TOWARDS A RECONFIGURATION FRAMEWORK FOR SYSTEMS ENGINEERING INTEGRATING USE PHASE DATA

Qasim, Lara (1,3); Hein, Andreas Makoto (1); Jankovic, Marija (1); Olaru, Sorin (2); Garnier, Jean-Luc (3)

1: CentraleSupelec,Laboratoire Génie Industriel,France; 2: CentraleSupelec,Laboratoire de Signaux et Systemes,France; 3: Thales Technical Directorate, France

ABSTRACT

One of the aims of systems engineering is to develop systems with a number of pre-defined configurations, in order to operate effectively and efficiently in different contexts and environments. Early in the design phase, system reconfiguration allows to propose and optimize these configurations. With regard to the literature review and industrial observation, pre-defining the standard configurations without relying on hints from end users has been raised as a major difficulty within the industry. In this paper, we propose a reconfiguration framework which considers data collected from the use phase in order to generate valid and optimized configurations with regard to stakeholders needs.

Keywords: Reconfiguration, Configuration design, Systems Engineering (SE), Complexity, Big data

Contact:
Qasim, Lara
École Centrale Paris
Laboratoire Génie Industriel
France
lara.qasim@centralesupelec.fr

Cite this article: Qasim, L., Hein, A.M., Jankovic, M., Olaru, S., Garnier, J.-L. (2019) 'Towards a Reconfiguration Framework for Systems Engineering Integrating Use Phase Data', in *Proceedings of the 22nd International Conference on Engineering Design (ICED19),* Delft, The Netherlands, 5-8 August 2019. DOI:10.1017/dsi.2019.268

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

1 INTRODUCTION

Systems are evolving during their life cycles with respect to the operational context and the environment. System configuration management is essential to ensure the effective management of an evolving system during its lifecycle (ISO/IEC/IEEE/15288, 2015; Walden and Roedler, 2015). System configuration can be characterised with regards to economic, environmental, legal, operational, behavioural, structural, and social aspects that are necessary to demonstrate a capability. Any change of these aspects can lead to "System Reconfiguration" in order to maintain an operational effectiveness. Consequently, "System Reconfiguration" is defined in this paper by subsequent changes of the system configurations with the objective of maintaining or adapting (increasing or decreasing) the capabilities provided by the system.

A current trend in Systems Engineering is Model-Based Systems Engineering (MBSE). In MBSE the system model is placed at the centre of the system development process (Wymore, 1993). Similarly, one can consider MBSE as a possible venue to support system reconfiguration with the objective to support knowledge capture, reconfiguration rules; and aim at cost and time reduction due to the model use and reuse.

Within the current industrial practice of a large international aerospace, space, ground transportation, defence and security company, system architects and engineers design new configurations based on use case scenarios provided and brainstormed with end users. Feedback from the use phase is rarely integrated automatically in the design phase. Only insights from the teams operating the system is considered during the design phase, via the channel of end user company representatives.

To our knowledge, no system reconfiguration framework taking use phase data integration into consideration has been discussed in the existing literature. This paper aims at proposing a system reconfiguration framework based on data collected from the use phase. For this purpose, a rigorous field study has been conducted in order to identify current industrial challenges pertaining to system reconfiguration. The field study is based on interviews, direct observations and company reference documents. Section 2 discusses the background on existing approaches and methods for system configuration and reconfiguration. Section 3 details research design and protocols. The results of the field study are discussed in section 4, identifying difficulties and challenges of the system reconfiguration process. For addressing these challenges, a system reconfiguration framework is proposed in section 5. Section 6 presents our conclusions and future work.

2 LITERATURE REVIEW

We first investigate the literature in different domains treating the reconfiguration problem in general (Section 0). Next, we examine the relevant literature with regard to our specific context. Section 0 examines literature in MBSE that treats reconfiguration. Section 0 explores an approach which is already used for capitalizing data collected during the use phase for reconfiguration purposes. Section 0 considers literature dealing with collecting data from all lifecycle phases and exploiting them in the design phase.

2.1 Model-based reconfiguration

In order to address model-based reconfiguration, we performed a Scopus data base search, which generated 102 hits. The aim was to identify different research areas where model-based reconfiguration is discussed. With regard to when the system reconfiguration is done in the life cycle, reconfiguration mechanisms and objectives can be different. Hence, to investigate how the reconfiguration problem has been treated in different domains and objectives and mechanisms addressed by these domains during design-time and run-time, we propose to classify the reviewed domains into two categories; design-time and run-time related domains.

