CDL-Flex Industry Newsletter

September 2016

Editorial

Dear reader,

making engineering collaboration more effective and efficient in a heterogeneous software tool landscape is an ongoing issue for software solution providers and users. The step-by-step migration from current tools towards *AutomationML* is feasible and useful (see the *Andritz Hydro* application case). The *AML.hub* is useful for the migration towards *AutomationML* and brings considerable advantages regarding productivity, collaboration, and quality assurance. For 2017, we plan an *AML.hub* version that supports the upcoming guideline *VDMA 66415 Einheitsblatt*, which facilitates the simple participation in *AutomationML* for SMEs.

At the upcoming **AutomationML User Conference** and the **SPS IPC Drives 2016 Electric Automation – Systems and Components**, experts from the Christian Doppler research laboratory CDL-Flex at the Vienna University of Technology will present and discuss applications for industry partners based on the **AML.hub** for the versioned storage and integration of *AutomationML* data coming from heterogeneous engineering models and tools.

At the 4th *AutomationML User Conference*, scheduled for October 18-19 at Festo AG in Esslingen, Germany, researchers will present and discuss solution concepts for efficient data exchange based on the *AutomationML Hub* and collaborative review support for improving engineering projects in multi-disciplinary engineering environments.

At the *SPS/IPC/Drives 2016* exhibition, scheduled for November 22 - 24 in Nuremberg, Germany, you can find us at the booth of our industry partner **logi.cals** in **hall 6-230**, see details in the section on upcoming events. Further, experts from the CDL-Flex and industry partners will present recent research results at the *Modeling of Embedded Systems Conference*, *Tag des Systems Engineering*, and the *Embedded Software Engineering Kongress*.

In this edition of the newsletter, you will find results from CDL-Flex research and evaluation:

- Lessons learned from Industry Use Cases: **AML.hub – Round Trip Engineering** with heterogeneous engineering models.
- Research Use Case: **Collaborative *AutomationML* Review Support** for efficient and early defect detection in engineering models.
- Inside View: **Model-Driven Engineering for Cyber-Physical Production Systems** to enable the efficient derivation of software tools.
- New Research Results: **Simulation Generation** with *AutomationML* from heterogeneous engineering models.
- Consider taking part in the **upcoming events** with experts from the CDL-Flex.

We hope you enjoy the articles and find food for thought on potential improvements and new solutions in your environment. On request, we will be happy to provide you with the cited papers. We are looking forward to discussing your suggestions on issues for research and development to foster alternative solutions for better software data, model, and tool integration in engineering environments.

Heinrich Steininger	Prof. Dr. Stefan Biffl	Peter Lieber
CTO logi.cals	Head of the Christian Doppler laboratory CDL-Flex at TU Wien	CEO LieberLieber
Heinrich.Steininger@logicals.com	Stefan.Biffl@tuwien.ac.at	Peter.Lieber@LieberLieber.com
www.logicals.com	cdl.ifs.tuwien.ac.at	www. LieberLieber.com

COORDINATING ROUND-TRIP ENGINEERING WITH HETEROGENEOUS ENGINEERING MODELS IN THE AML.HUB

Software and systems engineering projects require the cooperation of several engineering disciplines, such as electrical, mechanical, and software engineering. However, in engineering tool networks the distributed engineering of automated systems often relies on point-to-point data exchange [2], which a) does not sufficiently enable quality and consistency management, b) complicates round-trip engineering, and c) hampers the traceability of changes across engineering disciplines.

The need for round-trip engineering arises when the same information is present and relevant in multiple engineering domains and therefore inconsistencies may occur if not all related system elements are consistently updated to reflect a given change. Engineering views on the plant model are not automatically synchronized and changes between engineering operations in cross-discipline context not made visible to the engineers.

