

Context-Specific Multi-Model-Template Retrieval

Frank Hilbert, Raimar Scherer

► **To cite this version:**

Frank Hilbert, Raimar Scherer. Context-Specific Multi-Model-Template Retrieval. Luis M. Camarinha-Matos; Lai Xu; Hamideh Afsarmanesh. 13th Working Conference on Virtual Enterprises (PROVE), Oct 2012, Bournemouth, United Kingdom. Springer, IFIP Advances in Information and Communication Technology, AICT-380, pp.234-241, 2012, Collaborative Networks in the Internet of Services. <10.1007/978-3-642-32775-9_24>. <hal-01520414>

HAL Id: hal-01520414

<https://hal.inria.fr/hal-01520414>

Submitted on 10 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Context-specific Multi-model-template Retrieval

Frank Hilbert and Raimar J. Scherer

Institute of Construction Informatics, Dresden University of Technology,
Dresden, Germany
{Frank.Hilbert, Raimar.Scherer}@tu-dresden.de

Abstract. Although the construction industry, in contrast to the stationary industry, is characterized by One of a Kind fabrication, there can be found similarities in project, product, process and organizational structures. The focus of this paper is to identify relationships, roles and model types, which are often needed in a specific project context and summarize these similarities by the use of a collaboration ontology providing for Multi-Model logistics. Reference models on the one side and a model to describe the project context on the other side should ensure that all partners retrieve precisely situation-specific selections of the application models which are necessary for their tasks. Furthermore, a pilot scenario is presented, which evaluate this approach on an Azure Cloud collaboration platform. As a result, we obtain an approach, which allows simplified handling and reuse of complex project-, product- and collaboration-models in order to support the collaboration within a virtual Organization in Construction Industry.

Keywords: Context model, Multi-Model, Project Collaboration Ontology

1 Introduction

In History of the construction industry, always separate independent organizations of different domains and disciplines joined together in various short-term forms of organisation to combine their core competencies for handling of large and complex construction projects [1]. To allow a collaborative partnership with cooperative aspects, it is necessary to set up transparent rules for cooperation with clearly defined competencies, rights and duties of the participants, therefore various project-specific organizational, process and product models are necessary. Organizational models have evolved from tayloristic, hierarchical organizational structures towards to new forms of organization, like Virtual Organization, Collaborative Networked Organization or VO Breeding Environments as can found in [2]. For the definition of organizational roles in the building industry, there are several classifications developed, such as IFC-Actor roles [3] and OmniClass [4]. For the modelling of processes it is essential if there are predominantly material or information-transforming processes. For both process model types a variety of methods, languages and notations have been developed, here a detailed summary is given in [5]. In our Work we

focus on information transforming processes. To facilitate the representation and exchange of product information, data product models are an efficient way and in the course of time, they have evolved from the pure 2D drawings with small reuse value to rich building information models (a detailed summary is given in [6]). The current trend is concerned with modeling of implicit dependencies between different and inhomogeneous elementary models in the form of so-called Multi-Models (see chapter 2.2).

Although AEC is typically characterized by an One of a Kind fabrication, there can be found similarities in organizational, process and product structures, because a construction process is a repeated but mostly not identical course of action, while a construction product is a unique composition of common construction processes [5]. Similarities can be found by comparison of model metadata. Therefore the focus of this paper is the use of such implicit similarities to support project collaboration. With the help of ontologies dynamic entity centric context models [7] (actor, process and product context) should be generated, which summarize the information obtained by crosslinking the metadata. It is expected that with the help of such a context model significantly more context-specific collaboration possibilities can be determined.

In chapter 2 the developed project collaboration ontology is described and the generation of the context model is presented in Chapter 3. The last paragraph summarizes the results and gives a short outlook.