2.1.1 Design-time related domains

In this category, the following concepts have been investigated: product platform design, design for reconfigurability, and design for autonomy. These concepts are considered enabling concepts for reconfiguration. We considered the product platform design literature as relevant, as product configuration during development is achieved by introducing platform-based engineering. Deciu *et al.* (2005) define

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

configuration as the arrangement of components. According to them, the configuration task consists of component selection, determination of relationships between components, component arrangement (configuration), and evaluation of product configuration. Martin and Ishii (2002) describe a step-by-step method that aids companies in developing product platform architecture to support design for variety. A methodology that supports finding candidates for modifications of the function structure and configuration of a platform is proposed by Umeda *et al.* (2005). Simpson *et al.* (2001) introduces the product platform concept exploration method (PPCEM) to facilitate the design of a family of products based on a scalable product platform. To tackle future customization and market uncertainty, de Weck and Suh (2006) propose a flexible platform design process. To support the flexible platform design, Elsner *et al.* (2010) propose a general approach for constraint checking across arbitrary configuration files. All these studies address product design within the platform design paradigm. On the other hand, Haubelt *et al.* (2002) extend the reconfiguration problem to reconfigurable, platform-based systems that implement time dependent functionality. In this study the authors introduce a hierarchical graph-model for describing alternatives for the behaviour of the system.

In the design for reconfigurablity domain, researchers aim at enhancing performance across a variety of diverse and changing system operating conditions. Reconfigurable systems are designed to maintain a high level of performance by changing their configuration to meet multiple functional requirements or a change in operating conditions within acceptable reconfiguration time and cost (Siddiqi *et al.*, 2006). Flexibility is a term that is often used in the context of reconfigurability. Ferguson *et al.* (2007) give a classification scheme for flexibility and its application to reconfigurable system design. In further works, Siddiqi and de Weck (2008) propose to represent the reconfigurable systems by Markov models and meta control frameworks. In the same study, Siddiqi and de Weck (2008) also propose a methodology and guidelines for reconfigurable systems design. Siddiqi and de Weck (2009) develop metrics for assessing the impact of reconfigurability and a methodology for determining optimal designs of reconfigurable systems.

In order to meet resilience needs, it is necessary to enhance systems with self-aware and self-adaptation capabilities. Sanz *et al.* (2014) describe some developments in the direction of model-based and architecture-centric processes for autonomous controller construction in the form of reusable design patterns. A general design solution to build autonomous systems capable of reconfiguration is proposed by Hernández *et al.* (2015). Khederzadeh and Beiranvand (2018) present a new method to model and deal with the cascading failures in an autonomous microgrid in order to protect it against risks that may lead to its complete collapse.

To summarize, reconfiguration in the design-time related literature has different meanings. Within the product platform domain, reconfiguration concerns product rearrangement to satisfy new requirements. In design for reconfigurability and design for autonomy domains, the issue is how to make a product reconfigurable and how to design the deployed configurations.

2.1.2 Run-time related domains

Reconfiguration while the system is in use has been widely treated in the literature as a transition between system modes; functional or failure modes. In embedded systems and control theory, configuration corresponds to a functional mode (Krichen and Zalila, 2011; Provan and Chen, 1999). This mode corresponds to how components, elements, and processes are integrated and how they interact. Reconfiguration is referred to as transition between systems modes when triggering events occur. In reconfigurable manufacturing systems researchers define configuration as the composing modules (software and hardware) of the manufacturing system and their on-site set up (Regulin *et al.*, 2016). The change in structure, hardware, and software components to quickly adjust production capacity and functionality within a part family is referred to as Reconfiguration (Alsafi and Vyatkin, 2010). Configuration of computing and informational systems is defined as assembled components that deliver functions (Saxena *et al.*, 2010). The authors of Saxena *et al.* (2010) define reconfiguration as swapping faulty components for working ones in the event of either component failure or an evolved mission goal.

2.2 Model-based systems engineering (MBSE) and reconfiguration

We have searched for literature in MBSE that treats reconfiguration. We find publications that primarily deal with the reconfiguration of simulation models in the early stage of systems development, specifically in the system architecting phase. Reconfigurable simulation models can be

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

used to simplify the optimization of system architecture or topology (Haris and Dagli, 2011; Wichmann *et al.*, 2015). In simulation based reconfiguration, the effects of the environment are considered to reconfigure initially selected architectures. Production reconfiguration is essential for variety management (Wu *et al.*, 2013). In system of systems development, evolutionary computation algorithms are used to evaluate and generate architectures (Simpson and Dagli, 2008). Others use agent-based techniques (Acheson *et al.*, 2013).