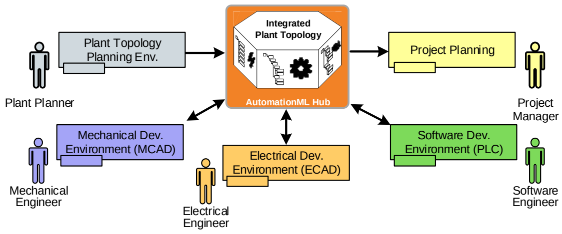

Figure 1: Simple engineering process reflecting the need for Round Trip Engineering

Figure 1 shows a simple engineering process and project role setting. While the plant planner is responsible for defining the overall topology of the automated system, the mechanical engineer, the electrical engineer, and the PLC programmer are in charge of creating and changing detailed engineering data linked to the plant topology.

However, another characteristic of tool networks in multi-disciplinary engineering environments is the vast amount of various data formats and heterogeneous data models. While the emerging *AutomationML* (AML) standard [3] supports structuring engineering data and modeling automation systems, project managers and system integrators may hesitate to migrate all data models of company specific services and tools to AML at once – preferring a step-wise migration of their settings to *AutomationML* to mitigate risks.

The **AutomationML Hub** (AML.hub) concept, as shown in Figure 2, systematically integrates tool networks regardless of the data model of participating engineering tools and enables the automation of engineering processes. While available software tools support individual engineering disciplines quite well, they only represent a discipline-specific view on the engineering plant. Therefore, the *AML.hub* deals in two aspects with engineering information.

On the one hand, the *AML.hub* reflects at its core contributions of all involved disciplines on a so-called *integrated plant model* in a structured manner [1]. That plant model captures and combines all different views into one *AutomationML*-based representation in order to provide an overarching, discipline-independent view on the engineering plant.

On the other hand, the *AML.hub* analyzes the data model of exchanged engineering data and transforms the data into a discipline-specific AML representation in case of *non-AML* models. Once the transformation has been executed, the newly created AML representation is merged into the integrated plant model. This approach provides the following advantages:

- Engineering roles may define and maintain their discipline-specific topology tree of and their tool-specific view on the automated system.
- Engineering projects are AML-ready even if the tools do not export AML.
- The coexistence of engineering tools exporting and importing AutomationML

models and of tools that do not yet facilitate AML is supported.

- A migration strategy from traditional engineering tool networks to AML-based tool networks may be defined.

The **AML.hub** approach facilitates the efficient versioning of exchanged AML models in tool networks and of operations performed on links between various topology trees and views to improve the traceability of changes across disciplines. Versioning also enables deriving the impact of changes on the integrated plant model and reporting differences to the engineer for improvement of their awareness.

The automation of engineering processes facilitates the synchronization of views on the *integrated plant model* and the execution of advanced processes such as test automation for quality assurance.

In an industrial example, the **AML.hub** was evaluated by a hydro power plant builder using various non-AML models for exchanging information about signals across engineering disciplines in their tool network.

