Model-Based Management of Asset Information

David Kampert,* Ulrich Epple
Chair of Process Control Engineering, RWTH Aachen University
{kampert, epple}@plt.rwth-aachen.de

Martin Mertens†
Ineos Köln GmbH
martin.mertens@ineos.com

1 Introduction

Software tools for engineering tasks typically aim at well-defined aspects of an overall project, for which they employ domain-specific models. Managing information on assets is an important cross-sectional task, which makes the exchange of information valuable to many domains. In order to enable software tools to exchange information, the underlying meta-model needs to be open and common. This primarily accounts for the contents of the meta-model (*what* is modeled), which is a different issue than the question *how* the model is technically represented. Hence, in this article the combination of an appropriate meta-model for asset information along with its integration into a service-oriented architecture is discussed from a conceptual perspective. It is not intentended to provide details of a possible implementation.

2 Meta-Models in Information Exchange

Model-driven approaches for the engineering of automation systems allow handling individual tasks in a generic way. This is accomplished by common meta-models, which define the syntax and semantics of individual models generically and to a reasonable extent. In case of P&I-Diagrams, for example, the meta-model contains graphical symbols for the representation of various devices in individual plant models, which may provide information on represented devices up to a certain level of detail. Software tools usually have an implicit knowledge on the meta-model, but interoperation of different software tools has the prerequisite of using an open meta-model, i.e. common knowledge on structure and semantics of the exchanged information. Common technical means to represent the information is also mandatory, but may be answered by respective model transformations. Standardized meta-models on the logical level are thus the basis for interoperation of engineering tools, independent from the technical representation (e.g. files, databases, ontologies,...).

Finding an appropriate meta-model for information exchange means finding the coarsest-grained building-blocks that are common for all model instances. Here, the term “coarse-grained” not only applies to the size of information chunks, but also to structural degrees of freedom. The appropriate level is a case-by-case tradeoff and can only be found if the objects that are modeled are analyzed carefully.

3 Meta-Models for Asset Information Management

In industrial automation, models under consideration are often models of technical resources, which are intended to reflect the resources’ properties. Though there are certainly other important kinds of models (e.g. functional or structural models), descriptions based on properties are among the most frequently met paradigm for providing information on assets. It is therefore an important goal to structure them in a common meta-model. For limited application purposes this was accomplished by defining meta-models for the description of technical devices, of which IEC 61360 [3], IEC 62264 [2] and NE 100 [5] are prominent examples. These aim at standardized specifications of devices as needed in e-commerce, i.e. information on device classes. They are neither intended to be descriptions of assets in general, nor do they address the management of individual assets in conjunction with device classes. Consequently, these solutions are applied in electronic catalogs, but not on a larger scale for information exchange on industrial assets. There is nonetheless reasonable hope that a meta-model for managing information on industrial assets can be defined, since all of the above-mentioned standards are sharing the same idea of describing assets by means of properties. The common meta-model needs to be unbiased in terms of the kind of modeled assets and business perspectives, but should incorporate the information that it is meant for describing real-world assets. However, the key requirement for successful integration of information from different sources is not to use identical data structures, but to have a common idea of the information contents. With this goal in mind, a reasonable approach to find the common ground of all models is to consider them as representations of things (of certain kinds) that have properties and property values. A corresponding meta-model was introduced in [4]. In the proposed modeling concept assets are seen to be property carriers with respective properties. These properties are classified and mutually independent characteristics that can be associated with values by means of statements, e.g.

* The author gratefully acknowledges support by the German Research Foundation (DFG).

† The author contributed to this work during his affiliation with the Chair of Process Control Engineering, RWTH Aachen University.

statements whether the value was measured, is required or is asserted [6]. The definition of individual properties is beyond the scope of the meta-model; existing property definitions may be used for this purpose. An implementation of the model is currently under development.

