parallel Systems or Big Bang. Moreover, in this phase also the viability of the designed solution is tested, ensuring that the next phase is only initiated when a viable planning has been meet.

In the *Implementation* phase the selected solution option is realized, accordingly to the migration strategy defined in the design phase, and verified. Here, the partly transition from the old system to the new CPPS-like system is carried out.

Finally, with the *Deployment* phase the new system is installed and further validated, in a real-environment state, to ensure that the new system performs as intended. If the results do not match with the expected benefits, the user can repeat the previous phases and select a different option or re-defining the goal of the migration. The process is repeated for each migration step.

This five-phase migration process shapes the migration path towards the target system iteratively and step-wise, defining the intermediate technical solutions and migration strategies at every iteration of the process.

The process provides an orientation for the user to migrate

from the legacy systems with a defined sequence of activities. It is also possible to change their order repeating some activities as soon as it is required by any type of change, e.g. the process environment, the market demand, the short-term goals or even the long-term vision. For the same reason, the possible technical solutions collected during the Option Investigation phase can be stored and re-considered in case the selected one fails or is not realizable in the following phases due to some unexpected changes.

Concluding, the migration of a manufacturing system is an overwhelming and complex task that cannot be simply handled by selecting a migration strategy. The migration must be carefully considered and the proposed process acts as a glue in the migration process, unifying different concepts that otherwise would act in a loose manner. Even though the identified migration strategies provided a mean to promote the migration step, they lacked the underlying critical decisional and control features. The proposed five-phase migration overcomes this by building an iterative, logical and structured migration process around the migration strategies, defining the necessary steps to accomplish a secure migration process.

VI. CONCLUSION

Recently, several technologies have been developed in the manufacturing domain to increase production flexibility and reconfigurability in order to meet the market demands. However, manufacturers are still conservative in adopting those new solutions, especially because of the lack of roadmaps and techniques that support the smooth migration from legacy systems to the next generation of production systems.

This paper briefly presented the generic PERFoRM migration process towards CPPS. First, the three main migration strategies, namely Big Bang, Parallel Systems and Phased Introduction, have been described and compared, highlighting their pros and cons. Secondly, the state-of-the-art migration processes for legacy software and production systems have been briefly reviewed. The described processes present a similar structure but they cannot be applied as a generic migration process suitable to migrate from traditional to CPPS.

From this literature review a set of requirements for the migration process towards CPPS has been derived and a five-phases process, developed within the PERFoRM project, has been described at high-level manner.

Future research steps will focus on the design and refinement of the PERFoRM migration approach with techniques to support manufacturers in the execution of the five phases of the process. Furthermore, the migration approach will be applied to different industrial use cases and, thus, tested and validated within the PERFoRM project.

ACKNOWLEDGMENT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 680435.



REFERENCES

- [1] E. Lee, "Cyber physical systems: design challenges," in *11th IEEE international symposium on object/component/service-oriented real-time distributed computing*, may 2008, pp. 440–451.
- [2] L. Monostori, "Cyber-physical production systems: Roots, expectations and r&d challenges," in *Procedia CIRP*, vol. 17, 2014, pp. 9–13.
- [3] J. Estefan, "Survey of candidate model-based systems engineering (mbse) methodologies, rev. b," 2008.
- [4] Critical Manufacturing, *MES Migration Strategies [White Paper]*, (2015). [Online]. Available: <http://www.criticalmanufacturing.com/en/resources/documents/mes-migration-strategies>
- [5] P. Madkan, "Empirical Study of ERP Implementation Strategies-Filling Gaps between the Success and Failure of ERP Implementation Process," *International Journal of Information & Computation Technology*, vol. 4, no. 6, pp. 633–642, 2014. [Online]. Available: <http://www.ijict.com>
- [6] HSO, *Choosing the right ERP implementation strategy for your company [White Paper]*, (2015). [Online]. Available: <https://www.hso.com/uk/knowledge/news-events/detail/article/whitepaper-choosing-the-right-erp-implementation-for-your-company>
- [7] Open Text, *Top 10 Best Practices in Content Migration [White Paper]*, (2009). [Online]. Available: http://connectivity.opentext.com/common/whitepapers/Whitepaper_Content_Migration_Best_Practices_Report.pdf
- [8] J. Bisbal, D. Lawless, B. Wu, J. Grimson, V. Wade, R. Richardson, and D. O'Sullivan, "A survey of research into legacy system migration," Tech. Rep., 1997.
- [9] G. Lewis, E. Morris, D. Smith, and S. Simanta, "Smart: Analyzing the reuse potential of legacy components in a service-oriented architecture environment," Tech. Rep., 2008.
- [10] S. Cetin, N. I. Altintas, H. Oguztuzun, A. H. Dogru, O. Tufekci, and S. Suloglu, "A Mashup-Based Strategy for Migration to Service-Oriented Computing," in *IEEE International Conference on Pervasive Services*. IEEE, jul 2007, pp. 169–172.
- [11] C. Zillmann, A. Winter, A. Herget, W. Teppe, M. Theurer, A. Fuhr, T. Horn, V. Riediger, U. Erdmenger, U. Kaiser, D. Uhlig, and Y. Zimmermann, "The SOAMIG Process Model in Industrial Applications," in *15th European Conference on Software Maintenance and Reengineering*. IEEE, March 2011, pp. 339–342.
- [12] P. V. Beserra, A. Camara, R. Ximenes, A. B. Albuquerque, and N. C. Mendonca, "Cloudstep: A step-by-step decision process to support legacy application migration to the cloud," in *IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA)*, September 2012, pp. 7–16.
- [13] T. B. Jerker Delsing, Fredrik Rosenqvist, Oscar Carlsson, Armando W. Colombo, "Migration of Industrial Process Control Systems into Service Oriented Architecture," in *38th Annual Conference of IEEE Industrial Electronics Society (IECON12)*, 2012.
- [14] R. Fuentes-Fernández, J. Pavón, and F. Garijo, "A model-driven process for the modernization of component-based systems," *Science of Computer Programming*, vol. 77, no. 3, pp. 247–269, March 2012.
- [15] G. S. Lynn, J. G. Morone, and A. S. Paulson, "Marketing and Discontinuous Innovation: The Probe and Learn Process," *California Management Review*, vol. 38, no. 3, pp. 8–37, April 1996.
- [16] D. Dörner, *Die Logik des Misslingens: Strategisches Denken in komplexen Situationen*. Rowohlt, 2007.
- [17] P. G. Smith and D. G. Reinertsen, "Shortening the Product Development Cycle," *Published in Research-Technology Management*.
- [18] U. Lindemann and M. Lorenz, "Uncertainty handling in integrated production development," may 2008.
- [19] M. Rother, *Toyota Kata*. McGraw Hill, New York, 2010.
- [20] *PERFoRM – Production harmonizEd Reconfiguration of Flexible Robots and Machinery*, 2017. [Online]. Available: <http://www.horizon2020-perform.eu>
- [21] *VDI-Richtlinie 3695 - Engineering von Anlagen - Evaluieren und optimieren des Engineerings*, 2009.
- [22] E. Lodgaard, I. Gamme, and K. E. Aasland, "Success factors for pdca as continuous improvement method in product development," in *Advances in Production Management Systems (APMS12). Competitive Manufacturing for Innovative Products and Services*, 2012, pp. 645–652.