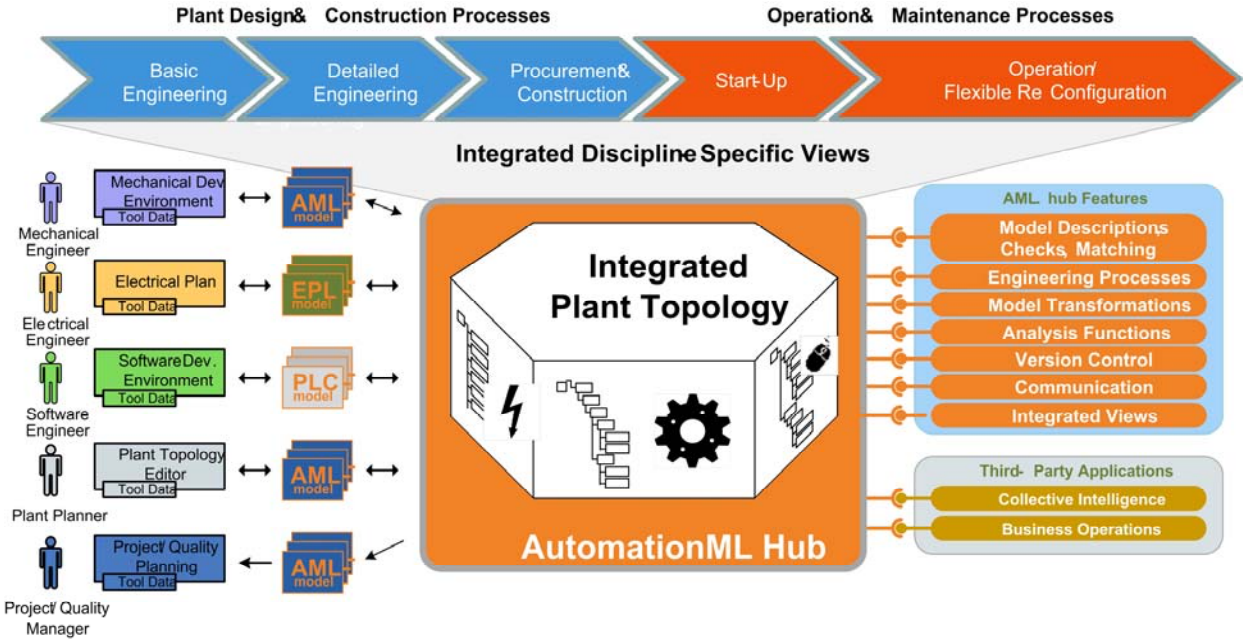

Figure 2: The AutomationML Hub manages engineering-role specific views on an integrated plant model.

References

1. Mordinyi, R.; Biffl, S., "Versioning in Cyber-physical Production System Engineering -- Best-Practice and Research Agenda," in Software Engineering for Smart Cyber-Physical Systems (SEsCPS), 2015 IEEE/ACM 1st International Workshop on , vol., no., pp.44-47, 17-17 May 2015
2. Schmidt N., Lüder A., Biffl S., Steininger H. (2014): "Analyzing requirements on software tools according to functional engineering phase in the Technical Systems Engineering Process", In: Proceedings of the 19th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, pp.1-8.
3. DIN EN 62714 (2015): "Datenaustauschformat für Planungsdaten industrieller Automatisierungssysteme – Automation Markup Language – Teil 1: Architektur und allgemeine Festlegungen", DIN EN 62714-1:2015-06, IEC 62714-1:2014, EN 62714-1:2014.

(Richard Mordinyi)

COLLABORATIVE AUTOMATIONML REVIEW SUPPORT

In **Multi-Disciplinary Engineering** (MDE), stakeholders from different disciplines have to collaborate and exchange data. Heterogeneous engineering models and data typically suffer from risks of defects and inconsistencies, especially if multiple disciplines are involved. The *AutomationML Hub* builds on the standardized data exchange format *AutomationML*[1] that supports the synchronization and data exchange between disciplines. However, additional quality assurance activities are required to identify defects effectively and efficiently. We identified a set of stakeholder needs that include (a) systematic and traceable review processes, (b) effective and efficient defect detection support, and (c) improved engineering artefact quality.

Reviews and inspections are well-established approaches in Software Engineering to identify defects in engineering artefacts early in the engineering process. In real-world contexts, experts typically conduct manual reviews on paper. Because of a high amount of engineering data (e.g., 40k data entities in typical hydro power plant projects), there is a high risk to overlook important defects. Thus, we observed a strong need for process support to drive review processes and tool support to focus on most critical and important deviations. In this context, deviations can be changes (received from different engineers) or defects that need review.

Based on a traditional review process, Figure 3 presents an **Adapted Review Process *(AML Review)*** for *AutomationML* models, applicable for MDE environments, process inputs and outputs, and tool support during for defect detection [1]. The process takes as input the plant topology, specification documents, and review objects, e.g., engineering design documents or engineering changes, derived from the *AutomationML Hub*. Outcome of the planning step is the review package for the review team.