2.3 Health and usage monitoring systems (HUMS)

The functions of HUMS systems are: 1) to assess the well-being of vital components in the system (Health), 2) to determine the time to failure of some major components (Usage), 3) the means by which the information is gathered (monitoring), and 4) on-board equipment and human implemented processes (systems) to fulfil an objective (Land, 2001). This subject is relevant in the context of this paper, as HUMS is an approach which is already used for capitalizing data collected during the use phase for reconfiguration purposes. Data gathered by HUMS can be used to reconfigure systems; either online by means of fault detection and isolation (Samy *et al.*, 2011), or offline by collecting data to perform support operations. Offline reconfiguration includes corrective and predictive maintenance (Ferreiro *et al.*, 2016).

2.4 Data-driven design and model learning

Further developments towards a Data-driven and Model-based Design (DMD) paradigm is needed (Tripakis, 2018). This new system design paradigm suggests to benefit from artificial intelligence and machine learning to exploit data from all types of sources. These novel techniques and tools extend the system design process for learning models from data. We consider the DMD literature, as it deals with collecting data from all lifecycle phases and exploiting them in the design phase. In data-driven design, the sources of data can be different. System designers might need to integrate new technologies into their design models. To explore new design opportunities and to solve uncertainty on what to design, researchers underline the importance of using patent search (Luo *et al.*, 2017; Song and Luo, 2017). Data from legacy systems or deployed systems is precious as it reflects the system's performance in a real-world context. Customer reviews and comments on products and systems can give new ideas for system development (Jiang *et al.*, 2017). When designing new systems, engineers usually seek for performance optimisation. The optimisation problem through a parameter change has been addressed in different studies (Chattopadhyay *et al.*, 2017; Zhang *et al.*, 2017).

2.5 Synthesis and research gap

The previously discussed literature on HUMS underlines the importance of considering use phase data to support the operations phase (i.e., preventive and predictive maintenance activities). Using these data for designing systems configurations appears to be missing in the literature treating MBSE and reconfiguration as they rely mainly on simulation data. Data-driven design and model learning domains make use of use phase data for new product development. However, these domains do not close the loop back to the design phase. More specifically, the data is not propagated back for re-organizing the system configurations (reconfiguration), reflecting change requests captured during the use phase. Thus, we attempt to address this gap in this paper by proposing a reconfiguration framework that uses different sources of data including use phase data. We argue that these data can be analysed to capture necessary changes and propagate them into the existing models to allow their reconfiguration (As detailed in section 5).

3 RESEARCH METHODOLOGY

This research is a collaboration with a large international company developing systems (aerospace, space, ground transportation, defence and security) for military and civil applications. The research conducted is action-based (Ottosson, 2003). This means that at least one of the researchers is involved in industry and is examining the practices followed within the company. The research methodology is based on the exploration of the current literature via the examination of papers and existing norms and standards, supported by data collection. Blessing and Chakrabarti (2009) highlight the importance of observation and data gathering (descriptive study) in order to identify the current challenges within an industrial context. The data collection is based upon the data triangulation: interviews, direct

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

observations and company reference documents. An extensive analysis of possible gaps between the existing literature and identified industry needs has been conducted.

3.1 Documentation exploration and observation

Internal company reference documents, best practices, the engineering guide related to the configuration and reconfiguration process were investigated and analysed. This was done in order to better understand the configuration/reconfiguration process as imagined and the process "as is", i.e. implemented concretely in the design process. Over a period of six months, we observed five different meetings and workshops where the configuration/reconfiguration problem has been discussed. Furthermore, brainstorming meetings as well as regular progress meetings with experts were conducted in order to discuss and review observations.

3.2 Qualitative interviews

To identify challenges, limitations, and opportunities, interviewing can be used to support engineering design research (Summers and Eckert, 2013). Although the effort needed to conduct and analyse the interviews is not trivial, this method allows interviewees to express naturally their concern without influencing them. The selected interviewees have different levels of involvement in system management and the configuration/reconfiguration process. Since the considered company deals with different types of systems in various operational contexts, the identified persons were classified into two categories: 1) people working in transversal activities and 2) subject matter experts. 18 experts with different roles have been interviewed (Table 1).

