A Standards-based UML-Profile for Message-Based Information Dissemination

Jörn Guy Süß

Technische Universität Berlin, Germany
Computergestützte InformationsSysteme (CIS)
jgsuess@cs.tu-berlin.de

Abstract. Integration of information systems using message queues and eXtensible Markup Language (XML)-documents is attractive, because it is comparatively simple to establish and reliable in operation. Combined with transformation it has been called the 'preferred EAI engine'. Such integration systems are however difficult to design and manage, because requirements are hard to trace over a large number of components. The Unified Modelling Language (UML) is a universal language for modelling systems. It provides the ability to integrate through its underlying metamodel. This thesis develops a UML profile to model the aforementioned domain. In the defintion of that profile, it relies only on provisions of the UML standard. Specifically, it exclusively uses Object Constraint Language (OCL) constraints on the level of the UML metamodel. The results are verified through an example from industryand an OCL-Profile validator.

The thesis addresses two areas: standardized UML as a language for blueprints of industrial systems, and standardized XML as a language for the creation of enterprise integration solutions. Section 1 describes current use of Profiles and contributions to enable standards-based Profile definition. Section 2 introduces the application domain and research basis of the profile definition. Section 3 describes some preliminary results and reports on the status and timetable.

1 UML applied for Blueprinting

One of the intended uses of the UML is the consistent design of business systems in the form of a blueprint. UML Profiles are intended as a means to adapt the UML to a business domain. Research in this thesis practically validates, if this approach is workable and feasible. The following section describes the UML Profile mechanism, state of the art and limitations in its application, and the author's approach and contribution to remedy these limitations.

The UML is the result of the unification of a set of object oriented modelling languages and as such reflects concepts of object oriented programming languages and their contemporary modelling techniques. Its intended field of application however encompasses business systems in general:

> The Unified Modeling Language (UML) is a language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.[1]

To validate this claim, uses of UML have to be categorized. Application domains are not very useful as categories, because the set is open-ended and criteria of success are subjective. The alternative is categorization along the role of the model in the process[2], as a sketch, blueprint or program. Here factors are the degree of consistency of model elements among each other and with the system under discussion, and how the model is transferred or used for communication.

In Sketching, the model content is made up of visual fragments on the whiteboard of the developer. They are transient, used for visual communication only and kept consistent by the mind of the architect or developer. In Blueprinting, models are kept in a tool along with diagrams, which show different viewpoints of the system under discussion. The models thus are persistent and can be communicated as files among a team. They are kept consistent by integrity constraints built into the meta-model of the tool or the model itself. In Programming, models are eventually mapped directly to executable components by means of functions without any human intervention. Integrity constraints ensure the semantics of the model at the domain level as in Blueprinting and additionally those enforced by the technological constraints of the target platform. The extreme form of the last mode is the Model Driven Architecture (MDA)[3], where several different models are used for the whole process down to the point of execution.

Since the UML is an OMG-standardized language, its promise in all three modes of usage outlined above, lies in the exchangeabiliy of the description of the modelled system under discussion. Unification should eleviate the need to rely on propietary formats, be it special symbols in Sketching, special file formats and constraint systems in Blueprinting or vendor-provided code generators in Programming. Inversely, this implies that in any of those three modes, the standards provided effectively cover everything which needs to be defined to use the UML in such a way. Any lack in definition thus directly affects viability. Expert opinions on viability of the three modes of usage vary greatly. Fowler sees Sketching as the only viable mode. Initiatives like TopModel hold that MDA is truly capable to generate systems[4,5]. This thesis looks at the viability of UML for Blueprinting, because viability of UML for Sketching is obvious and viability for Programming cannot be decided yet, as the MDA is still in the process of standard definition.

Standards required for Blueprinting use of UML were complete with the release of version 1.1 in 1997. It contains extension mechanisms enabling tailoring of the language to domains, using stereotypes and a constraint language to enforce domain requirements in modelling. Such collections of domain-descriptions are released in 'UML Profiles'. In 1999, their form was more precisely defined in the "Requirements for UML Profiles"[6]:

> A UML Profile is a predefined set of Stereotypes, TaggedValues, Constraints, and notation icons that collectively specialize and tailor the UML for a specific domain or process (e.g. Unified Process Profile). A Profile does not extend UML by adding any new basic concepts. Instead, it provides conventions for applying and specializing standard UML to a particular environment or domain.