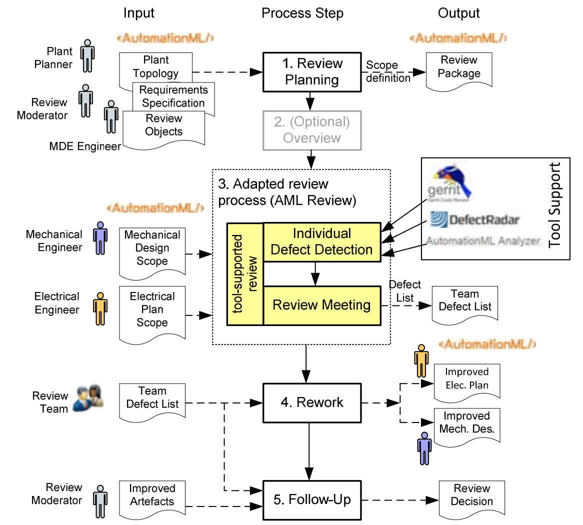

Figure 3: Adapted Review Process (AML-Review) with Tool Support.

Defect Detection with Tool Support. During the defect detection process, several tools can support experts in the review process: *Gerrit Code Review*[2] is an established code review tool in Software Engineering based on changes on code level. *Gerrit* provides a difference view of the review artefacts compared to data already available in common code bases, such as GIT repositories. Reviewers can focus on changes and accept/reject or comment on changes. Note that any (structured) and textual data are applicable, such as *AutomationML* data sets. *DefectRadar*[3], an established tool in building automation for issue reporting, enables the efficient annotation of candidate defects in plans and *pdf* files, representative document types for engineering plans. These annotations can help to identify model entities and support experts in defect detection. The *AutomationML Analyzer*[4] uses integrated *AutomationML* data sets (derived from the *AutomatonML Hub*)

[1] *AutomationML*: https://www.AutomationML.org
[2] Google *Gerrit*: https://www.*Gerrit*codereview.com
[3] DefectRadar: https://www.defectradar.com
[4] *AutomationML Analyzer* Prototype Implementation: http://data.ifs.tuwien.ac.at/aml/analyzer

and queries that enable automated defect detection, browsing through the plant topology, and reporting of reviewing activities [2].

Identified, agreed, and reported defects are the foundation for artefact improvement during the rework process step. Finally, the follow-up step enables the review moderator to decide whether an artefact is mature enough for release.

Based on discussions with industry and research experts the *AML Review* approach provide a set of benefits: (a) Traceable review process; (b) More efficient and effective defect detection processes; (c) *AutomationML* support; and (d) Tool support for important tasks during the defect detection step.

(Dietmar Winkler)

References

1. Winkler D., F.J. Ekaputra, Biffl S.: "AutomationML Review Support in Multi-Disciplinary Engineering Environments", Proc. of ETFA, 2016.
2. Sabou M, Ekaputra FJ, Kovalenko O: “Supporting the Engineering of Cyber-Physical Production Systems with the AutomationML Analyzer”. Proc. of the CPPS Workshop, 2016.

MODEL-DRIVEN ENGINEERING FOR CYBER-PHYSICAL PRODUCTION SYSTEMS

Cyber-Physical Systems (CPSs) are combinations of computational elements/processes with physical elements/processes. CPSs involve the use of embedded devices, networks, and software components with feedback loops, where physical processes affect computations and vice versa. CPSs have been used in a wide range of fields including manufacturing [3]. In this context, *Industrie 4.0* [4] is a collective term embracing a number of contemporary automation, data exchange, and manufacturing technologies. The modern production systems engineering projects, envisioned by *Industrie 4.0*, are large and complex, and often involve heterogeneous engineering disciplines (e.g., mechanical, electrical, and software engineering) and, thus, heterogeneous artefacts (from software programs to hardware platform specifications and simulation models).