Table 1: Experts interviewed

Role	Number
System architects	10
Expert on engineering processes focusing on Norms & Standards	1
Expert on engineering processes focusing on Integration, verification, validation, qualification and testing (IVVQT)	1
Expert on engineering processes focusing on reliability, availability, maintainability and testability (RAMT)	1
Manager of future services	1
Modelling and Artificial intelligence expert	1
Research & Technology Engineers	2
Software development Engineer	1

To ensure objectivity, the interview has been designed according to a structured list of questions. A set of 16 questions has been prepared. The objective of the interviews is to find out the definitions related to system management (including System Configuration and System Reconfiguration) in terms of artefacts and processes that govern system or product life cycle activities, such as Systems Engineering, manufacturing and in-service operations. Moreover, questions about the different methods and tools used in the configuration management and System Reconfiguration processes have been included in our survey. The interviews had the following sequence: we started the interview with an introduction to our research work. This was followed by some warm up questions and the main body of the interview. The interview has been concluded with a cool off and wrap up. The interviews were different in terms of duration, ranging from 46 minutes up to 126 minutes with an average of 76 minutes. To avoid neglecting important facts and information audio-recording was preferred to note taking. The audio files then have been used to fully transcribe the interviews. Following the transcription, the coding process comes where we went through all textual data in a systematic way. Declarations of the interviewees were clustered into 16 categories, covering configuration and reconfiguration during different life cycle phases, and also a short biography and additional remarks. In order to avoid biases, each interviewee has been asked to verify and validate what has been discussed. The coding has been also done and validated by 3 additional persons that have not participated in the interviewing process in order to avoid biases in our understanding. This has been done also for category identification. Category identification has been afterwards presented to a larger

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

committee (10 people including Director of research in Systems Engineering and several experts in systems engineering) who have confirmed the observations coming from the field.

4 CONFIGURATION PROCESS

The objective of the field study was to understand and model the existing configuration process. Related difficulties and challenged have been identified and confirmed with stakeholders of this process. Section 0 gives an overview of the current configuration process. Section 0 highlights the challenges within the configuration process. In section 4.3 we propose to extend the configuration process.

4.1 Modelling of the current configuration process

In the current practices and system engineering process in the company, the system technical specification document should define the different configurations of the product or the solution that is delivered to the customer. The configuration process described in this section details the steps for how to construct this document. The configuration process (represented by the black part in Figure 1) starts by understanding the customers' needs. For this, system architects translate the Operational Concept Document (OCD), which contains use case scenarios, into requirements in order to define standard configurations. The objective is that the system has one of these configurations at any time during its life cycle. On the other hand, domain models (business, logistics, manufacturing, regulations, etc. provided by stakeholders) are used to define internal or external triggers, events or changes, yielding in a new system configuration. The defined configurations (system modes) and the corresponding triggering events are then described using States & Modes diagrams. The behaviour of the system is then described for the different configurations. After this, a verification for the completeness and correctness of configurations is done. Once the obtained configurations are in agreement with the requirements, the configurations will be validated. Finally, the System Specification Document will be issued. If the validated configurations are different from the pre-defined ones, an engineering change request (ECR) will be issued in order to change the requirements for facilitating new product development.

4.2 Challenges within the configuration process

As discussed in the previous section, system architects and designers use operational scenarios to capture the essential needs of the end user when using the system. They define the standard configurations contractually with the end user company based on workshops and brainstorming sessions. However, this activity is found to be insufficient because of the lack of imagination, information as well as uncertainty that is related to the design process. This leads to limited creativity in defining configurations to make available for the operations phase. In this context, the major difficulty that has been raised by actors in this process is "how to pre-define the standard configurations and the triggers without relying completely on hints from end users who will be exploiting the system?".

To allow generating relevant configurations, a reliable knowledge base containing technical and operations data is needed. The elements composing the system need to be managed in terms of resources to satisfy system functions. Configuration optimisation and selection with regard to functional and non-functional criteria are also difficulties that need to be addressed. Moreover, incompleteness and incorrectness of configurations can be a real issue, as there is no generic and automatic tool to support configuration generation. This activity is not trivial and demands a considerable engineering effort, especially when considering systems with increasing levels of complexity. To consider large number of configurations implies incompleteness of models. Furthermore, existing tools in industry do not allow for modelling system transitions and behaviour.

4.3 Proposal of an extended configuration process

To overcome the difficulties underlined in the previous section, one can consider integrating real data from the use phase into the design phase relevant for defining configurations. We propose to observe the system and its environment as well as its operational context during usage. System analysts can then use observed data in order to define system failures, peaks, needs or simply abnormal behaviours the system might need to handle. The impact of these behaviours is then evaluated, and a change of

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

configurations (reconfigurations) and domain models is requested. We named this proposal “data capitalisation” in the reconfiguration process (Red part in Figure 1). In this context, the originality of our proposal lies in propagating the changes into system configurations based on real data collected from the use phase, and not only hints from end users about their needs.