2 Project Collaboration Ontology

The Fundamental idea of the Project Collaboration Ontology is a generic formal description of organisation, process and product information to enable the utilization in a framework of integrated processes. Thereby Collaboration includes all the process and product related activities between entities (actors, processes and products), whose common goal is the creation of a product (or service) [9]. Basic elements of collaboration are coordination; communication and cooperation (see also [10]). For the representation of the entities we orient on the Semantic Web approach [11] and expand all entities with descriptive, machine-readable metadata, organized in ontologies, which are defined in [12] as “a formal, explicit specification of a shared conceptualization”. This allows automated interpretation, evaluation and processing of information and enables us to extract the implicit knowledge, which can represent similarities. Thereby Ontologies stores concepts as well instances [13]. The concepts consist of classes of descriptive properties with cardinality, transitivity and symmetry qualities. Each property has a definition range and a set of values, distinguishing here between data type properties, representing simple Data types and class properties, which map classes to other and can be organized in hierarchical structures. Ontology containers can be evaluated through description logic algorithms used by Reasoner, for example RACER [14] or JESS [15]. Another important advantage is the possibility to integrate other ontologies. Figure 1 illustrates the structure of the used Project Collaboration Ontology integrating various ontologies necessary for the con-

text determination, as as described below. If several Organizations joining collaboration the same underlying concepts and terminology has to be used.

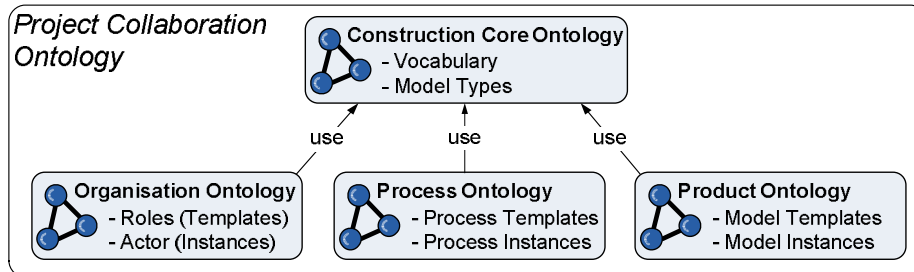


Fig. 1. Project Collaboration Ontology (PCO)

The Purpose of the *Construction Core Ontology* on the top is to define concepts and relationships between them, used in the construction domain. More information about this core ontology can be found in [17].

(1) The scope of the *Organisation Ontology* is describing the actors, rules of their collaboration as well as roles and permissions of a virtual Organization in an organizational model. In our approach, we use an organisational model, which closely follows the IFC standard [3] mapping the competences of involved partners to roles, which consist of permissions on product model types (e.g. multi-model types, defined in the construction core ontology). Such permissions may also be required for the execution of processes. The distinction between potential roles (competences) and current user roles, as described in [18] will be neglected here.

(2) Reusable Processes can be depicted as Configurable Reference Process Models (CRPM) as described in [5]. In our scenario we use Business Process Modelling Notation (BPMN) [19] for the representation of common reference processes, which specify used resources, produced results and necessary permissions. Such Reference Processes are stored in the *Process Ontology* and can have a large number of attributes. Important for our scenario is the description of in- and output models as well as necessary actor-permissions.

(3) The basic idea of the *Product Ontology* is the description of product model instances and templates for a better assignment of actors and processes. Essential with the use of product models are the knowledge of their dependencies. For this requirement in the German research project Mefisto [20] the multi-model approach was developed, which provides the externalization of implicit relationship of semantically and structurally inhomogeneous models. The Multi-Model was transferred in a multi-model container (MMC) which contains a generic link model, describing the relationships between models elements together with the application models, described by metadata (more information is given by Fuchs in [21]). Reference models can be described by multi-model templates (MMT), which consist of partly filled MMC with metadata about the required application models [22]. Considering the different skills and tasks of the involved actors, it is not necessary for them to know all product models in every technical detailing. Therefore MMCs may consist of a task depending set of various application models, described as MMT. Input models, required for the

instantiation of processes and their output models are specified as multi-model templates.

3 The Derivation of the Context Model

Entity models cannot capture all the relationships between the entities of a construction project that affect collaboration processes. Some dependencies are beyond the scope of the reference templates (e.g. process patterns or MMTs), but can help to identify exceptions or conflicts. Therefore a context model to characterize the considered project situation is necessary, as described in [23]. Several context models for every entity type can be considered. As example we show the generation of the actor-context model, which should specify the available collaboration options for an actor to instantiate processes, use existing models or create new models in the current state of all project entities. As the actor context model reflects the relationship information of this actor to other entities, their structure follows the main aspects of the derived entity ontologies. In our case, the actor context model consists of relationship information from this actor to processes, to multi-models and to other actors. The generation of the context model occurs in three steps (see Fig.2) as described below:

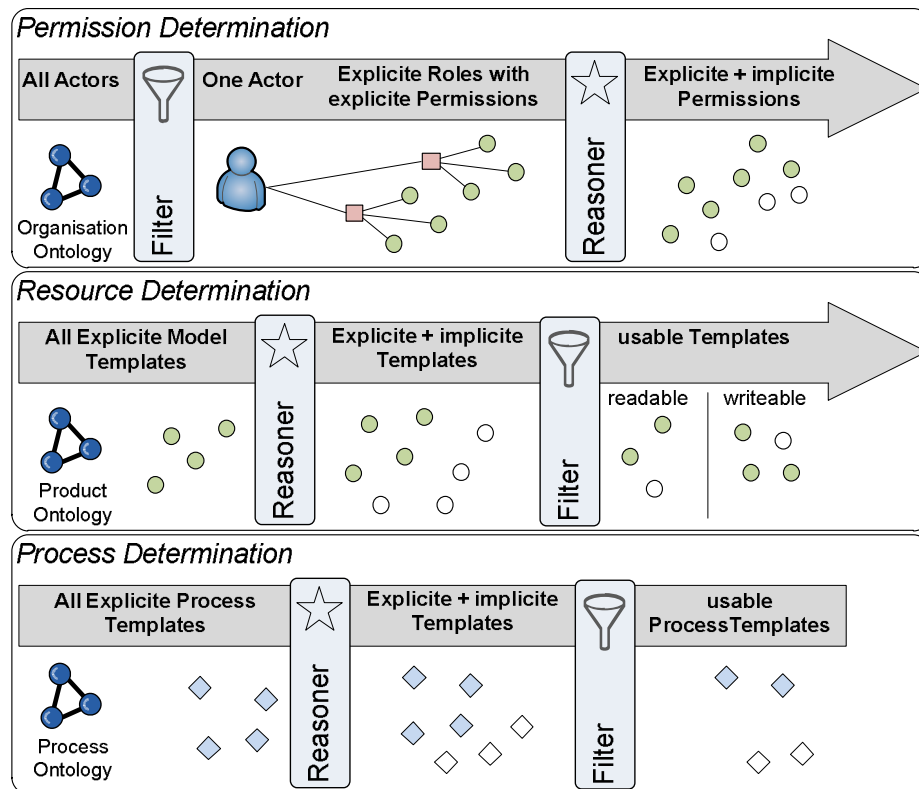


Fig. 2. Context model determination in three steps

(1) *Permission determination*: To determine utilizable processes and product models it is necessary in a first step, to collect actor roles with their permissions. The explicit permissions are determined from the actor instance of the organization ontology. If roles are organized hierarchic (e.g. OmniClass Organizational Roles [4]) so explicitly assigned roles (e.g. 34-251100 Space Designer) includes also implicit roles (e.g. 34-250000 Interior Designer). The same can apply for permissions: From a write permission can be derived the read permission, which can be necessary to start a process. It should be noted that the permission accumulation of roles should correspond to specific conditions (e.g. Separation of Duty [24]). To generate the extended permissions list, the organization ontology is evaluated by an ontology Reasoner who can derives further facts, either from the description of the organisation structure as well as from the instance data.

(2) *Resource determination*: In the Resource ontology explicit process templates are specified by their metadata. The semantic and structure of these metadata can be used to evaluate similarities, dependencies and inheritance structures. In our example, we limit ourselves to four class attributes: Domain, Project Phase, Level of Detail and Status, whose structure and vocabulary is defined in the construction core ontology. In step two, the structure of the metadata can now be used to inflate the object space of the product model templates (MMT). For example, a MMT has the domain attribute "BIM.BES.LBS", project phase "tender preparation", the Level of Detail value "5" (fine) and the status "beta", it can shortly described as follows: MMT(BIM.BES.LBS, tender preparation, LoD 5, status beta). This template also implicitly corresponds to even more coarsely defined templates, as e.g. MMT(BIM, Tender, LOD 1, beta). In addition we can use the information about the included application models to determined potentially compassable multi-model Templates (without linkmodel). In this manner, the object space of the product model templates can be expanded with a "implicit hull". A subsequent filter uses the permissions identified in Step 1 as well as actor competences and capacity and collect readable and writable product model templates.