In *Model-Driven Engineering* (MDE), the abstraction power of models is applied to tackle the complexity of systems [2]. MDE follows the principle "everything is a model" for driving the adoption and ensuring the coherence of model-driven techniques in the direction of simplicity, generality, and integration. Historically, MDE has been mainly applied in software engineering [2], but in recent years, the application of MDE has been increasing in the industrial automation domain [7].

We build on the foundations of MDE and existing standards for system modeling (*SysML*), data exchange (*AutomationML*), and simulation (*Functional Mockup Interface - FMI*) to provide dedicated support for the engineering process of *Cyber-Physical Production Systems* (CPPSs) realizing the *Industrie 4.0* vision (see Figure 4).

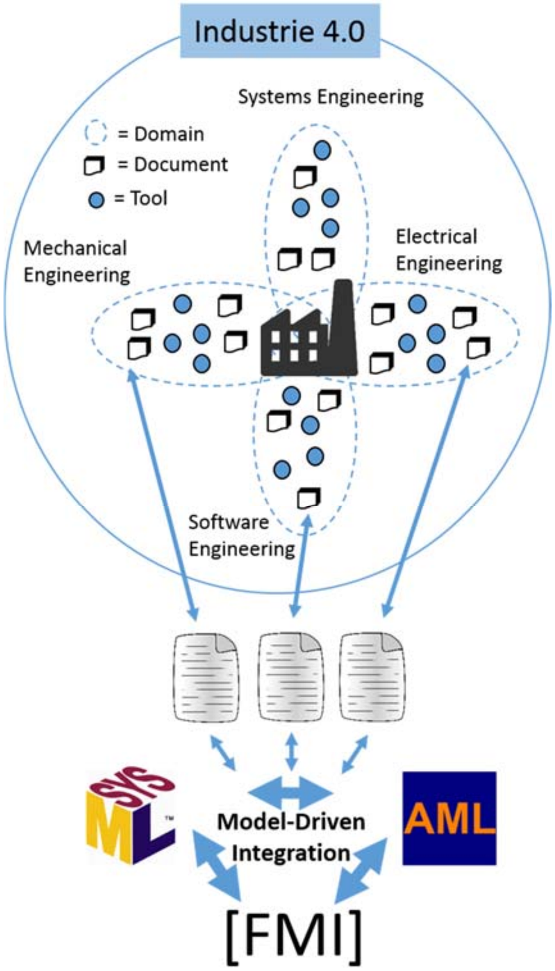

Figure 4. Engineering of CPS in *Industrie 4.0* based on Model-Driven Integration technologies.

AutomationML (AML) [5] is a neutral, free, open, XML-based, and standardized data exchange format for sharing production system (i) structure, (ii) geometry and kinematics, and (iii) logic data. AML has been developed by the *AutomationML* consortium, which consists of companies and academic institutions, including the *Technische Universität Wien*.

SysML [6] is a graphical modeling language standardized by *Object Management Group* (OMG) for the development of large-scale, complex, and multi-disciplinary systems in a model-based approach. *SysML* derives from the *Unified Modeling Language* (UML) and provides modeling concepts for representing the requirements, structure, and behavior of a system in a coherent system model that provides the basis for designing, implementing, and analyzing the system.

Functional Mockup Interface (FMI) [1] is a tool-independent standard to support both model exchange and co-simulation of dynamic models representing physical systems using a combination of XML-files and compiled C-code.

Our research work on MDE for CPPS is currently focusing on:

- The integration of *SysML* and AML models through model transformations and state-of-the-art model-driven technologies – confer also Figure 5;
- Versioning, linking, and co-evolution support for AML models; and
- The integration of discrete and continuous simulation models created with UML/SysML, and FMI standards as well as academic initiatives (e.g., the *Performance Interchange File* format, PMIF [8]). .