It states further, that Profiles should subset, not extend, the UML standard. We assume, that because compliant tools are required to check UML Well-formedness Rules (WFRs), which are given in OCL, they should either contain an interpreter or be built using a compiler for that language. Consequently, Profiles should exclusively be defined using OCL WFRs which can be executed to validate the model in such an environment.

Table 1 shows a survey by the author to establish the state of Profiles available. It shows that none of them fulfills the aformentioned requirement. Instead, they divide into documents which define Sketching Conventions, addressing presentation issues of diagrams, or Metamodel Extensions, introducing new Meta-Object Facility (MOF) metamodels that cannot be correlated with UML without the aid of mappings as proposed by Queries, Views and Transformations (QVT)[7]. The underlying problem is the lack of infrastructure for the validation of OCL constraints in UML tools in particular, and the lack of standardized model manipulation infrastructures in general. As a result, the UML WFRs themselves contain inconsistencies and errors. By application of the USE system, Gogolla **??** found 39 out of 71 OCL expressions in the UML 1.3 standard to be erroneous. 20 of those errors were minor, but 13 were type errors and 5 even serious problems. OCLE, introduced by Choirean**??**, contains a corrected set of UML 1.5 WFRs. However, the author has found that the tool itself produces models, it itself identifies as invalid (e.g. all Use Case Models). Also, many WFRs were altered in OCLE to work around shortcomings of models stored by the Rational Rose tool. Summarily, Profile authors do not have the technical means to create, check for correctness and apply Profiles, because the necessary support for OCL WFRs is lacking. As a work-around, vendors create proprietary domain extensions to their products, eliminating UML's capacity for exchange of models.

Because these problems emanate from the fact that UML tools cannot provide the necessary service, and users will not give up their tools, there is a need to separate the concerns of model manipulation by a user and model processing by services. The validation of WFRs and Profiles is recognized as such a service.

As a result, the author has designed and implemented the distributed model service framework EVE [8] and equipped it with a OCL validator service operating on MOF-level M2 [9]. This validator is used to carry out the standards-based Profile validation. The framework itself has further applications, e.g. as a basis for QVT[10] and integration with semantic representation standards[**?**]. The framework and validator can be applied to MOF-based models in general.

Profile		UML	ST	TV	CO	IC	MM Ext.	Just.	UML Val.	Format
Integrating Function Blocks into the UML	[11]	(2.0)	6	7	6 (OCL)	0	(2.0)	yes	no	no
'Agent-Oriented Modeling' / 'External AOR Models'	[12]	?	22	0	21 (PE)	yes	3 ?	no	no	no
Aspect Oriented Modeling	[13]	(1.x)	5	?	11 (PE) 4 (OCL)	0	yes	yes	no	no
Modeling Workflow and Business Processes	[14]	(1.x)	7	0	22 (PE)	10	yes	yes	no	no
Real-Time Constraints with the OCL	[15]	(2.0)	7	0	0	0	yes	yes	no	no
Requirements engineering with KAOS	[16]	?	21	16	15 (PE) 1 (OL)	4	yes	no	no	no
MultiTEL	[17]	(1.x)	3	?	?	0	yes	yes	no	no
Automated Business Processes with BPEL 1.0	[18]	1.4	24	0	?	0	?	yes	no	no
Modelling QoS	[19]	(1.4)	14	20	19 (PE)	0	yes	yes	no	no
Reverse Engineering, Prog. Comprehension, and Reengineering	[20]	(1.x)	43	14	5 (PE)	0	yes	yes	no	no
XML Schema	[21]	(1.x)	26	0	49 (PE)	0	yes	yes	no	no
Model-Based Risk Assessment	[22]	(1.x)	12	0	0	11	no	yes	no	no
Service-oriented Architectures	[23]	(1.x)	7	0	0	0	?	yes	no	no
Software Architecture Descriptions	[24]	(1.x)	?	?	?	14	yes	yes	no	no
Data Modeling	[25]	(1.x)	13	>3	?	3	?	yes	no	no
archetypes and archetype patterns	[26]	?	4	0	?	1	?	yes	no	no
Navigation and Presentation in Web Applications	[27]	(1.x)	18	0	?	21	?	yes	no	no
EJB (Status: JCP Public Draft)	[28]	(1.x)	24	19	?	0	no	yes	no	no
CORBA	[29]	1.3	35	10	72 (OCL)	0	yes	yes	no	no
CORBA Components	[30]	(1.4)	15	2	45	0	yes	yes	no	no
UML Profile and Interchange Models for EAI	[31]	(1.x)	14	10	0	0	yes	yes	no	no
Enterprise Distributed Object Computing	[32]	(1.x)	46	15?	42	7	yes	yes	no	no
Schedulability, Performance, and Time Specification	[33]	1.4	57	137	13 (PE)	0	yes	yes	no	no
Testing	[34]	(1.x)	21	1	26 (PE)	0	yes	yes	no	no