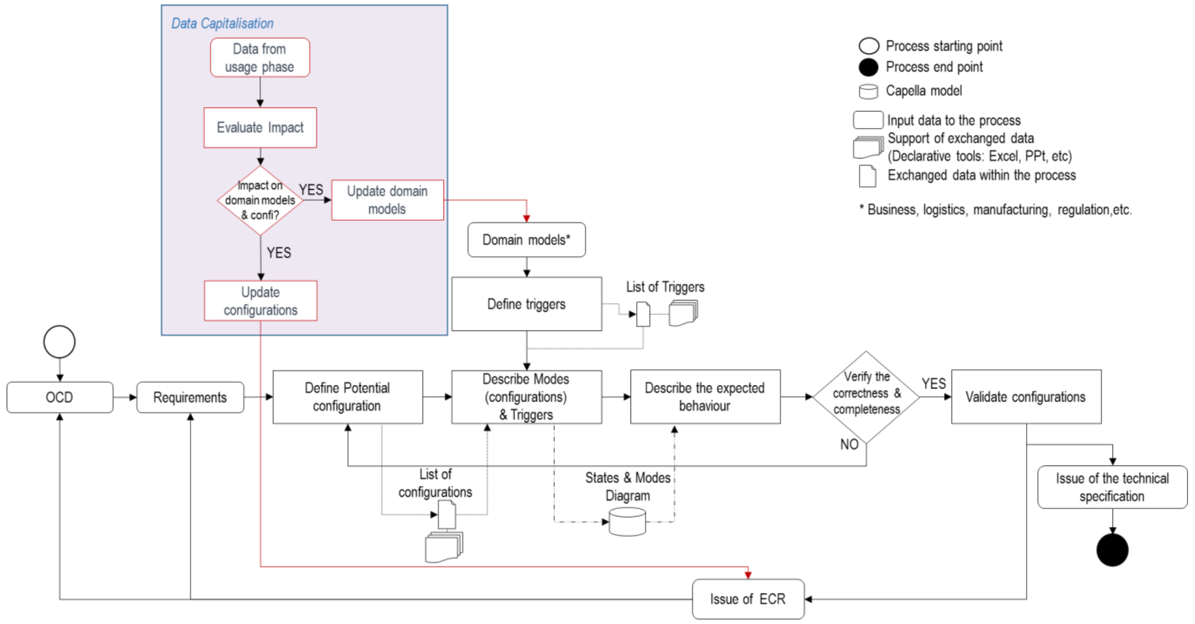
Data Capitalisation
Data from usage phase
Evaluate Impact
Impact on domain models & confi?
YES
Update domain models
YES
Update configurations
Domain models*
Define triggers
List of Triggers
OCD
Requirements
Define Potential configuration
Describe Modes (configurations) & Triggers
Describe the expected behaviour
Verify the correctness & completeness
YES
NO
Validate configurations
List of configurations
States & Modes Diagram
Issue of the technical specification
Issue of ECR
Process starting point
Process end point
Capella model
Input data to the process
Support of exchanged data (Declarative tools: Excel, PPt, etc)
Exchanged data within the process
* Business, logistics, manufacturing, regulation,etc.

Figure 1: Configuration and reconfiguration process with data capitalisation

The difficulties that are related to this proposal are the following ones:

- Data collection: Data availability and accessibility can be a real issue, more specifically if data are shared across stakeholders.
- Data storage: Data capitalisation requires the storage a considerable quantity of data. Due to cost and space limitation, a strategy for data storage is needed in order to optimise available resources.
- Data verification and uncertainty: Data from the use phase can have different sources. Independent or embedded sensors provide internal technical data about the system, while external observations from operators or maintenance teams uncover information about the performance. Indeed, the uncertainty about the collected data is variable. Data verification is necessary before starting the reconfiguration process to address different levels of trust.
- Data combination: Data capitalisation implies combining a priori and a posteriori data. From the use phase, data are collected and analysed to propose new configurations. Data combination is not trivial and requires in-depth data analysis to consider the degrees of uncertainty.
- Data modelling: Data from the use phase need to be structured and modelled to allow reconfiguration. Nevertheless, modelling can be a real issue, especially when considering different types of systems (i.e. continuous or discrete). Moreover, multi-level system modelling might be needed. Depending on the context considered, different levels of systems are involved in the reconfiguration process: the system of interest (the system fulfilling the operational mission); but also enabling systems for development, manufacturing, maintenance, health monitoring, supervision, and control.

5 RECONFIGURATION FRAMEWORK

The previously discussed literature (Section 2) shows a gap in capturing changes from use phase data to close the loop back to the design phase by propagating the changes into existing models to re-organize system configurations (reconfiguration). With regard to the identified gap and the extension of the reconfiguration process (proposed in section 4.3), we propose a generic reconfiguration framework (Figure 2) that uses different sources of data including the use phase data. This framework suggests to generate feasible configurations of the system under development. For this purpose, the framework uses different sources of data and a Configuration Manager.