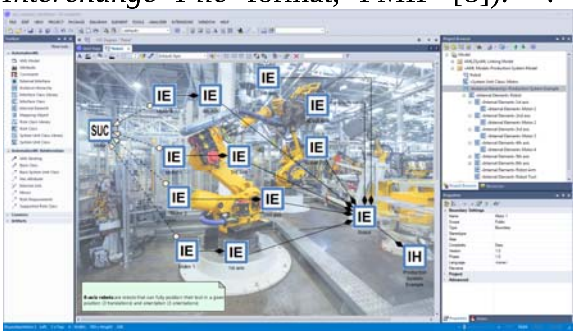

Figure 5. Enterprise Architect based CPPS Modeling Environment.

The integration of considered standards (SysML, AML, FMI) is conducted in cooperation with and with technology support of our industry partner *LieberLieber* (http://www.-lieberlieber.com)

Further information on our research can be found at www.sysml4industry.org.

References

1. T. Blochwitz et al. Functional mockup interface 2.0: The standard for tool independent exchange of simulation models. In Proceedings of the 9th International Modelica Conference, 2012.
2. M. Brambilla, J. Cabot, and M. Wimmer. Model-Driven Software Engineering in Practice. Morgan & Claypool, 2012.
3. EPoSS. Strategic research Agenda of EPoSS - The European Technology Platform on Smart System Integration, 2015.
4. M. Hermann, T. Pentek, and B. Otto. Design Principles for Industrie 4.0 Scenarios: A Literature Review. Working Paper, TU Dortmund, 2015.
5. IEC. IEC 62714:2014 Engineering data exchange format for use in industrial automation systems engineering - Automation markup language. http://www.iec.ch, 2014.
6. OMG. Systems Modeling Language (SysML), July 2015. Version 1.4, http://www.omg.org/spec/SysML/1.4
7. V. Vyatkin. Software Engineering in Industrial Automation: State-of-the-Art Review. IEEE Transactions on Industrial Informatics, 9(3):1234–1249, 2013.
8. L. Berardinelli, E. Maetzler, T. Mayerhofer, M. Wimmer, "Integrating Performance Modeling in Industrial Automation through AutomationML and PMIF". to appear in: INDIN, IEEE, 2016

(Oliver Alt, Luca Berardinelli, Erwan Bousse, Manuel Wimmer)

SIMULATION MODEL DESIGN 4.0

One of the corner stones of the fourth industrial revolution (*Industrie 4.0*) is the *virtualization* of all entities in the manufacturing value chain. The core parts of the virtualization are simulation models that are suitable for simulating real industrial plants as well as their automation and control systems.

Simulation models play an important role for various scenarios in industrial system engineering and run time. Simulation models can be used to train and to test human operators, to verify control systems, or to enable advanced process control.

One of the main issues, which limit the use of simulation models in daily industrial practice, is a *time-consuming and error-prone engineering phase*, which was based on manual work in the past. The main goal of the presented research is to semi-automate the design and re-design phases of simulation models in order to make the development and maintenance of simulations fast and efficient. The basic process steps of the improved simulation model design are depicted in Figure 5.

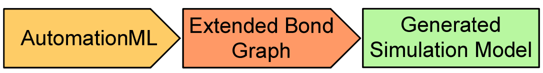

Figure 5: High-level workflow for designing simulation models.

The presented method targets especially large-scale industrial systems as the method assumes that systems consist of *pre-defined atomic components* (such as pumps, pipes, or tanks). We assume that the functionalities of these single components are modeled in a simulation library. The task of the simulation design is thus to select appropriate components from the library, to instantiate them, and to inter-connect them according to the structure of the real system. To facilitate sharing knowledge about the real system structure, the industrial plant model is imported in the *AutomationML* data format. The proposed method cannot accept any arbitrary *AutomationML* file, but rather a *plant model* or a *system model*, whose entities are interpretable from the plant topology viewpoint as well as from the signal viewpoint.

