Comprehensive Integration of Executable Business Process Models with Semantic Concept and Task Specifications

Ralph Hoch, Hermann Kaindl, Roman Popp and Dominik Ertl
Institute of Computer Technology, Vienna University of Technology
Vienna, Austria
{hoch, kaindl, popp, ertl}@ict.tuwien.ac.at

***Abstract*—Business process models mostly exist without a corresponding reference ontology of the business and its domain, or without semantic specifications of their basic tasks to be executed through software. This work shows our 'big picture' of integrated support for modeling business processes in BPMN 2.0 (Business Process Model and Notation). It illustrates representing semantic information of such business processes and its links. So, this paper presents our comprehensive integration of executable business process models represented in BPMN 2.0 with semantic concept and service task specifications. Our integration can be used for model verification and generation, as well as model execution with assigned (software) services (or objects).**

Keywords-Business process modeling; semantics of business process models; service tasks

I. Introduction

Executable business process models should facilitate both the business itself and the development of software supporting it, which was a major motivation for the development of BPMN (Business Process Model and Notation) 2.0 [1] for representing the business process models. It has a standardized specification of business processes with the possibility to directly execute them, based on the defined BPMN 2.0 meta-model and assigned (Web) services, although we found a pitfall in this regard [2]. Instead of hand-crafting both the process models and the related software, we have a vision that integrated semantic concept and task specifications may improve on that by their possible use for automated generation and formal verification [3] as well as general support for related software development in various usage scenarios.

The remainder of this paper is organized in the following manner. First, we discuss related work on previous approaches to partial integration. Then we present our 'big picture' of integrated support for modeling business processes in BPMN 2.0. Based on that, we sketch potential use of this integration. Finally, we draw tentative conclusions and propose future work.

II. Related Work

Earlier work on the interplay of business process models, service/task ontologies, and domain ontologies was carried out in the project SUPER (Semantics Utilised for Process Management within and between Enterprises) [4], where a tool named "Maestro for BPMN" was developed. Born et al. [5] describe how user-friendly semantic annotation of process artifacts with tags/markups can be achieved in business process models via Maestro. These annotations refer to semantics in domain ontologies, and based on them, this tool allows one to automatically compose activities within business processes [6]. Maestro also supports certain consistency checks of the control flow against semantic annotations of such annotated processes [6]. Born et al. [7] describe how "adequate" services can be identified for specific tasks through match-making by use of the semantic annotations. Maestro was based on a previous BPMN version, which is not suitable for direct execution of business processes. This is different to our work, which specifically focuses on *executable* business process models in BPMN 2.0. Our work also envisages a more comprehensive integration of such models with semantic concept and task specifications, and especially on even more ambitious scenarios of use of the integration also with executable models. In addition, we also focus on validation and formal verification of business processes [3].

Burkhart et al. [8] define in more recent work a *structural description of business models*. Their synthesis of eight existing ontologies is extended with state-of-the-art understanding and research progress on business models. This work proposes transformation of such structural descriptions to business process models, which is a different but possibly complementary approach to ours.

The automatic execution of business process has been studied for quite some time, primarily with *Business Process Execution Language for Web Services (BPEL4WS)*. Aslam et al. [9] describe how BPEL4WS can be enhanced with semantic information via the *Web Ontology Language for Web Services (OWL-S)*. They present a mapping strategy as well as tool support for that. BPEL4WS was primarily designed for orchestration of Web services in the sense of automatic execution of business process models using these services. Our work builds on the more recent BPMN 2.0 for this purpose, since it additionally facilitates a graphical representation of business process models, which is better understandable also by business process managers. In this way, BPMN 2.0 intends to narrow the gap between the business and the software worlds.