Table 1. UML Profiles and Related activities based on [35]. Columns show the name, uml version, number of stereotypes, tagged values, constraints and icons, whether a metamodel is introduced, whether justification for profiling is given, if OCL-based validation is possible and if the document follows the format in the standard.

2 The domain of Information Dissemination (ID)

Enterprise Application Integration (EAI) is inaccesible to small and medium sized businesses because of high cost and complexity. Standards-based methods to plan for EAI, which allow validation of the plan, do not exist. Research in this thesis develops such a method within the scope of a UML Profile definition for a subdomain of message-based EAI, called ID.

In message-based EAI, information systems are connected by marshalling information as XML documents. Every noteworthy change to some piece of information held in a local information system is published as an XML document based on a Document Type Definition (DTD) or XML Schema Definition (XSD) and forwarded to a subscriber. To avoid the creation of a new DTD for each connection, which would lead to the maintenance of $2(n^2 - n)$ connections in a tightly meshed system[36], source and target formats are mapped to intermediate standard formats using eXtensible Stylesheet Language Transformations (XSLT)[37]. Thus maintenance is improved and a canonical set of enterprise events established, which in turn simplifies the connection of new systems. On the other hand, this approach potentially requires application of a number of transformations. These can be costly in terms of effort when applied to large documents. Thus, transformers are distributed among computing nodes. To ensure reliable delivery of documents, transactional message queues are used as connectors. Traditional management techniques for such projects involve document-based description of all artifacts. These are tedious to maintain and lack the traceability required for management. Vendor tools are available, but do not cover the whole process and often only provide visualization, not validation.

The thesis assumes that UML can be usefully applied to this field. To reduce complexity of the domain, it is viewed as simplified version of the integration server reference model[38], which contains the following layers:

Message Transport provides transport, persistency, security and transaction, made up of message queues and adapters.

Format Engine recognizes the format of documents and transforms them if required. Transformation is stateless and free of side-effects.

Rule Engine routes messages based on rules defined against message content. Routing is also stateless and side-effect free. Because it is generally impossible to determine that selection criteria actually cover all possible messages, the rule engine contains a dead letter queue holding undelivered messages. Because selection criteria in rules may overlap, rules need to be prioritized.

Business Process Automation orchestrates different business processes in a workflow. It is notified of every message that is being received or passed. It can issue messages on its own to compensate steps of failed transactions that have already been committed on local systems.

Information Dissemination (ID) is defined to comprise the first two layers, while the others are only available in full integration solutions for larger enterprises.

Consequently, ID does not overlap with the Electronic Business XML Initiative (ebXML) or Business Process Execution Language for Web Services (BPEL4WS). As a further simplification, projects are expected not to involve (and therefor do not need to model directly) issues of security arising from organisational boundaries, such as a message passing from one company to the next. However, there are indirect ways to express such situations. The method also is not intended to model Computer Supported Cooperative Work in the office.