(3) *Identifying usable processes*: A similar procedure is applied in the evaluation of process templates. A Reasoner generates a transitive hull by the recombination of explicit processes from the process ontology. New implicit (in this case transitive) processes can occur, which have the same input model like the first process and the same output model as the last process. The enlarged object space is reduced again by a filter using the previously determined readable (input) and writable (output) models.

If we connect in a last step, the process templates with the available input model instances, we obtain (for the current project context) all processes which can be used and all product models, which can be created. Thereby, if implicit models are potential input models for process templates, they may also increase the number of executable processes. Therefore, iterations are worthwhile.

As result a rich actor-context model is created with added further elements by the use of Ontologies. In a similar manner, the context models for the consideration of processes (collect usable actors and product models) and product models (shows authorized actors and executable processes) can be determined.

4 Results, Pilot and Outlook

By the use of the context model it is possible to identify similarities as well as transitivity's to determine on the basis of explicitly specified processes further implicit processes and suitable implicit models in the form of an "implicit envelope". It turns out that with the use of additional context information it is possible, to improve the selection of proportionate templates either qualitatively (by selection) and quantitatively (by the generation of the implicit envelope). So a context-dependent prediction is possible, whether a specified process can be started or a special model is available (or can be generated) in a current situation of the project. As a result, we obtain an Approach for generating rich context models, which allows the use of structural similarities and dependencies of product models within processes. Using semantically described task and model templates on the one side and a rich context model describing the project context on the other side should ensure that all partners receive a situation-specific selection of process and model templates which are necessary for their task and reflect the project situation at the current moment.

The use of the presented approach is being evaluated in the German BMBF research project Mefisto aims at overcoming client-contractor interoperability problems in construction processes based on partnership [25]. The overall architecture of the Mefisto collaboration platform is a hybrid SOA-based system on a Microsoft Azure cloud environment [26] with various platform services providing for the integration of local legacy applications, such as 3D CAD, scheduling, quantity take-off, ERP and PPM systems. So each project partner continues use its own familiar tools and environment meanwhile the communication with other partners is achieved by means of harmonised Multi-Model Containers, created and guided in process-centric manner by means of the project collaboration ontology. The use of the in Chapter 2 presented ontologies (jena framework) and Reasoner (jess Reasoner) are here encapsulated as methods of platform services for horizontal and vertical information integration. In a pilot scenario for bidding processes as described in [25] MMTs for tenders and offers are exchanged and evaluated with potential suppliers. Due to the shortness of the paper, most of the addressed themes could only be briefly highlighted. Details can be found the technical reports of the Mefisto project.

Several Approaches for collaborative Networks exist, especially for virtual enterprises, professional virtual communities or collaborative virtual laboratories [2]. Plisson present in [27] an Ontology for virtual organisation breeding environments (VBE) focused on organisational structures. He uses actor and role concept without a context-and domain-specific relation.

The focus of this paper is not the invention of a new organizational model, but rather the integration of contextual and model-structure information in a collaboration structure. In the presented approach, only a reduced set of entity attributes was considered. By using data mining techniques it is possible to consider complex dynamic attribute sets. Also, we have only used the entity ontologies to determine the context model. Here the consideration of further (Project and World) factors is necessary. This approach can also be used for optimization purposes, for example, to define model templates that satisfy reasonable process orchestration, or to create Roles

which have accurately privileges identified in the transitive processes - so that as few as possible Roles are needed. In this paper, we focus on information-transforming processes in a project planning phase, but the approach can be adapted for use in subsequent phases - including material transforming processes.

Acknowledgments

The research described in this paper was enabled by the financial support of the German Ministry of Education and Research, Department of ICT under Contract No. 01|A09001A, which is gratefully acknowledged. Moreover, the authors would like point out that the results presented have been developed in collaboration with industry and academic partners (www.mefisto-bau.de) of the Mefisto project.

