

# Formal Information Model for Representing Production Resources

Niko Siltala<sup>(✉)</sup>, Eeva Järvenpää, and Minna Lanz

Department of Mechanical Engineering and Industrial Systems,  
Tampere University of Technology, Tampere, Finland  
{niko.siltala,eeva.jarvenpaa,minna.lanz}@tut.fi

**Abstract.** This paper introduces a concept and associated descriptions to formally describe physical production resources for modular and reconfigurable production systems. These descriptions are source of formal information for (automatic) production system design and (re-) configuration. They can be further utilized during the system deployment and execution. The proposed concept and the underlying formal resource description model is composed of three different description levels, namely Abstract Resource Description (ARD), Resource Description (RD) and Resource Instance Description (RID), each having different scope and objectives. This paper discusses in details the content and differences between these description levels.

**Keywords:** Production resource model · Production resource description · Production system design · Reconfigurable manufacturing system · RMS · XML

## 1 Introduction

The requirements on production systems are continuously being shifted towards higher flexibility and adaptability. Increasing volatility in the economies, shortening innovation and product life cycles, and ever increasing number of variants, call for production systems, which comply with these changing demands. There is a need for rapidly responding production systems that can timely adjust to the required changes in processing functions and production capacity. System reconfiguration is required on three levels: physical, logical and parametric [1]. System reconfiguration is the enabler for re-use and sustainability of production resources.

Despite the high efforts towards reconfigurable production systems and standardization activities focusing on unification of mechanical as well as communication and control interfaces, reconfiguration of assembly systems is still rare in real factories. The usual business today, when the product model changes, is to scrap the existing resources and build a new assembly system from a scratch. This is due to high engineering, integration, and programming efforts and skills

needed to re-configure the existing system, as well as uncertainties related to the needed effort. One of the main reasons for infeasibility of reconfiguration is the lack of sufficient and accurate information about the production resources, and their capabilities associated to the current system, its life cycle, and usage history [2]. In addition to Hardware (HW) and Software (SW) interfaces, efficient methodologies, tools and information models are needed to support planners and engineers in the reconfiguration process, and also to allow logical and parametric reconfiguration to take place autonomously while the system is running.

The European Commission funded project ReCaM [3], started in November 2015, aims to find solutions for the above mentioned issues. It targets to develop a set of integrated tools for rapid and autonomous reconfiguration of production systems, integrated with the existing production planning and scheduling tools - MES. The ReCaM approach is based on intelligent plug-and-produce capable self-describing Mechatronic Objects (MOs), which are able to auto-program and self-adjust to the required task. The formal resource descriptions are proposed to capture comprehensively and thoroughly the characteristics of a production devices, providing a foundation for rapid creation of new system configurations.

The objective of this paper is to open the insights of the resource description model and its formal resource descriptions. The paper is organized as follows: Sect. 2 discusses other existing resource description models. Section 3 represents the proposed three level resource description model. Finally, Sect. 4 discusses the future developments and concludes the paper.

## 2 Existing Production Resource Descriptions

There exist some device related descriptions, however having a slightly different focus. Electronic Device Description Language (EDDL) [4–6] and its continuation Field Device Integration (FDI) [7, 8] are used to describe process automation components, having focus on lower level elementary components. The various fieldbus device descriptions for e.g. ProdiBus/-Net, DeviceNet and EtherCAT, all fall into same category, having focus even lower level components and configurations for control systems.

Contrary, AutomationML [9–11] has focus in system integration and representation. The basic assumption of the AutomationML is that the templates of modules are injected to the system description, but they are always customised and modified extensively for the project’s purposes. This means that the modules are not stable and reusable entities, but are engineered or customised over and over again. Furthermore, the concept requires that the internal design and implementation of a module is revealed extensively and completely for the system design. Nevertheless, the generality of the approach brings the expression power for this concept. It is more probable that the same concept can cover all unexpected situations arriving in realistic production system design landscape. In context of AutomationML, there exist some publications representing aspects of re-usable components like [12] with mechatronic units and [13, 14] with Smart-Components. These both get close to objectives of the work reported in this



for a collection of similar kind of production resources (e.g. grippers, feeders, glue dispensers, welding devices, etc.). In other words, ARD is a generalisation, which can be specialised as a physical production resource. The objective of ARD is to enable compatibility, interchangeability and connectivity between different production resources, through harmonisation and grouping of interfaces specifications across different types of resources. For example, the task of ARD is to ensure that a robot can be connected to a base, and that a gripper can be connected to a robot, and interchanged with another gripper. The purpose of ARD is to provide harmonisation over RDs and its content is controlled by a user group(s) or standardisation bodies.

ARD cannot be directly instantiated as a physical resource, but it is composed of one or more Profiles (See Fig. 1), which are reflecting production resources. Profile is an integral and inseparable part of ARD and cannot exist alone outside of ARD. Profile defines a reusable construction block of definitions, a structure which is used to specify the detailed section of an ARD. It includes information related to interfaces, capabilities, properties and other features that are composing the generalisation for a set of production resources. This way the information is defined only in one place, which then can be referenced and re-used in other parts of the descriptions. This improves the quality and consistency of the descriptions, as e.g. typing or mishaps related errors can be reduced, consistency of information can be increased, and the maintenance of descriptions is facilitated.

