Biuletyn WAT
Vol. LXII, Nr 2, 2013

UML profiles for architecture description of an integration platform

TOMASZ GÓRSKI

Military University of Technology, Cybernetics Faculty, Information Systems Institute,
gen. S. Kaliskiego Str. 2, 00-908 Warsaw, Poland,
gorski@wat.edu.pl, tomasz.gorski@rightsolution.pl

Abstract. The paper presents a new Unified Modeling Language profiles devoted to modeling the architecture of the integration platform. The paper contains a description of the notations and languages of information systems architecture modeling, such as BPMN, UML, and SoaML. These notations and modeling languages do not provide a complete set of semantic structures necessary to present integration platform's architecture. Therefore, a set of UML stereotypes was proposed which describes the needed additional semantic structures. These stereotypes have been grouped into two UML profiles, "UML Profile for Integration Platform" and "UML Profile for Integration Flows". In the first profile, stereotypes relating to the structural elements of the integration platform were placed. The second profile contains stereotypes representing mediation mechanisms. The paper presents a new semantic extension of an activity diagram to model the mediation flows. Thus, it was proposed a new UML diagram: mediation flows diagram.
Keywords: integration, UML profiles, architecture of information system, mediation flows

1. Introduction

Large companies usually have many information systems which support their business. This situation results from the fact that the introduction of new services or refinement of the existing ones, over the years, were associated with the development of new IT systems. Often, they are produced in newer technology than existing information systems. This implies the need to build integration solutions comprised of IT systems and a communication layer that enables cooperation between these systems because it is necessary to provide appropriate support to business processes by means

of information technology. A solution of this type is called an integration platform or an integration solution. Service Oriented Architecture (SOA) is the concept of a system architecture which defines services. A service represents a part of information system's functionality and is defined by an interface. Interface definition is independent from the implementation provided by an information system. SOA is widely described in literature [6, 7, 10, 11, 24]. When designing integration solutions, it is essential to be able to model their complete architectural description. For this purpose, one needs to have an architectural views model which allows for modeling integration platforms and a set of model elements enabling presentation of the integration platform's entire architecture [1, 13, 25]. The paper uses an architectural views model "1+5" presented in [13, 15]. From the perspective of notation, models are expressed in the following notations: BPMN, UML, and SoaML. Integration of many various IT systems makes the integration project highly complex [12, 15, 16, 24]. Using clearly defined set of modeling components, the Integration Architect [16] is able to manage the complexity of the models and their relationships.

The aim of the paper is to present a set of UML stereotypes which was proposed due to model integration platform's structural elements and mediation flows. These UML stereotypes have been grouped into two UML profiles, *UML Profile for Integration Platform* and *UML Profile for Integration Flows*. In the first profile, stereotypes relating to the structural elements of the integration platform were placed. The second profile contains stereotypes representing mediation mechanisms, which are used in mediation flows. Furthermore, the paper presents a new semantic extension of an UML activity diagram to model the mediation flows. It is a proposal of a new UML diagram: mediation flows diagram.

The remainder of the paper is structured as follows. Section 2 presents views, models, and diagrams of the architectural views model "1+5". Section 3 contains an overview of publications dedicated to similar problems. Section 4 describes the stereotypes which are enclosed in *UML Profile for Integration Platform*. In this section, the examples of using selected stereotypes, from that profile, on UML diagrams were shown. Section 5 presents details of the second profile *UML Profile for Integration Flows*. In this section, a new UML diagram was introduced: mediation flows diagram. The section shows application examples of mediation flows diagram. Section 6 concludes the paper, summing up the subject and outlining the advantages of using the proposed UML profiles. Besides, the conclusion outlines the directions for further work.

2. Architectural views model "1+5"

Consistency and quality of the architectural description of IT solutions is a significant matter and is a subject of studies today [1, 2]. A variety of models exist, with different sets of architectural views, such as e.g.: "4+1", RM-ODP, Siemens, and

SEI views [25]. Yet, they do not allow for a complete description of the integration solutions architecture. The model of architectural views "1+5" proposed here has been accommodated to suit the process of an integration platform designing [13, 14, 16]. The following architectural views have been distinguished within the model: *Integrated Processes*, *Use Cases*, *Logical*, *Integrated Services*, *Contracts*, and *Deployment*.