To correlate the ID reference model with properties of available market components these were analysed in context of two diploma thesis [39,40] and summarized in the description of a minimal abstract broker machine [41].

The method proceeds in the anticipated manner: Analysis with Use Cases diagrams establishes the participating systems and occuring events, Design with Class diagrams describes documents as Class trees and mappings as Associations, Assembly and deployment with Component Diagrams show how Classes map to Artifacts, which aggregate into Components that are distributed to Nodes.

The corresponding Profile consists of seven Sub-Profiles, which methodically divide into two subsets: a local set of four Profiles which ensure consistency within each modelling phase, and a global set of three Profiles which ensure consistency across modelling phases. This subdivision is only a means to structure the overall Profile, which enables the user to selectively work on phases in the process.

3 Status of Work / Preliminary Results

Implementation of the service platform, validator, and the local Profile for Analysis is finished. The local Profile for Design and the connecting global Profile are nearing completion. The remaining ones are drafted. A real industry example is modelled in parallel as a test case to ensure that the method laid down in the Profiles can actually be applied.

Work up to this point has uncovered a number of imprecisions and limitations of the standards, both in the definition of the relation of UML and OCL as exhibited in Profiles and the OCL as a language for authoring Profile constraints. These include:

- The OCL grammar structures files, packages and contexts. In Profiles, constraints reside as free text inside the metamodel. The standard does not define, which syntactical element should be stored there.
- Definition constraints extend the model and can reference each other. However, the standard does not define the order in which these are to be resolved.
- Transient closures, a recurring part in Profile definitions, must be tediously defined, as OCL does not provide a short-hand notation.
- Language-dependent expressions are expected in many model elements. OCL's string manipulation functions are weak, so consistency between elements of the model and parts of an expression is hard to establish without writing

parsing code, which obscures the semantics of the constraint. The author suggests inclusion of an operator for regular expressions, as these are based on standards and sound formalism.
- It is possible to cast from an OCL Set to a Sequence, which introduces non-determinism in constraints. This is unpleasant, as validation becomes unreliable. The author suggests, that respective operations should either be removed, require an ordering function as a mandatory parameter, or interpreters and compilers be required to flag the construct and all dependencies.

All of the points addressed above are still present in the current revisions of the OMG standards. Some additional effects, like mandatory flattening, make work cumbersome in the UML 1.4 version chosen as the basis of this project.

Work is expected to finish in the first or second quarter of 2005. It will comprise the aforementioned technical environment and a profile validated using it. As a contribution, it will enable researchers and engineers to create standard UML Profiles which can be used to validate models.

References

1. Object Management Group (OMG): Unified Modeling Language Specification, Version 1.4. (2001) `http://cgi.omg.org/docs/formal/01-09-67.pdf`. 1
2. Fowler, M.: What is the point of the UML? In Stevens, P., Whittle, J., Booch, G., eds.: UML 2003 - The Unified Modeling Language. Model Languages and Applications. 6th International Conference, San Francisco, CA, USA, October 2003, Proceedings. Volume 2863 of LNCS., Springer (2003) 325 1
3. Miller, J., Mukerji, J.: Model Driven Architecture(MDA). Technical Report ormsc/2001-07-01, Object Management Group(OMG), Architecture Board ORMSC (2001) 1
4. Bézivin, J., Farcet, N., Jézéquel, J.M., Langlois, B., Pollet, D.: Reflective model driven engineering. In Stevens, P., Whittle, J., Booch, G., eds.: UML 2003 - The Unified Modeling Language. Model Languages and Applications. 6th International Conference, San Francisco, CA, USA, October 2003, Proceedings. Volume 2863 of LNCS., Springer (2003) 175–189 1
5. Muller, P.A., Studer, P., Bézivin, J.: Platform independent web applikation modeling. In Stevens, P., Whittle, J., Booch, G., eds.: UML 2003 - The Unified Modeling Language. Model Languages and Applications. 6th International Conference, San Francisco, CA, USA, October 2003, Proceedings. Volume 2863 of LNCS., Springer (2003) 220–233 1
6. Object Modelling Group: Requirements for UML Profiles. 1.0 edn. (1999) 1
7. Object Management Group (OMG): Request for Proposal: MOF 2.0 Query / Views / Transformations RFP. (2002) `http://www.omg.org/cgi-bin/apps/do_doc?ad/2002-04-10.pdf`. 1
8. Süß, J.G., Leicher, A., Weber, H., Kutsche, R.D.: Model-centric engineering with the evolution and validation environment. In Stevens, P., Whittle, J., Booch, G., eds.: UML 2003 - The Unified Modeling Language: Modeling Languages and Applications, 6th International Conference, San Francisco, CA, USA. Volume 2863 of LNCS., Springer (2003) 31 – 43 1