Two kinds of Profiles exist - *abstract* and *implementable* ones, but only the latter can be instantiated as RD and its physical implementation. A Profile can be built from  $N$  other Profiles with concepts of inheritance or referencing. Figure 2 provides an example of inheritance hierarchy in case of *Gripper* ARD. The content of Profile characterises the module, and can be used for comparison, evaluation, and selection purposes. Included interfaces and capability definitions are the enablers for mating and fitting the devices together, and making sure that the composition of production modules is able to provide requested functionality for the production task. The interfaces include comprehensive information from mechanical and electrical to service and communication. Generally, it captures all information needed for connecting a production module to other ones or to external world. As a rule, Profiles are introducing the existence and purpose of Properties, but not yet fixing the property values. This provides harmonisation and predictability across the RD definitions, which are later setting the values for these properties.

The *Gripper* ARD and its *Force controlled 2-finger gripper* Profile are used as a practical example. It provides an illustration of re-use, construction block behaviour (i.e. define only once), and use of the two types of Profiles. Figure 2 illustrates this example and aforementioned characteristics of Profiles. Each rectangle is a Profile entity. The rectangles with white background and italics are abstract Profiles, of which physical module cannot be created of, and ones with grey background are implementable Profiles, from which the production modules and corresponding RD can be created of. The implementable Profile of our



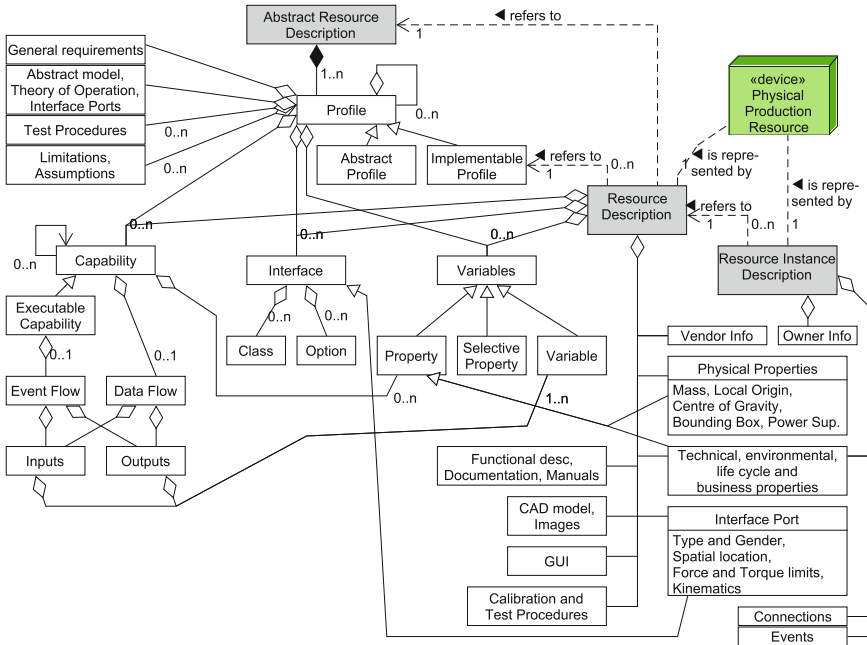


Fig. 3. Resource description model

### 3.3 Resource Instance Description

Condition and capabilities of the resources evolve during their individual life cycles and usages. Resource Instance Description (RID) is a digital representation of an individual physical instance of a resource. It carries the resources' current state and historical data events – it is an accumulating information storage. It appends the RD with information that cannot be generalised over all instances of the same resource type, but is specific to one instance only. For instance, if the capability or life cycle parameters (such as Mean Time Between Failure (MTBF) or tolerances) are changed during the resource life cycle, the RID will contain the updated information. The RID should travel all the time with the physical production resource.

### 3.4 Example Instances of Resource Description Model

Table 1 summarises the characteristics of different top level descriptions composing the Resource Description Model, by connecting these with compacted definition and an illustrative example of each description. ARD represents specific technologies such as grippers, axis-systems or feeders, and collect all associated Profiles together. Profile provides generalised specification of specific kind of entities, such as 2-finger grippers. These are the parts composing a single ARD. RD turns focus to the module providers (vendor  $V_A$ ). They provide a detailed

**Table 1.** Comparison of main entities of production resource model

Entity	Definition	Example
ARD	is container for Profiles	Grippers ARD
Profile	is description of abstract module and part of ARD	Abstraction of a 2-finger gripper
RD	is digital representation of physical module. Derived from one Profile	2-finger gripper from vendor $V_A$ w/ type: T1
RID	is digital representation of a specific module instance	Type: T1 gripper w/ serial: SN123

description of their module (third row in Table 1). This description respects the definitions made in aforementioned Profile and ARD. Finally, when vendor  $V_A$  produces physical entity of such gripper of type T1, they assign a serial number to this piece of HW and create a RID for it (fourth row in Table 1).