The view of Integrated Processes is the basic architectural view here. In this view, business processes are modeled which are devoted to automation on the integration platform. The next four views (Use Cases, Logical, Integrated Services and Contracts) present the integration platform design. The *Use Cases* view contains functional requirements for the system being integrated within the platform. The view of *Integrated Services* presents services exposed from IT systems and the way how they are connected to the service bus. The *Contracts* view shows components representing IT systems and contracts defined between them. This view encompasses also mediation flows for each contract. The last view — *Deployment* — shows the way how the integration platform elements are deployed on a certain runtime environment. Figure 1 illustrates the architectural views model "1+5".

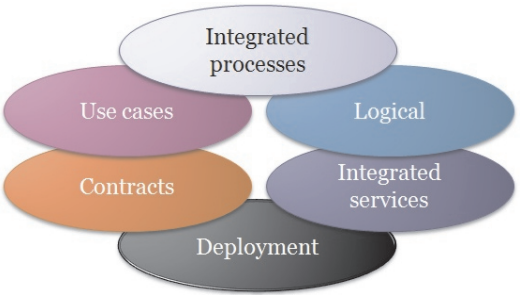


Fig. 1. The architectural views model "1+5"

A detailed description of the architectural views model referred to the above and examples of its application can be found in the literature of the subject [13, 14, 15, 16]. In the approach being analyzed here, models and diagrams of BPMN [3, 5] and UML [23] languages have been used for modeling the integration platform architecture (tab. 1).

Table 1

Elements for modeling the integration platform architecture

Model	View	Diagram
Processes	Integrated Processes	(BPMN) Business Process
Use Cases	Use Cases	(UML) Use Case
		(UML) Activity

cd. table 1

Design	Logical	(UML) Sequence
		(UML) Communication
		(UML) Class
Services	Integrated Services	(UML) Component
	Contracts	(UML) Component
		(UML) Activity
		(UML) Composite Structure
Deployment	Deployment	(UML) Deployment

Furthermore, stereotypes from Service oriented architecture Modeling Language (SoaML) UML profile were applied. SoaML is an open source specification project from the Object Management Group (OMG), describing a UML profile and metamodel for the modeling and design of services within a service-oriented architecture. The existing models and meta models for describing system architectures turned out to be insufficient to describe SOA in a precise and standardized way. The UML itself seems to be too general for the purpose of describing SOA and needed clarification and standardization of even basic terms. SoaML has been created to support the following modeling capabilities:

— Identifying services, dependencies between them and services requirements,
— Specifying services (functional capabilities, consumer expectations, the protocols and message exchange patterns),
— Defining service consumers and providers.

In the architectural views model there are used, among others, the following SoaML stereotypes:

— Capability — offered functionality or resource,
— Provider — functionality exposed on the enterprise service bus by information system,
— Consumer — the recipient of services from enterprise service bus.

Other stereotypes of SoaML can be also used in the design of integration platforms if there is a need to present relevant abstracts to the specific project.

The aforementioned modeling languages, however, do not provide full set of semantic structures necessary to describe an integration platform architecture. Therefore, there were proposed additional semantic constructions in form of stereotypes. These stereotypes were grouped into two UML profiles *UML Profile for Integration Platform* and *UML Profile for Integration Flows*.

3. Related work

An overview of the literature follows the process described in the Systematic Review method [19]. Main notation used for architecture documentation is the Unified Modeling Language [23]. UML is used for design, specification, and documentation of artifacts created during information system development process. Moreover, UML provides mechanisms which support extending of their semantics. Basic structure which allows for extending UML semantics is a stereotype. Furthermore, a collection of stereotypes can be aggregated in a profile. There are studies which deal with influence of stereotypes on the comprehension of UML diagrams [8]. In publications of the recent years, an UML language has been extended for different purposes. A SoaML is an example of UML profile for description of service oriented architecture [27]. Another example is an UML profile for the conceptual modeling of data-mining in data warehouses [28]. Moreover, the Object Management Group defined two profiles which are used to document SOA architecture. UML profile *Modeling and Analysis of Real-Time and Embedded Systems* (MARTE) [22] supports modeling of real-time systems and embedded systems and it is destined for performance analysis based on models. *UML Profile and Interchange Models for Enterprise Application Integration Specification* (EAI) is used to describe the mediation flows [21]. The second profile assumes that UML sequence diagrams are used to present mediation flows. From that profile one can use e.g.: «*EAIPrimitiveOperator*» stereotype to model a simple message processing and «*EAICompoundOperator*» stereotype to express complex message flows. The paper proposes to use UML activity diagrams instead. Furthermore, in the paper, a new authorial profile *UML Profile for Integration Flows* was proposed which groups integration patterns [18] in the following areas: messaging systems, message routing, message transformation, and messaging endpoints. The profile contains, among others, the following stereotypes for the mediation mechanisms: «*EnvelopeWrapper*», «*Translator*», «*ContentFilter*», «*Splitter*». This ensures ability to model complete mediation flows. Moreover, each of stereotypes has its own icon which can be easily recognized. With the use of partitions, responsibilities for performing activities in a mediation flow can be clearly defined. The full set of integration patterns can be found at address [9].