Interoperation of software tools benefits from the proposed meta-model because multiple domains can contribute and request information to and from one asset model. Furthermore, information may be provided on single devices or entire device classes. Different domains may maintain their point of view to assets and are yet able to integrate their information because the meta-model only incorporates the common ideas. E.g., statements regarding the current operational parameters of a pump can be made on the part of plant operation, while the pump's vendor can provide assertions on these properties for all pumps of that kind. The information can be exploited to detect inconsistencies automatically by comparing statement values, independent from knowledge about the properties' semantics. In general, any application that needs to access information on asset properties is enabled to do so on the level of granularity that is required for exactly this purpose.

4 Incorporating the Meta-Model in IT Systems

By establishing a generic meta-model for industrial assets, many important aspects of the information environment in an industrial facility can be modeled in a unified way. This is a mandatory prerequisite for information exchange between software tools, but needs a technical representation of the model in order to make data accessible. However, it must be noted that the crucial design decisions in the development of a corresponding system are taken by the meta-model's definition.

In the past years, service-oriented architectures (SOAs) have gained increasing attention and are seen to be a future paradigm of industrial IT infrastructures (see e.g. [1]). SOAs may be composed by "black-box" systems that only have a known interface, which makes common meta-models inevitable: The internal data strucutres may differ from system to system, but a common idea of the exchanged information needs to exist. In case of asset management, a valuable advantage of SOAs is the possibility to centralize information access, which supports the meta-model's strengths from the technical side. Contrary to file-based data exchange, information may be spread over multiple locations (i.e. multiple domains may contribute information), while allowing to access information using a centralized interface, which provides a single virtual model. This can be realized by a central service for accessing asset property information, which in background retrieves information from various registered sources, forming a federated database. Servers that contribute information to the overall system need to implement a standardized service interface, which allows for accessing the internal data as defined by the common meta-model.

In practice, the proposed architecture brings three major advantages towards file-based information exchange, which still is widely spread. The first one obviously is that inconsistencies due to differing file versions cannot occur. Secondly, information regarding one asset can be automatically analyzed for inconsistencies in terms of contradictory statements on identical properties. Consistency checks that involve property semantics may be implemented under domain-specific knowledge of specialized engineering tools. The third mayor advantage is the possibility to employ generic algorithms to enrich the information contents. E.g., devices that are able to fulfill a modeled role may be automatically found or contradictory statements may be detected in order to resolve conflicts in an early planning phase.

5 Conclusion

Basically, it can be concluded that an appropriate meta-model for asset information along with information exchange over service interfaces constitutes a promising design pattern for interoperation of engineering tools. It demands a common basic model, which is realizable in case of managing asset information since involved software tools are dealing with the same information environment. The way in which individual tools handle the contained information is not affected and leaves the necessary flexibility. In order to realize the proposed architecture, the remaining task is twofold: On the one hand, model elements need to be standardized in order to obtain common semantics. Existing standards may be employed for this purpose. The second part of the task is the definition of the necessary service interface, especially in terms of supported functionality and operation granularity. This is subject to ongoing research work and will be implemented in a software prototype.

References

[1] Thomas Bangemann, Christian Diedrich, Armando Walter Colombo, and Stamatis Karnouskos. SOCRADES – Service Oriented Architecture in der Automatisierungstechnik. In *Automation 2008, Baden-Baden*, pages 65–68, 2008.

[2] International Electrotechnical Commission. IEC 62264: Enterprise-control system integration – Part 1: Models and terminology, March 2003.

[3] International Electrotechnical Commission. IEC 61360: Standard data element types with associated classification scheme for electric items – Part 2: EXPRESS dictionary schema, February 2004.

[4] Ulrich Epple. Merkmale als Grundlage der Interoperabilität technischer Systeme. *at – Automatisierungstechnik*, 59:440–450, 2011.

[5] NAMUR: User Association for Automation in Process Industries. NE 100, Version 3.2: Use of Lists of Properties in Process Control Engineering Workflows, June 2010.

[6] Martin Mertens and Ulrich Epple. Plant Asset Management Functions driven by Property Models. In *ETFA 2009: 14th IEEE International Conference on Emerging Technologies and Factory Automation*, 2009.