## 4 Discussion and Conclusions

This paper presented an overview of a three level model to describe production resources. This model and descriptions can be utilised in various phases of Reconfigurable Manufacturing System (RMS) production system design, reconfiguration and commissioning. How to apply these models is discussed in [18, Ch.7].

In the due course of the ReCaM project, there are already identified improvements and new requirements related to the production resource descriptions. These are: (a) further development, synchronisation, and tighter integration with the Capability model; (b) detailed definition of Executable Capability concept, i.e. interfaces for the controls and SW services; (c) improvements on mechanical interface description in relation to interfaces connecting the resource to the produced items; (d) including generalised Graphical User Interface (GUI) definitions into descriptions, from which resource specific GUIs can be generated automatically; and finally, (e) editors for creating and modifying the resource description files. These improvements are in progress. There is interest to research if this model, especially RD, could be translated to format of AutomationML. The presented models and proposed improvements will be validated during the ReCaM project.

**Acknowledgement.** This research has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 680759 [3].

## References

1. ElMaraghy, H.A.: Flexible and reconfigurable manufacturing systems paradigms. *Int. J. Flex. Manuf. Syst.* **17**, 261–276 (2005)

2. Järvenpää, E.: Capability-based adaptation of production systems in a changing environment. Ph.D. dissertation. Tampere University of Technology, 190 p. (2012)
3. ReCaM - Rapid reconfiguration of flexible production systems through capability-based adaptation, auto-configuration, integrated tools for production planning - project, EU Horizon 2020, GA No 680759. <http://www.recam-project.eu>
4. Merget, O.: FDT and EDDL: the electronic integration of field devices. *Comput. Control Eng.* **14**(5), 22–23 (2003). doi:[10.1049/ccc:20030504](https://doi.org/10.1049/ccc:20030504)
5. IEC: IEC 61804-3: 2006. Function blocks (FB) for process control - Part 3: Electronic Device Description Language (EDDL) (2006)
6. FDI Cooperation: Electronic Device Description Language. <http://www.eddl.org/>
7. Neumann, P., Simon, R., Diedrich, C., Riedl, M.: Field device integration. In: 8th International Conference on Emerging Technologies and Factory Automation (ETFA 2001), Proceedings (Cat. No.01TH8597), pp. 63–68. IEEE (2001)
8. FDI Cooperation. <http://www.fdi-cooperation.com>
9. AutomationML/IEC.: IEC 62714-1: 2014 Engineering data exchange format for use in industrial automation systems engineering - automation markup language - Part 1: Architecture and General Requirements (2014)
10. AutomationML. <http://www.automationml.org/>
11. Lüder, A., Hundt, L., Keibel, A.: Description of manufacturing processes using AutomationML. In: 2010 IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010), 8 p. (2010)
12. Lüder, A., Foehr, L.H.M., Wagner, T., Zaddach, J.-J., Holm, T.: Manufacturing system engineering with mechatronical units. In: IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010), 8 p. Bilbao, Spain (2010)
13. Bartelt, M., Schyja, A., Kuhlenkötter, B.: More than a Mockup - SmartComponents: reusable fully functional virtual components from scratch. *Prod. Eng.* **8**, 727–735 (2014)
14. Schyja, A., Bartelt, M., Kuhlenkötter, B.: From conception phase up to virtual verification using AutomationML. *Procedia CIRP.* **23**, 171–177 (2014)
15. Siltala, N., Hofmann, A.F., Tuokko, R., Bretthauer, G.: Emplacement and blue print-an approach to handle and describe modules for evolvable assembly systems. In: Hideki, H. (ed.) Proceedings of 9th International Symposium on Robot Control (SYROCO 2009), pp. 86–91. IFAC, Gifu, Japan (2009)
16. Siltala, N., Tuokko, R.: Emplacement and blue print-electronic module description supporting evolvable assembly systems design, deployment and execution. In: Huang, G.Q., Mak, K.L., Maropoulos, P.G. (eds.) Proceedings of 6th CIRP-sponsored International Conference on Digital Enterprise Technology (DET), pp. 773–788. Springer, Heidelberg (2009)
17. Siltala, N., Tuokko, R.: Use of electronic module descriptions for modular and reconfigurable assembly systems. In: 2009 IEEE International Symposium on Assembly and Manufacturing, pp. 214–219. Suwon, Korea (2009)
18. Siltala, N.: Formal digital description of production equipment modules for supporting system design and deployment. Ph.D. dissertation. Tampere University of Technology, 211 p. (2016)
19. Järvenpää, E., Siltala, N., Lanz, M.: Formal resource and capability descriptions supporting rapid reconfiguration of assembly systems. In: 2016 IEEE International Symposium on Assembly and Manufacturing. USA, Texas (to be published)
20. Emplacement Web Service, <http://resourcedescription.tut.fi/EmplacementWS/>
21. Siltala, N., Tuokko, R.: A web tool supporting the management and use of electronic module descriptions for evolvable production systems. In: 2010 IEEE International Symposium on Industrial Electronics, pp. 2641–2646. Bari, Italy (2010)