Marzullo et al. [10] proposed another integration effort, with

 ISBN: 978-1-61208-395-7

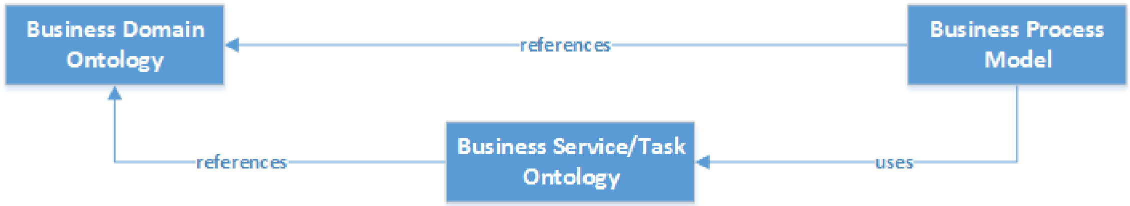

Figure 1: Conceptual View of Integration

the purpose of supporting domain-driven software development. So, it centers around a shared domain specification to be used as a reference point for software applications. The central domain repository allows exchanging information in a standardized way between different projects or companies. So, the focus is clearly on efficient software development, even though Marzullo et al. [11] describe possibilities to include business process modeling as well. In contrast to our approach, the domain specification is not based on a formal specification language or ontology. Saiyd et al. [12] describe a similar approach to Marzullo et al., but propose an ontological foundation for domain-driven design. However, their work is more focused on the specification of the ontological concepts than their actual use. So, neither of these approaches has such a comprehensive integration and the scenarios of its use in mind that we propose in our work.

While BPMN 2.0 has, in contrast to the previous version BPMN 1.0, a defined meta-model, it is not based on a logic foundation. Therefore, Natschlaeger et al. [13] propose an OWL-based (Web Ontology Language) upper ontology for BPMN 2.0 to allow a formalized specification of BPMN 2.0 processes. Using it in our integration approach would certainly be possible and interesting, since it would make it completely based on ontologies. As it stands, however, our early feasibility prototypes indicate that using the meta-model of BPMN 2.0 should be sufficient for our currently envisaged scenarios of use.

Cabral et al. [14] show in their work how business process modeling can benefit from semantic information. They describe the ontology BPMO (Business Process Modeling Ontology), which includes semantic knowledge about organizational context, workflow activities and other business process parts. Using this ontology, it is possible to refer to semantically annotated data and services for working in a coherent way. In contrast, our approach uses BPMN 2.0 as modeling and orchestration language. In addition, we focus on combining BPMN with OWL semantics rather than representing business processes in an ontology.

Semantic Business Process Management (SBPM) helps handling the life cycle of business process management through ontologies and Web services, as proposed by Filipowska et al. [15]. They illustrate with various scenarios how SBPM can be used in the business process management area. Hepp et al. [16] describe a set of ontologies for SBPM, which target the spheres of enterprise structures and operations. This work is based on *Web Service Modeling Ontology (WSMO)* [17] and its closely related representation language *Web Service Modeling Language (WSML)* [18] for combining semantic Web services with business process management [19]. In contrast, we use OWL-S for semantic specification of services, and BPMN 2.0 for execution of business processes, but we do not strive for representing business processes in an ontology.

III. Big Picture of Integration

We propose here a comprehensive integration of business processes with semantic concept and service task specifications. Figure 1 shows the conceptual view of our big picture of this integration. The *Business Domain Ontology* represents the concepts/objects of the business domain, and the *Business Service/Task Ontology* represents the services/tasks. Services operate on objects and manipulate their states. They may even be coupled with a specific object and define the operations that are possible or allowed on this object. In an artifact-centric approach, they enable the life-cycle defined for the object. The *Business Process Model* specifies how services/tasks are composed and on which objects they operate on. Taken together, this proposed integration specifies the essence of the business.

Several technologies exist for realizing each of the parts shown in Figure 1. Figure 2 shows the big picture of integration with concrete technologies chosen by us for this realization. In our approach, we chose technologies based upon their widespread availability and use, and how well they support our integration. Semantic repositories are often specified via ontologies represented in Web Ontology Language (OWL) [20], a knowledge representation language. We decided to use OWL as well, since it is widely used for representing ontologies and has a wide range of applications. The semantic specification of services is provided through an OWL-S (an ontology built upon OWL for semantic descriptions of Web services) repository, since it allows direct reference to existing OWL ontologies [21]. Furthermore, it allows additional specifications (pre- and post-conditions) for services, which expand on the typical input-/output-specification of services per se as given in WSDL (Web Service Description Language) descriptions [22].