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

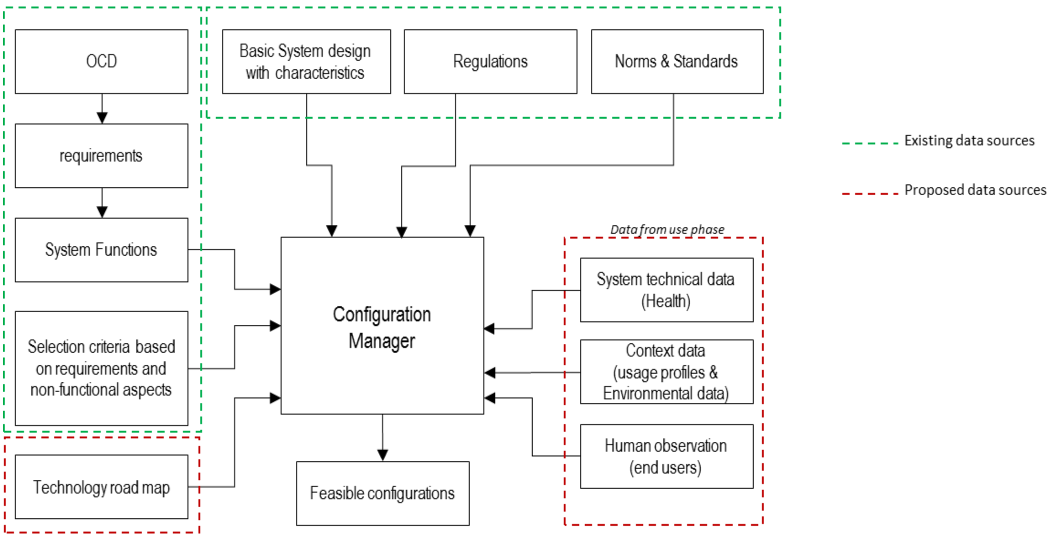
OCD
requirements
System Functions
Selection criteria based on requirements and non-functional aspects
Technology road map
Basic System design with characteristics
Regulations
Norms & Standards
Configuration Manager
Feasible configurations
Data from use phase
System technical data (Health)
Context data (usage profiles & Environmental data)
Human observation (end users)
Existing data sources
Proposed data sources

Figure 2: Generic reconfiguration framework

5.1 Data sources

Input data considered include data already existing in industry, supplemented by data linked to our proposal. Input data sources that are existing in industry and used to define configurations are:

- Basic System design with characteristics;
- System Functions: OCD contains elements that are important to define configurations. Hence, OCD gives the system functional and non-functional requirements. From these requirements we can deduce System Functions. The system mission's change (given by the OCD) is an important driver for a reconfiguration. In this case, we might need to add or remove new system capabilities. Consequently, new System Functions will lead to a change in the configuration;
- Selection criteria: Non-functional requirements can provide selection criteria that can be used to assess and select configurations;
- Regulations: Experts have also emphasised that configurations should follow regulations, both national and international;
- Norms & Standards: For the system to conform to norms and standards, any change of them might lead to a need for reconfiguration. This is particularly relevant for certification.

- Data sources that we propose to add are:
- Technology road map: this data has been discussed in the literature as an important element to deal with technology evolution. The emergence or obsolescence of technologies (given by the Technology road map) may require configurations to be modified;
- Data collected from the use phase: feedback data from the use phase can provide knowledge about the real system behaviour. These data include:
- System technical data which gives indications on which elements or modules to optimize (re-design);
- Context data containing both usage profiles and environmental data: which can help to optimize the system configurations to better meet the requirements specified by the client taking into account the environmental conditions and user behaviours;
- Human observation: Human observation in terms of opinions and views can also impact the way the new configurations are developed.

5.2 Configuration manager

The Configuration Manager uses data collected from the use phase and other sources of change. The impact on the Basic System design, represented by a model, is then evaluated. If a decision on a change is validated, it is propagated in the system model to generate new configurations. The resulting configurations are then simulated and tested to verify their behaviour. In the subsequent optimisation step the configuration elements are optimised with regard to the mission. After that, configurations selection is made based on system requirements and non-functional aspects.

5.3 Framework validation

The validation process has started and is currently undergoing. The internal validation by experts represents the first step in the validation process. For this purpose, we have conducted interviews and workshops with participants from industry. The next step is a scenario-based validation, where the framework is applied to use cases from industry. This step is essential for capturing particularities from different operational contexts and system typologies, i.e., closed system, platform, distributed systems and systems of systems, relevant for our industrial context.