Besides, the paper presents a new, authorial profile *UML Profile for Integration Platform* including the «*IntegratedSystem*» and «*ESB*» stereotypes. In this profile, stereotypes relating to the structural elements of the integration platform were placed. The paper focuses on presenting these two new UML profiles for modeling integration platforms. Both UML profiles have been designed according to the requirements of "1+5" architectural views model. Using two profiles provides clear division of a set of stereotypes into a group of semantic elements connected with a structure of integration platform and those related to mediation flow mechanisms.

In publications of the recent years, an issue of service-oriented architecture is still a current one. Especially, the areas of adaptable decentralized service-oriented architecture and dynamically reconfigurable workflows are under recent studies [6, 11]. In context of workflows, BPMN standard is under scrutiny [3, 5]. Moreover, one of the most important issues in integration solutions is data exchange. In that field, there are studies which deal with development and evolution of XML-based languages [20]. There are also recent studies in the field of nonfunctional requirements which deliver proposals to detect errors in Enterprise Application Integration solutions [12] and deal with reliability, availability, and performance tradeoff [24].

4. UML Profile for Integration Platform

The *UML Profile for Integration Platform* contains stereotypes needed to show the elements of the structure of an integration platform. This profile was created in an IBM Rational Software Architect 8.0. The profile can be applied to projects of UML modeling. In that manner, there were obtained elements of notation and environment ready for modeling the structure of an integration platform. Stereotypes defined in the profile were shown in Table 2.

Table 2

Descriptions of stereotypes from profile *UML Profile for Integration Platform*

Stereotype name	UML classifier	UML diagram	Stereotype description
IntegratedSystem	Actor	Use case	Information system which is connected to the integration platform.
Protocol	Association	Deployment	Protocol used to communicate with the integration platform.
ESB	Component	Component	Enterprise service bus.
AdapterType	Component	Component	A type of adapter which is used by an integration platform.
ServiceRegistry	Component	Component	Service registry.

This profile is open for extension and additional stereotypes can be added which represent structural elements of integration solution.

The profile has been designed according to requirements of "1+5" architectural views model which is devoted for an integration platform architectural description [13, 14, 15, 16]. Later in the paper, an application of the profile was presented in three architectural views of the model "1+5": *Use Cases*, *Integrated services* and *Contracts*.

As an example, a railway company was chosen and area of railway transport. In the company, there are problems connected with rail links delays. Those problems touch the following branches: customer service, planning and scheduling, rolling stock management, and railway track management. Each branch poses its own information system. The basic problem is exchange of data originally created in each of information systems. As a solution, integration of existing information systems was proposed. Each system has to expose services with data originally created in it. Furthermore, each system has to specify needs for data from other company's information systems.

First of all, use case diagrams were created for each of information systems. In Figure 2, a UML use case diagram was presented which shows actors and use cases for *Planning and scheduling* information system. In the diagram, «*IntegratedSystem*» stereotype was used. This stereotype was applied to actors which represent other information systems in company. The stereotype was used to emphasize the fact that those information systems are connected through integration platform.