Especially for the *Business Process Model*, several languages and technologies could be used as many companies even use their own approach to specify business processes. We chose BPMN 2.0 as it is the current version of a standardized and open specification language for business processes [1]. It also allows the automatic execution of business processes when software is available for so-called service tasks (in the form of Web services or Java objects). BPMN 2.0 is currently

 ISBN: 978-1-61208-395-7

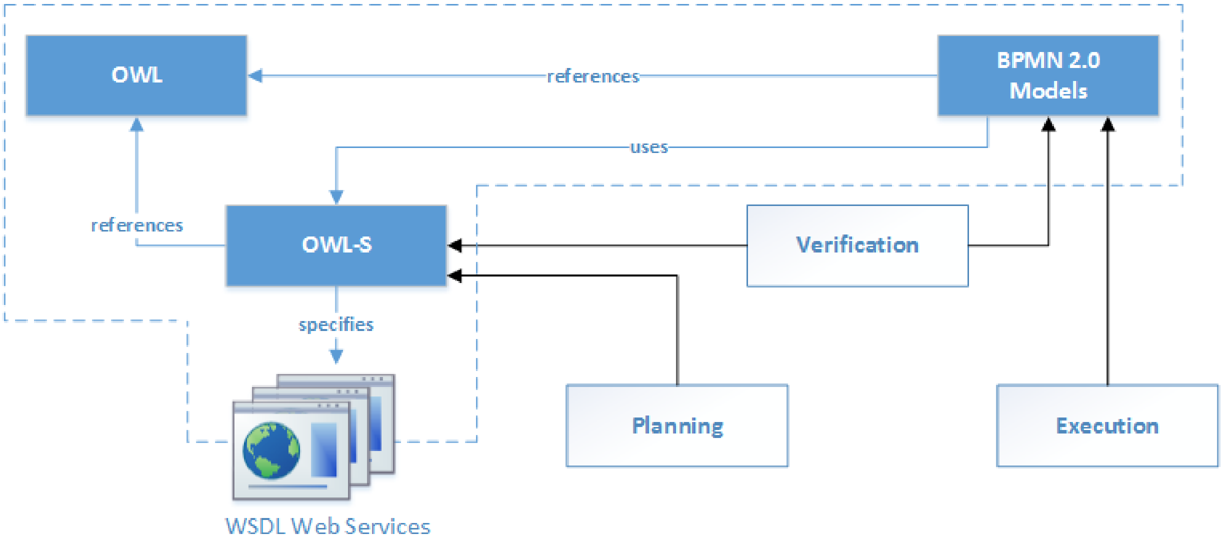

Figure 2: Big Picture of Integration and Use with Concrete Technologies

supported by a wide range of execution frameworks for that purpose.

However, BPMN 2.0 does not specify any direct relationship to OWL-S services, and only WSDL services are described in the standard. Therefore, a wrapper is needed that allows the transformation of OWL references to existing services or objects in the overall software framework. The wrapper has to be built around an existing BPMN 2.0 execution framework. Figure 3 shows how the wrapper operates and how it is integrated into the framework. Technically, the wrapper sits on top of an BPMN 2.0 execution engine and deals specifically with OWL references. To accomplish this, the OWL references have to be translated to their corresponding implementations, i.e., to Web Services specified through WSDL. The resulting address of the corresponding WSDL is then used to directly call its Web Service, and by doing so, executing the BPMN 2.0 process. In our approach, we focus on Web Services described by WSDL, but other approaches such as a mapping to implemented methods or objects are also possible.

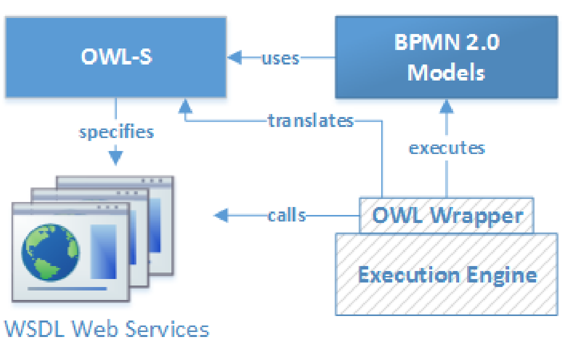

Figure 3: Wrapper for Execution Engine

IV. Potential Use of the Integration

This comprehensive integration can (potentially) be used for the following approaches, once it is implemented in an integrated tool environment:

- domain-driven development of business software with (fragments of) a business process model as its target;
- top-down development based on business process models linked to ontologies;
- automated generation of (certain kinds of) business process models through plan generation;
- formal verification of (certain kinds of) business process models through a logic-based approach [3];
- automated generation of (parts of) user interfaces for business software based on a domain ontology, and
- execution of business process models as business software [2].