6 CONCLUSION & FUTURE WORK

The work presented in this paper is based upon a field study that has been conducted in an international aerospace, space, ground transportation, security, and defence company. This field study allowed us to model the reconfiguration process and to capture the challenges within this process. Among these challenges is the capitalisation of "use phase data" to support the reconfiguration process. Integrating use phase data in the design phase has not yet been treated in the existing literature. To address this gap, we proposed a reconfiguration framework for integrating data collected from the use phase. The framework aims at supporting the design of future IoT (internet of things) and connected systems which generate large amounts of data in the use phase. Future work should concentrate on testing and validating the framework with use cases representing different contexts and system types (Radar, plane, Air traffic control). In addition, studying the techniques used in learning models (machine learning techniques) and rule-based reasoning to build the Configuration Manager and to allow reconfiguration seem promising.

REFERENCES

Acheson, P., Dagli, C. and Kilicay-Ergin, N. (2013), "Model based systems engineering for system of systems using agent-based modeling", *Procedia Computer Science*, Vol. 16, Elsevier B.V., pp. 11–19.

Alsafi, Y. and Vyatkin, V. (2010), "Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing", *Robotics and Computer-Integrated Manufacturing*, Vol. 26 No. 4, pp. 381–391.

Blessing, L.T.M. and Chakrabarti, A. (2009), *DRM, a Design Research Methodology, Focus*, Vol. 1, available at: https://doi.org/10.1007/978-1-84882-587-1.

Chattopadhyay, P., Mondal, S., Bhattacharya, C., Mukhopadhyay, A. and Ray, A. (2017), "Dynamic Data-Driven Design of Lean Premixed Combustors for Thermoacoustically Stable Operations", *Journal of Mechanical Design*, Vol. 139 No. 11, p. 111419.

Deciu, E.R., Ostrosi, E., Ferney, M. and Gheorghe, M. (2005), "Configurable product design using multiple fuzzy models", *Journal of Engineering Design*, Vol. 16 No. 2, pp. 209–233.

Elsner, C., Ulbrich, P., Lohmann, D. and Schröder-Preikschat, W. (2010), "Consistent product line configuration across file type and product line boundaries", *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 6287 LNCS, pp. 181–195.

Ferguson, S., Siddiqi, A., Lewis, K. and De Weck, O.L. (2007), "Detc2007/dac-35745", pp. 1–15.

Ferreiro, S., Konde, E., Fernández, S. and Prado, A. (2016), "INDUSTRY 4.0 : Predictive Intelligent Maintenance for Production Equipment", *European Conference of the Prognostics and Health Management Society*, pp. 1–8.

Haris, K. and Dagli, C.H. (2011), "Adaptive reconfiguration of complex system architecture", *Procedia Computer Science*, Vol. 6, pp. 147–152.

Haubelt, C., Teich, J., Richter, K. and Ernst, R. (2002), "System design for flexibility", *Proceedings -Design, Automation and Test in Europe*, DATE, pp. 854–861.

Hernández, C., Fernández, J.L., Sánchez-Escribano, G., Bermejo-Alonso, J. and Sanz, R. (2015), "Model-Based Metacontrol for Self-adaptation", *International Conference on Intelligent Robotics and Applications*, Springer, pp. 643–654.

ISO/IEC/IEEE/15288 (2015), *Systems and Software Engineering- System Life Cycle Processes*.

Jiang, H., Kwong, C.K. and Yung, K.L. (2017), "A methodology for predicting future importance of customer needs based on online customer reviews", *Journal of Mechanical Design*, No. c, available at: https://doi.org/10.1115/1.4037348.

Khederzadeh, M. and Beiranvand, A. (2018), "Identification and Prevention of Cascading Failures in Autonomous Microgrid", *IEEE Systems Journal*, Vol. 12 No. 1, pp. 308–315.

Krichen, F. and Zalila, B. (2011), "Towards a Model-Based Approach for Reconfigurable DRE Systems Towards a Model-Based Approach for Reconfigurable DRE Systems", *European Conference on Software Architecture*, Springer, pp. 295–302.

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press

Land, J.E. (2001), “HUMS -The Benefits --Past, Present and Future”, In *Aerospace Conference*, IEEE? Vol. 6, pp. 3083–3094.

Luo, J., Yan, B. and Wood, K. (2017), “InnoGPS for Data-Driven Exploration of Design Opportunities and Directions: The Case of Google Driverless Car Project”, *Journal of Mechanical Design*, Vol. 139 No. November, pp. 1–13.

Martin, M. V. and Ishii, K. (2002), “Design for variety: developing standardized and modularized product platform architectures”, Research in Engineering Design, Vol. 13 No. 4, pp. 213–235.