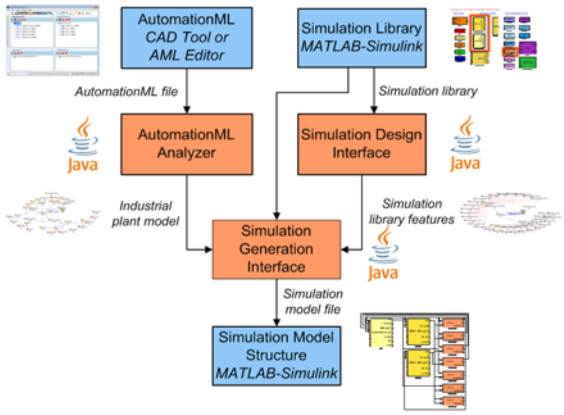

Figure 6: Proposed method for simulation model generation.

The core part of the proposed method, depicted in Figure 6, is the **Simulation Generation Interface**, whose algorithms are based on the well-proven *Bond Graph* theory, which was extended to better reflect the needs of the latest CAE tools and design principles required by *Industrie 4.0* and engineering teams.

The *Bond Graph method* is an engineering approach for describing energy flows in mechatronic systems and for creating simulation models for these systems. *Bond Graphs* are graphical notations of components, connections, and power flows. To illustrate how a *Bond Graph* looks like, we selected a hydraulic two-tank system; Figure 7 depicts the piping and instrumentation diagram, adopted from [2].

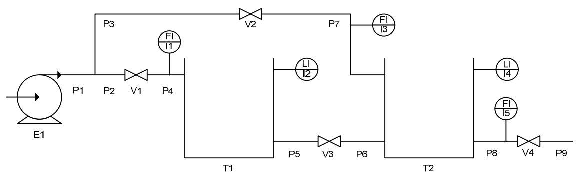

Figure 7: Two-tank hydraulic system.

The standard use of *Bond Graphs* means

to go through the diagram manually and to extract mathematical equations describing the behavior of the system. On the contrary, the *Extended Bond Graph* method focuses on finding a combination of available components in such a way that the topology of the system is properly modeled. The creation of a simulation model can be considered as a combinatory task of finding appropriate combinations of simulation components in such a way that input and output interfaces of the connected components are compatible.

Figure 8 depicts a *Bond Graph* resulting as a solution of the combinatory task. The graph includes the following system components: tanks (T_1, T_2), valves ($V_1 – V_4$), pipes ($P_1 – P_9$), and a pump (E_1). These components are interconnected via *1*-junctions and *0*-junctions that are abstractions of serial and parallel connections from the physical world. Furthermore, *Bond Graphs* include directed power flows, denoted by semi-arrows. Important issues of *Bond Graphs* are causality assignments that support determining which of the variables *effort* and *flow* is the input variable and which is the output variable for each bond.

Bond Graphs can support the following engineering tasks:

1. Automated or semi-automated design of simulation models using simulation component libraries;
2. Design of signal interfaces for the definition of co-simulation units;
3. Support for the specification of a simulation library structure;
4. Slicing complex simulation models into a set of simulation modules and their integration into a co-simulation.

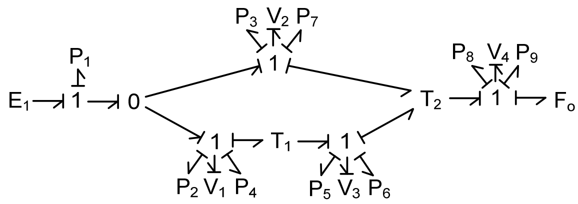

Figure 8: *Bond Graph* of the hydraulic system as an intermediate outcome [2].

The main innovation in the presented approach is the support for the *AutomationML* data format as the input plant model for the generation of simulation models in signal-oriented simulators. Compared to the previous version, a wider set of types of simulation blocks has become supported and an improved parameter management has been included. Supporting various types of simulations and seamless support for commonly used industrial simulations are the issues under current development.

References

[1] Gawthrop, P., Bevan, G.: Bond-Graph Modeling. IEEE Control Systems Magazine, vol. 27, no. 2, p. 24 – 45, April 2007.