While most of these approaches can be done in isolation, our proposed integration offers certain added value at least for some of them as indicated below.

Development of business software based on domain artifacts can, of course, be done based on any kind of domain model. However, ontologies may serve this approach better due to their formally defined concepts and services. In addition to bottom-up development, we envisage a certain top-down orientation by having at least fragments of a business process model as a target. We also envisage that the business software developed in this way would not just implicitly encode a business process, but also an explicit business process model should result. In an integrated environment, it would be linked with the ontologies. Possibly, the whole software could be centered around the business process model in such a way, that an engine interpreting it is integrated and drives it.

Starting development top-down from a business process model is also possible, also without our proposed integration.

 ISBN: 978-1-61208-395-7

It involves adding software parts (possibly in the form of Web services) to tasks in the model. In the proposed integrated environment, this should be facilitated especially through the semantic information in the ontologies, which can, e.g., help finding relevant artifacts through semantic matching.

Based upon the OWL-S specification of services, approaches for automatic *planning* algorithms have already been proposed, which try to automatically generate composite services out of atomic services (see, e.g., Klusch et al. [23] and Ziaka et al. [24]). Figure 2 indicates this through the `Planning` box. The line with the arrow pointing towards the `OWL-S` box indicates that OWL-S is the conceptual basis for planning. Logic-based technologies, such as the Fluent Calculus [25] and its supporting tool FLUX [26], seem to be even preferable for this purpose because of their well-defined semantics. Still, a transformation from OWL-S to FLUX is necessary, such as the one given in [27]. Note, however, that only certain kinds of business process models can be generated automatically by these planning algorithms. Overall, this planning approach using FLUX can handle everything that FLUX can. Our proposed integration would make the generated business processes directly available in BPMN 2.0.

When using FLUX, a formal verification of a business process model against the specification of the services composed in it can be performed automatically [3]. Figure 2 indicates this through the `Verification` box. Analogously to generation of process models, only certain kinds of business process models can be automatically verified using this approach. Our proposed integration would directly take them from a BPMN 2.0 representation as linked to OWL-S.

Integrating these technologies can also lead to a more flexible software application with an adaptable user interface based on the ontologies. Since the domain ontology already specifies what kind of attributes are related to a concept, a simple user interface can be generated based on this specification. Changes in the domain ontology would be directly reflected in the resulting user interface. In combination with additional descriptions of the user tasks, an individual user interface for each process can be automatically generated.

Execution of process models represented in BPMN 2.0 is, in principle, possible with properly attached Web services or Java objects (while there are certain intricacies, see [2]). Our proposed integration still allows it (as indicated through the `Execution` box in Figure 2), although this integration actually makes it more difficult. In fact, BPMN 2.0 normally links to WSDL directly, so that our integration needs the additional wrapper sketched above, around the execution engine for BPMN 2.0 models.

The real added value of our proposed integration, however, comes from the combined use of (some of) these approaches. Let us sketch a few envisaged scenarios of such combined use.

Scenario 1:
Create a business process model in BPMN 2.0 using the available and specified building blocks in the ontologies and corresponding service implementation.
Verify it against the semantic specifications of its parts.
Execute it with the service implementations.

The added value of this scenario is convenient modeling and verification in an integrated tool environment, where potential errors through model translations can be avoided.

Scenario 2:
Automatically generate a business process model for a given goal.
Execute it with the service implementations.

The added value of this scenario is similar to the one of Scenario 1, but the process is even automatically generated, and verified by its very construction.

Scenario 3:
Create a business process model from scratch or take a given one.
Create semantic specifications of services not yet available.
Verify the model against the semantic specifications of its services.
Implement missing services according to their specifications.
Execute the business process model with the service implementations, both old and new.

The added value of this scenario is that our integration enables software developers and business process experts to work together in a consistent business domain without potential mismatches between their respective domains. In addition, the semantic specifications of Web services can be verified against their composition or a goal specification, even before they will be implemented.

V. Conclusion And Future Work

This is work in progress and presents a vision. Since it is primarily based yet on previous work, a conceptual view supported by a few feasibility prototypes and work on verification as well as execution in this context, only tentative conclusions can be drawn. Still, we have already sketched new scenarios of potential use of our comprehensive integration for developing business software in ways not yet envisioned in previous work (to our best knowledge). Viewed from another perspective, this approach should lead to new ways of software reuse and reusability.