References

1. Binding, G.: Construction in the Middle Ages (in German), Wissenschaftliche Buchgesellschaft, Darmstadt, P. 269 ff (1993)
2. Camarinha-Matos, L.M. & Afsarmanesh, H.. A comprehensive modelling framework for collaborative networked organizations, *Journal of Intelligent Manufacturing*. (2007)
3. buildingSMART: Specifications of the Industry Foundation Classes (IFC) (2010) Standard, http://buildingSMART-tech.org/products/ifc_specification/ifc-releases
4. OmniClass – A Strategy for Classifying the Built Environment, Website: <http://www.omniclass.org/>
5. Sharmak W., Scherer R. J.: Process Risk Management using Configurable Process Models; In: Proc. PRO-VE '11, 12th IFIP Working Conference on VIRTUAL ENTERPRISES, São Paulo, Brazil (2011)
6. Weise, M.: Ein Ansatz zur Abbildung von Änderungen in der modell-basierten Objektplanung. Dresden: Dissertation, Institut für Bauinformatik, Technische Universität Dresden (2006)
7. Abowd, G. D. und Dey, A. K.: Towards a Better Understanding of Context and Context-Awareness. In: HUC '99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing, Pages 304–307. Springer-Verlag (1999)
9. Altmann, J.: Kooperative Softwareentwicklung – Rechnerunterstützte Koordination und Kooperation in Softwareprojekten. Dissertation, Linz. (1999)
10. Luczak, D., Bullinger, H.-J., Schlick, C. and Ziegler, J. (Hrsg.): Unterstützung flexibler Kooperation durch Software: Methoden, Systeme, Beispiele. Springer. Heidelberg. (2001)
11. Davies, J., Fensel, D. and van Harmelen, F.: Towards the Semantic Web: Ontology-Driven Knowledge Management. John Wiley & Sons (2003)
12. Studer, R., Benjamins, R. and Fensel, D.: Knowledge Engineering: Principles and Methods. *Data & Knowledge Engineering*, 25(1-2):161–198 (1998)
13. Baader, F.: The Description Logic Handbook : Theory, Implementation and Applications. Cambridge University Press, Cambridge (2003)

14. Haarslev, V. and Möller, R.: RACER System Description. In: International Joint Conference on Automated Reasoning, IJCAR' 2001, June 18-23, Siena, Italy, Seiten 701–705. Springer-Verlag (2001)
15. Friedman-Hill, E. J.: Jess, The Expert System Shell for the Java Platform, Sandia National Laboratories, Livermore, CA (2001)
16. Gomez-Perez, A, Fernandez-Lopez, M. and Corcho, O.: Ontological Engineering. Berlin, Germany, Springer-Verlag (2004)
17. Schapke S.-E., Kadolsky M., Scherer R. J.: Representing project information spaces based on semantic multi-model annotations; In: Proc. CONVR2011, International Conference on Construction Applications of Virtual Reality, Weimar, Germany (2011)
18. Hilbert F., Araujo L. and Scherer R.J.: Multi-model-based Access Control in Construction Projects, in Proc. Formal Aspects of Virtual Organisations (FAVO2011), Jeremy W. Bryans and John S. Fitzgerald (Eds.), Sao Paulo, Brazil (2011)
19. Object Management Group: Business Process Model and Notation (BPMN), <http://www.omg.org/spec/BPMN/>
20. MEFISTO: Management – Führung – Information – Simulation im Bauwesen. German BMBF research project (BMBF Project 01IA09001), Website: www.mefisto-bau.de
21. Fuchs S., Kadolsky M., Scherer R. J.: Formal Description of a Generic Multi-Model, Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE); In: 20th IEEE International Workshops on; Paris, France (2011)
22. Scherer, R. J., Schapke, S.-E. & Katranuschkov, P.: Mefisto: A Model, Information and Knowledge Platform for the Construction Industry (in German: “Mefisto: Eine Modell-, Informations- und Wissensplattform für das Bauwesen“), Project Presentation, BMBF Project 01IA09001, (2010)
23. Hilbert F.: Context sensitive access to multi-models (in German). In: Proc. 23rd European Conference Forum Bauinformatik. Cork, Irland. (2011)
24. Ferraiolo D. F., J. A. C., Kuhn D. R.: Role-based access control (rbac): Features and motivations, In Proceedings of 11th Annual Computer Security Application (1995)
25. Schapke S.-E., Fuchs S.: Mefisto – A multi-model-based platform for construction project management (in German), In: Proc. 2nd Mefisto Congress „MEFISTO: Management – Führung – Information – Simulation im Bauwesen“, pp. 11-41, Dresden, Germany (2011)
26. Microsoft Azure, <http://www.microsoft.com/windowsazure/>
27. Plisson, J., Ljubic, P., Mozetic, I. and Lavrac, N.: An ontology for virtual organization breeding environments. IEEE Trans. on Systems, Man, and Cybernetics, Vol. 37. (2007)