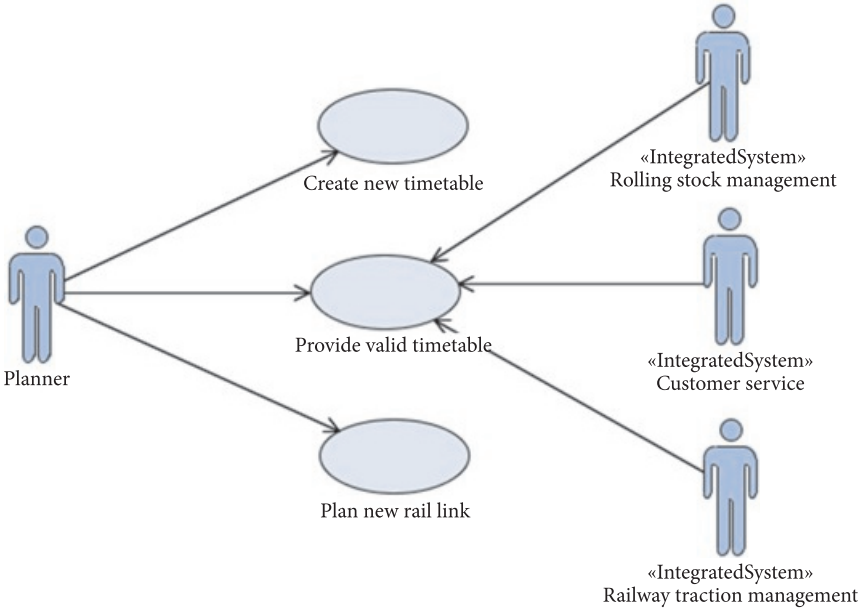


Fig. 2. Use case diagram for Planning and scheduling information system

Next, contracts were created. Contracts were presented in *Contracts view* in UML component diagrams. For each of use cases from use case diagram, a new contract was created in a component diagram. Furthermore, new components are created. One component for the modeled information system and one component for each of actors on use case diagram. In Figure 3, there was presented mapping of elements

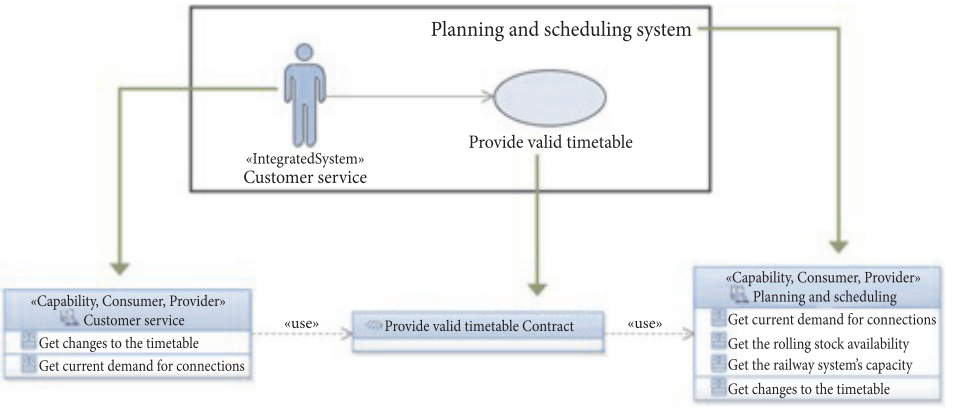


Fig. 3. Mapping of and UML use case diagram to a UML component diagram with contract

form use case diagram to elements in component diagram. In component diagram, stereotypes from SoaML language: «*Capability*», «*Consumer*», and «*Provider*» were used.