Future work will, of course, have to face the challenges involved in creating (or reusing) corresponding ontologies, in building and testing several parts of this comprehensive integration, and finally to evaluate.

VI. Acknowledgment

Part of this research has been carried out in the ProREUSE project (No. 834167), funded by the Austrian FFG.

References

[1] Business Process Model and Notation (BPMN), http://www.omg.org/spec/BPMN/2.0, The Object Managment Group (OMG) Std., Jan. 2011, [Online; accessed 08-February-2015]. [Online]. Available: http://www.omg.org/spec/BPMN/2.0

[2] C. Gutschier, R. Hoch, H. Kaindl, and R. Popp, "A Pitfall with BPMN Execution," in WEB 2014, The Second International Conference on Building and Exploring Web Based Environments, 2014, pp. 7–13.

[3] R. Hoch, H. Kaindl, R. Popp, D. Ertl, and H. Horacek, “Semantic Service Specification for V&V of Service Composition and Business Processes,” in Proceedings of the 48nd Annual Hawaii International Conference on System Sciences (HICSS-48). Piscataway, NJ, USA: IEEE Computer Society Press, 2015.

[4] SUPER, “Semantics Utilised for Process Management within and between Enterprises,” http://www.ip-super.org/, 2009, [Online; accessed 21-January-2015].

[5] M. Born, F. Doerr, and I. Weber, “User-friendly semantic annotation in business process modeling,” in Web Information Systems Engineering - WISE 2007 Workshops, ser. Lecture Notes in Computer Science, M. Weske, M.-S. Hacid, and C. Godart, Eds. Springer Berlin Heidelberg, 2007, vol. 4832, pp. 260–271. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-77010-7_25

[6] M. Born, J. Hoffmann, T. Kaczmarek, M. Kowalkiewicz, I. Markovic, J. Scicluna, I. Weber, and X. Zhou, “Supporting execution-level business process modeling with semantic technologies,” in Database Systems for Advanced Applications, ser. Lecture Notes in Computer Science, X. Zhou, H. Yokota, K. Deng, and Q. Liu, Eds. Springer Berlin Heidelberg, 2009, vol. 5463, pp. 759–763. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-00887-0_67

[7] M. Born, J. Hoffmann, T. Kaczmarek, M. Kowalkiewicz, I. Markovic, J. Scicluna, I. Weber, and X. Zhou, “Semantic annotation and composition of business processes with maestro,” in The Semantic Web: Research and Applications, ser. Lecture Notes in Computer Science, S. Bechhofer, M. Hauswirth, J. Hoffmann, and M. Koubarakis, Eds. Springer Berlin Heidelberg, 2008, vol. 5021, pp. 772–776. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-68234-9_56

[8] T. Burkhart, S. Wolter, M. Schief, J. Krumeich, C. D. Valentin, D. Werth, P. Loos, and D. Vanderhaeghen, “A comprehensive approach towards the structural description of business models,” in International Conference on Management of Emergent Digital EcoSystems, MEDES ’12, Addis Ababa, Ethiopia, October 28-31, 2012, J. Kacprzyk, D. Laurent, and R. Chbeir, Eds. ACM, 2012, pp. 88–102. [Online]. Available: http://doi.acm.org/10.1145/2457276.2457294

[9] M. Aslam, S. Auer, J. Shen, and M. Herrmann, “Expressing business process models as owl-s ontologies,” in Business Process Management Workshops, ser. Lecture Notes in Computer Science, J. Eder and S. Dustdar, Eds. Springer Berlin Heidelberg, 2006, vol. 4103, pp. 400–415. [Online]. Available: http://dx.doi.org/10.1007/11837862_38

[10] F. Marzullo, J. de Souza, and J. Blaschek, “A domain-driven development approach for enterprise applications, using MDA, SOA and Web services,” in E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services, 2008 10th IEEE Conference on, July 2008, pp. 432–437.

[11] F. Marzullo, J. Moreira de Souza, and G. Xexeo, “A domain-driven approach for enterprise development, using BPM, MDA, SOA and Web services,” in Innovations in Information Technology, 2008. IIT 2008. International Conference on, Dec 2008, pp. 150–154.