Ottosson, S. (2003), “Participation action research-”, *Technovation*, Vol. 23 No. 2, pp. 87–94.

Provan, G. and Chen, Y.-L. (1999), “Model-based diagnosis and control reconfiguration for discrete event systems: an integrated approach”, *Proceedings of the 38th IEEE Conference on Decision and Control*, Vol. 2, pp. 1762–1768.

Regulin, D., Schutz, D., Aicher, T. and Vogel-Heuser, B. (2016), “Model based design of knowledge bases in multi agent systems for enabling automatic reconfiguration capabilities of material flow modules”, *IEEE International Conference on Automation Science and Engineering*, Vol. 2016–Novem, pp. 133–140.

Samy, I., Postlethwaite, I. and Gu, D.W. (2011), “Survey and application of sensor fault detection and isolation schemes”, *Control Engineering Practice, Elsevier*, Vol. 19 No. 7, pp. 658–674.

Sanz, R., Hernandez, C. and Bermejo, J. (2014), “Improved Resilience Controllers Using Cognitive Patterns”, *IFAC Proceedings Volumes*, Vol. 47, Elsevier, pp. 683–688.

Saxena, T., Dubey, A., Balasubramanian, D. and Karsai, G. (2010), “Enabling Self-Management by Using Model-Based Design Space Exploration”, *Seventh IEEE International Conference and Workshops on Engineering of Autonomic and Autonomous Systems*, pp. 137–144.

Siddiqi, A. and de Weck, O.L. (2008), “Modeling Methods and Conceptual Design Principles for Reconfigurable Systems”, *Journal of Mechanical Design*, Vol. 130 No. 10, p. 101102.

Siddiqi, A. and de Weck, O.L. (2009), “Reconfigurability in planetary surface vehicles”, *Acta Astronautica*, Vol. 64 No. 5–6, pp. 589–601.

Siddiqi, A., De Weck, O.L. and Iagnemma, K. (2006), “Reconfigurability in planetary surface vehicles: Modelling approaches and case study”, *JBIS - Journal of the British Interplanetary Society*, Vol. 59 No. 12, pp. 450–460.

Simpson, J.J. and Dagli, C.H. (2008), “System of systems architecture generation and evaluation using evolutionary algorithms”, *IEEE International Systems Conference Proceedings, SysCon 2008*, pp. 64–69.

Simpson, T.W., Maier, J.R. and Mistree, F. (2001), “Product platform design: Method and application”, *Research in Engineering Design - Theory, Applications, and Concurrent Engineering*, Vol. 13 No. 1, pp. 2–22.

Song, B. and Luo, J. (2017), “Mining Patent Precedents for Data-Driven Design: The Case of Spherical Rolling Robots”, *Journal of Mechanical Design*, Vol. 139 No. 11, p. 111420.

Summers, J.D. and Eckert, C.M. (2013), “*Design Research Methods: Interviewing*”, *Workshop in ASME Conference*, Portland, Oregan, USA. p. 2013.

Tripakis, S. (2018), “Data-driven and model-based design”, *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, IEEE, pp. 103–108.

Umeda, Y., Kondoh, S., Shimomura, Y. and Tomiyama, T. (2005), “Development of design methodology for upgradable products based on function-behavior-state modeling”, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM*, Vol. 19 No. 3, pp. 161–182.

Walden, D. and Roedler, G. (2015), *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 4th Edition, available at: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118999401.html.

de Weck, O.L. and Suh, E.S. (2006), “Flexible Product Platforms: Framework and Case Study”, Vol. 1, *32nd Design Automation Conference, Parts A and B*, Vol. 2006, ASME, pp. 783–799.

Wichmann, A., Jäger, S., Jungebloud, T., Maschotta, R. and Zimmermann, A. (2015), “System architecture optimization with runtime reconfiguration of simulation models”, *9th Annual IEEE International Systems Conference, SysCon 2015 - Proceedings*, pp. 660–667.

Wu, D., Zhang, L.L., Jiao, R.J. and Lu, R.F. (2013), “SysML-based design chain information modeling for variety management in production reconfiguration”, *Journal of Intelligent Manufacturing*, Vol. 24 No. 3, pp. 575–596.

Wymore, A.W. (1993), *Model-Based Systems Engineering*, CRC press.

Zhang, L., Chu, X., Chen, H. and Xue, D. (2017), “Identification of Performance Requirements for Design of Smartphones Based on Analysis of the Collected Operating Data”, *Journal of Mechanical Design*, Vol. 139 No. 11, p. 111418.

https://doi.org/10.1017/dsi.2019.268 Published online by Cambridge University Press