9. Süß, J.G., Leicher, A., Chabarek, F.: Software model engineering and reuse with the evolution and validation environment. In Guelfi, N., Astesiano, E., Reggio, G., eds.: Scientific Engineering for Distributed Java Applications: International Workshop, FIDJI 2003 Luxembourg-Kirchberg. LNCS, Springer (2003) (to appear). 1
10. Süß, J.G., Leicher, A., Busse, S.: Oclprime - environment and language for model query, views and transformations. In: Workshop, OCL 2.0 - Industry standard or scientific playground?, UML 2003- The Unified Modeling Language: Modeling Languages and Applications, 6th International Conference, to appear. (2003) 1
11. Süß, J.G., Leicher, A.: Augmenting domain specific uml models with rdf. WISME04 at UML04, to appear (2004) 1
12. Heverhagen, T., Tracht, R., Hirschfeld, R.: A profile for integrating function blocks into the unified modeling language. Software Tools for Technology Transfer (2003) Talk at Specification and Validation of UML Models for Real Time Embedded Systems, SVERTS'03. 1
13. Wagner, G.: A uml profile for external agent-object-relationship (aor) models. In Giunchiglia, F., Odell, J., Weiß, G., eds.: AOSE. Volume 2585 of Lecture Notes in Computer Science., Springer (2003) 138–149 Agent-Oriented Software Engineering III, Third International Workshop, AOSE 2002, Bologna, Italy, July 15, 2002, Revised Papers and Invited Contributions. 1
14. Aldawud, O., Elrad, T., Bader, A.: Uml profile for aspect-oriented software development (2003) 1
15. v. Bochmann, G.: Activity nets: A UML profile for modeling workflow and business processes (2000) 1
16. Flake, S., Mueller, W.: A UML profile for real-time constraints with the OCL (2002) 1
17. Heaven, W., Finkelstein, A.: A UML profile to support requirements engineering with KAOS (2002) 1
18. [lff, A.V., Es, A.U., Troya, T., Fuentes, L.: Using UML profiles for documenting web-based application frameworks (2002) 1
19. Amsden, J., Gardner, T., Griffin, C., Iyenga, S.: Draft uml 1.4 profile for automated business processes with a mapping to bpel 1.0. Draft, IBM UK Laboratories, Hursley Park (2003) 1
20. Aagedal, J.Ø., Jr., E.F.E.: Modelling QoS: Towards a UML profile. In Jézéquel, J.M., Hussmann, H., Cook, S., eds.: UML 2002 - The Unified Modeling Language. Model Engineering, Languages, Concepts, and Tools. 5th International Conference, Dresden, Germany, September/October 2002, Proceedings. Volume 2460 of LNCS., Springer (2002) 275–289 1
21. St-Denis, G.: Rcr: un profil uml pour la rétroconception, la compréhension et la réingénierie de logiciels. Master's thesis, Université de Montréal, Département d'informatique et de recherche opérationnelle Faculté des arts et des sciences (2001) 1
22. Martin Bernauer, Gerti Kappel, G.K.: Representing xml schema in uml - a uml profile for xml schema. Technical report, Business Informatics Group, Vienna University of Technology, Austria (2004) 1
23. Houmb, S.H., den Braber, F., Lund, M.S., Stolen, K.: Towards a UML profile for model-based risk assessment. In Jürjens, J., Cengarle, M.V., Fernandez, E.B., Rumpe, B., Sandner, R., eds.: Critical Systems Development with UML – Proceedings of the UML'02 workshop, Technische Universität München, Institut für Informatik (2002) 79–92 1