[2] Novak, P., Kadera, P., Jirkovsky, V., Vrba, P., Biffl, S.: Engineering of Coupled Simulation Models for Mechatronic Systems. Borangiu, T., Thomas, A., Trentesaux, D.: Service Orientation in Holonic and Multi-Agent Manufacturing - Studies in Computational Intelligence 594. Part I, pp. 3 - 11. Springer-Verlag Berlin Heidelberg, 2015.

(Petr Novak)

UPCOMING EVENTS & ADDITIONAL INFORMATION

Modeling of Embedded Systems Conference (MESCONF), 6.10.2016, Infineon-Campus in München, http://mesconf.de/

4TH AUTOMATIONML USER CONFERENCE

- October 18-19, 2016
- Esslingen, Germany
- https://www.automationml.org/o.red.c/conference2016.html

Experts from the *CDL-Flex* will be present at the 4th *AutomationML User Conference*, scheduled for October 18-19 at Festo AG in Esslingen, Germany. Researchers from the *CDL-Flex* will present and discuss latest developments in context of efficient data exchange with the *AutomationML Hub* and collaborative reviews for improving engineering artifacts and projects in multi-disciplinary engineering environments.

Tag des Systems Engineering (TdSE) 25. – 27.10.2016, Schaeffler AG, Herzogenaurach, http://www.tdse.org/

SPS/IPC/DRIVES 2016 TOOLS FAIR

- November 22-24, 2016
- Nuremberg, Germany
- www.mesago.de/de/SPS/home.htm

Experts from *CDL-Flex* will be present also this year on the *SPS/IPC/Drives* from November 22 to 24 in Nuremberg, Germany. At the booth of the CDL-Flex industry partner **logi.cals (hall 6-230)**, you will get insight into the latest results from applied research and development results, which have been worked out in cooperation with industry partners. Come with a *Free Ticket* to the *SPS/IPC/Drives* and visit us for a Viennese *Apfelstrudel*! For more information, please contact Dietmar Winkler at dietmar.winkler@tuwien.ac.at.

Embedded Software Engineering Kongress (ESE) - 28.11 - 2.12.2016, Sindelfingen, http://www.ese-kongress.de/

Software Quality Days 2017 (SWQD)

- January 17 - 20, 2017
- Vienna, Austria
- http://software-quality-days.com/

Experts from *CDL-Flex* have been organizing the *scientific program* of one of the largest events on software quality and process improvement, the Software Quality Days in Vienna. "*Quality of Things – Complexity and Challenges of Software Engineering in Emerging Technologies*" is the motto of the upcoming event with more than 60 presentations, 30 industry demonstrations, and over 300 participants.

READER'S PICKS

Biffl S., Sabou M. "*Semantic Web Technologies for Intelligent Engineering Applications*", Springer, 2016.

Mordinyi R., Winkler D., Ekaputra F., Wimmer M., Biffl S.: Investigating Model Slicing Capabilities on Integrated Plant Models with AutomationML. in Proceedings of the 21th IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA), 2016.

Winkler D., Mordinyi R., Biffl S.: 'Qualitätssicherung in heterogenen und verteilten Entwicklungsumgebungen für industrielle Produktionssysteme', In: Vogel-Heuser B., Bauernhansl T., ten Hompel M. (Eds.): "*Handbuch Industrie 4.0*", 2. Auflage, Springer, 2016.

Winkler D., Ekaputra F.J., Biffl S.: 'AutomationML Review Support in Multi-Disciplinary Engineering Envi-ronments', In: Proc. of the 21st IEEE Int. Conf. on *Emerging Technologies and Factory Automation* (ETFA), IEEE, 2016

Biffl S., Lüder A., Gerhard D. "*Multi-Disciplinary Engineering for Cyber-Physical Production Systems*", Springer, to appear in March 2017.

CDL-FLEX RESULTS ONLINE

Do not miss the latest **presentations, use cases, videos, and screen casts** of implemented prototypes on the *CDL-Flex* Website at http://cdl.ifs.tuwien.ac.at