[12] N. Al Saiyd, I. Al Said, and A. Al Neaimi, “Towards an ontological concepts for domain-driven software design,” in Networked Digital Technologies, 2009. NDT ’09. First International Conference on, July 2009, pp. 127–131.

[13] C. Natschlaeger, “Towards a BPMN 2.0 ontology,” in Business Process Model and Notation, ser. Lecture Notes in Business Information Processing, R. Dijkman, J. Hofstetter, and J. Koehler, Eds. Springer Berlin Heidelberg, 2011, vol. 95, pp. 1–15. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-25160-3_1

[14] L. Cabral, B. Norton, and J. Domingue, “The business process modelling ontology,” in Proceedings of the 4th International Workshop on Semantic Business Process Management, ser. SBPM ’09. New York, NY, USA: ACM, 2009, pp. 9–16. [Online]. Available: http://doi.acm.org/10.1145/1944968.1944971

[15] A. Filipowska, M. Kaczmarek, M. Kowalkiewicz, I. Markovic, and X. Zhou, “Organizational ontologies to support semantic business process management,” in Proceedings of the 4th International Workshop on Semantic Business Process Management, ser. SBPM ’09. New York, NY, USA: ACM, 2009, pp. 35–42. [Online]. Available: http://doi.acm.org/10.1145/1944968.1944975

[16] A. Filipowska, M. Hepp, M. Kaczmarek, and I. Markovic, “Organisational ontology framework for semantic business process management,” in Business Information Systems, ser. Lecture Notes in Business Information Processing, W. Abramowicz, Ed. Springer Berlin Heidelberg, 2009, vol. 21, pp. 1–12. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01190-0_1

[17] W3C, “Web Service Modeling Ontology (WSMO) Primer,” W3C, June 2005, [Online; accessed 08-February-2015]. [Online]. Available: http://www.w3.org/Submission/WSMO-primer/

[18] W3C, “Web Service Modeling Language (WSML),” W3C, June 2005, [Online; accessed 08-January-2015]. [Online]. Available: http://www.w3.org/Submission/WSML/

[19] M. Hepp, F. Leymann, J. Domingue, A. Wahler, and D. Fensel, “Semantic business process management: a vision towards using semantic web services for business process management,” in e-Business Engineering, 2005. ICEBE 2005. IEEE International Conference on, Oct 2005, pp. 535–540.

[20] W3C, “OWL 2 Web Ontology Language Second Edition,” 12 2012, [Online; accessed 08-February-2015]. [Online]. Available: http://www.w3.org/TR/owl2-overview/

[21] “OWL-based web service ontology version 1.2,” http://www.daml.org/services/owl-s/1.2/, Web-Ontology (WebOnt) Working Group, March 2004, [Online; accessed 08-February-2015]. [Online]. Available: http://www.daml.org/services/owl-s/1.2/

[22] R. Chinnici, J.-J. Moreau, A. Ryman, and S. Weerawarana, “Web services description language (WSDL) 2.0,” http://www.w3.org/TR/wsdl20/, w3c, 6 2007, [Online; accessed 08-February-2015]. [Online]. Available: http://www.w3.org/TR/wsdl20/

[23] M. Klusch and A. Gerber, “Evaluation of Service Composition Planning with OWLS-XPlan,” in Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, ser. WI-IATW ’06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 117–120. [Online]. Available: http://dx.doi.org/10.1109/WI-IATW.2006.68

[24] E. Ziaka, D. Vrakas, and N. Bassiliades, “Web Service Composition Plans in OWL-S,” in Agents and Artificial Intelligence, ser. Communications in Computer and Information Science, J. Filipe and A. Fred, Eds. Springer Berlin Heidelberg, 2013, vol. 271, pp. 240–254. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-29966-7_16

[25] M. Thielscher, “Introduction to the Fluent Calculus,” Electron. Trans. Artif. Intell., vol. 2, 1998, pp. 179–192.

[26] C. Drescher, “Flux agent,” http://www.fluxagent.org/, [Online; accessed 21-January-2015].

[27] E. Chifu, I. Salomie, and E. St.Chifu, “Fluent calculus-based web service composition - from OWL-S to fluent calculus,” in Intelligent Computer Communication and Processing, 2008. ICCP 2008. 4th International Conference on, 2008, pp. 161–168.