24. Heckel, R., Lohmann, M., Thöne, S.: Towards a uml profile for service-oriented architectures. In: Workshop on Model Driven Architecture: Foundations and Applications. (2003) profiles/MDAFA-2003-FinalVersion.pdf. 1
25. Kandé, M.M., Strohmeier, A.: Towards a UML Profile for Software Architecture Descriptions. In Evans, A., Kent, S., Selic, B., eds.: UML 2000 - The Unified Modeling Language, Advancing the Standard, Third International Conference, York, UK, October 2-6, 2000, Proceedings. Volume 1939 of Lecture Notes in Computer Science., CH-1015 Lausanne EPFL, Switzerland, Springer (2000) 1
26. Gornik, D.: Uml data modeling profile (2002) profiles/RationalUML-RDB-Profile.pdf. 1
27. Arlow, J., Neustadt, I. The Addison-Wesley Object Technology Series. In: Archetypes and archetype patterns. http://www.awbc.com/samplechapter/032111230x.pdf edn. Addison Wesley (2002) 3–20 profiles/032111230X.pdf. 1
28. Koch, N., Baumeister, H., Hennicker, R., Mandel, L.: Extending uml to model navigation and presentation in web applications. In Winters, G., Winters, J., eds.: UML 2000 - Modeling Web Applications in the UML Workshop, Proceedings, Oettingenstr. 67; 80538 München, Germany (2000) profiles/ExtendingUML.pdf. 1
29. Greenfield, J.: Uml profile for ejb (2001) profiles/UMLProfileForEJBPublicDraft.pdf. 1
30. : UML Profile for CORBA Specification. (2002) profiles/02-04-01.pdf. 1
31. Born, D.M., Reznik, J.: UML Profile for CORBA Components. (2002) profiles/07-1_A_UML_Profile_for_CORBA_Components.pdf. 1
32. : UML Profile and Interchange Models for Enterprise Application Integration (EAI) Specification. (2002) profiles/02-01-21.pdf. 1
33. : UML Profile for Enterprise Distributed Object Computing. (2001) profiles/01-12-04.pdf. 1
34. : UML Profile for Schedulability, Performance, and Time Specification. (2003) profiles/03-03-02.pdf. 1
35. : UML Testing Profile. (2003) profiles/03-07-01.pdf. 1
36. Jeckle, M.: UML Profiles und sonstige UML-bezogene Aktivitäten. http://www.jeckle.de/uml_spec.htm (2002) 1
37. Linthicum, D.S.: Enterprise Application Integration. Information Technology Series. Addison Wesley Longman, Reading, Massachusetts (1999) 2
38. Clark (Eds), J.: "XML Transformations (XSLT) Version 1.1". W3C Working Draft (2000) http://www.w3.org/TR/xslt11. 2
39. : e-Business und Enterprise Application Integration: Der Schlüssel zum e-Erfolg. EAI Report, META Group (2001) 2
40. Mewes, M.: Anwendungsbezogener vergleich von message-brokern und konzeption eines optimalen message-brokers. Diplomarbeit, Technical University Berlin, Computation and Information Structures, TU Berlin FB13 Franklinstraße 28/29 · D-10587 Berlin (2000) 2
41. Emilov, E.: Evaluation von Message Brokern unter Berücksichtigung geschäftskritischer Anforderungen. Diplomarbeit, Technical University Berlin, Computation and Information Structures, TU Berlin FB13 Franklinstraße 28/29 · D-10587 Berlin (2001) 2
42. Süß, J.G., Mewes, M.: An architectural proposal for enterprise message brokers. In Emmerich, W., Tai, S., eds.: Engineering Distributed Objects (EDO 2000). Volume 1999 of LNCS., Berlin, Heidelberg, Springer (2000) 2