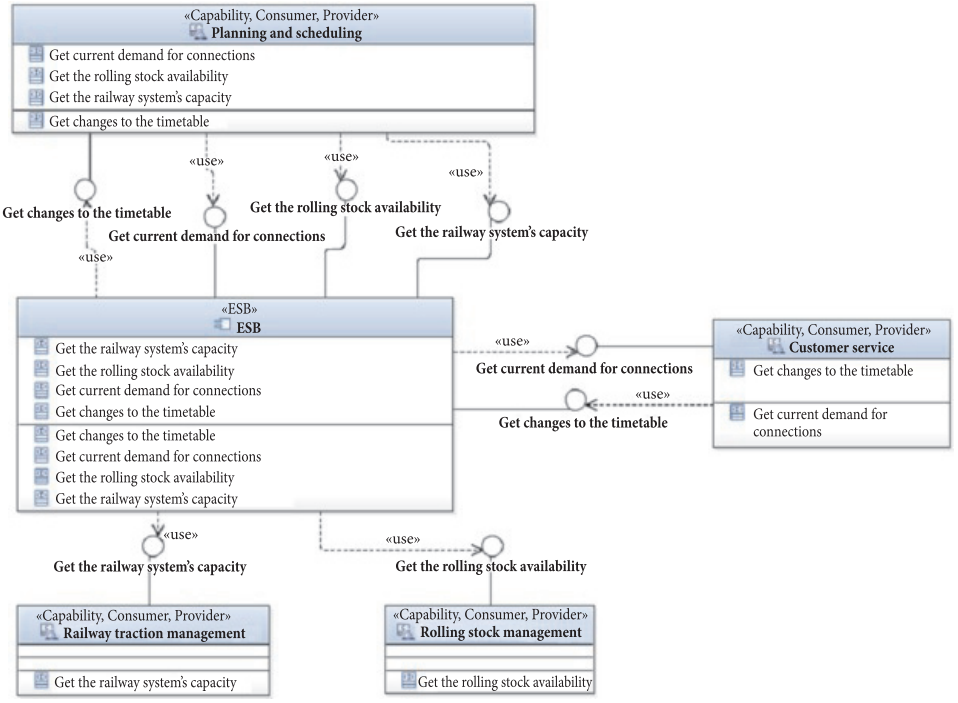


Fig. 4. UML component diagram for information systems of a railway company

In the *Integrated services view*, services exposed from information systems and services required by information systems were presented. In Figure 4, a component diagram was presented. In a component diagram, there were used: stereotype «*ESB*»for enterprise service bus and stereotypes from SoaML language: «*Capability*», «*Consumer*», and «*Provider*».

5. UML Profile for Integration Flows

The issue of integration patterns is discussed in [18]. The authors of this item also run a website with a full range of integration patterns [9]. They have divided integration patterns into the following categories: integration styles, messaging systems, messaging channel, message construction, message routing**,** message transformation, messaging endpoints, and system management.

The profile *UML Profile for Integration Flows* contains the patterns which enable complete description of mediation flows. In this profile, there were placed the patterns from the following categories: messaging systems, message routing, message transformation, and messaging endpoints.

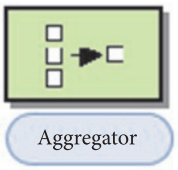
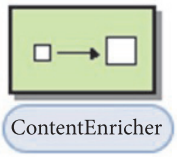
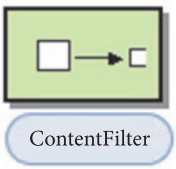
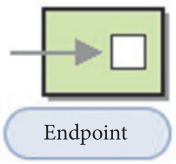
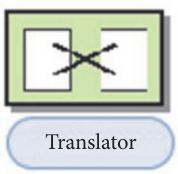
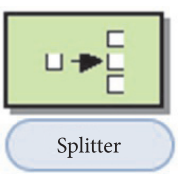
This profile was also created in an IBM Rational Software Architect 8.0. The profile can be applied to projects of UML modeling. For each of mediation mechanisms a unique icon was applied to allow clear identification of mediation mechanism on UML diagram. Those stereotypes are devoted to using on UML activity diagram.

In the profile, there were placed, among others, the following stereotypes:

- Aggregator — combines the results of individual but related messages so that they could be processed as a whole,
- ContentBasedRouter — routing because of the data contained in the message,
- ContentEnricher — enrichment of message content,
- ContentFilter — message content filter,
- DynamicRouter — using a set of rules, messages are sent only to those customers who meet certain conditions,
- Endpoint (Message Endpoint) — point of sending or receiving messages,
- EnvelopeWrapper — wraps the data to be sent in accordance with the requirements of the messaging system,
- MessageFilter — message filtering to prevent these unwanted,
- Resequencer — laying related messages in order,
- Splitter — divides a complex message into several smaller messages which are then separated,
- Translator — transformation of data formats.

Table 3 shows selected stereotypes from the profile with assignment of icons. In the table, there was also presented assignment of UML metaclasses, which are extended by stereotypes.

Table 3
Selected stereotypes from *UML Profile for Integration Flows* with icons and UML metaclasses

Pattern/stereotype name	Pattern/stereotype icon	UML metaclass
Aggregator		Action
ContentEnricher		Action
ContentFilter		Action
Endpoint (Message Endpoint)		InitialNode, ActivityFinalNode
EnvelopeWrapper		Action
Translator		Action
Splitter		Action

A UML activity diagram was extended and its particular form was obtained for mediation flows modeling on integration platform. Thus, a new UML diagram: mediation flows diagram was proposed. It is important that use of existing tools for mediation flows modeling is limited due to the small range of icons. Furthermore, some icons are attached to different mediation mechanisms. Repeated icons may confuse readers of the models. In the proposed profile, each stereotype has its own icon. Thus, a complete transparency of modeled mediation was obtained. In its present form, the profile with a set of mediation flows' stereotypes can be a great help for Integration Architect [16]. In Figure 5, mediation flow for the service "Get current demand for connections" was presented.

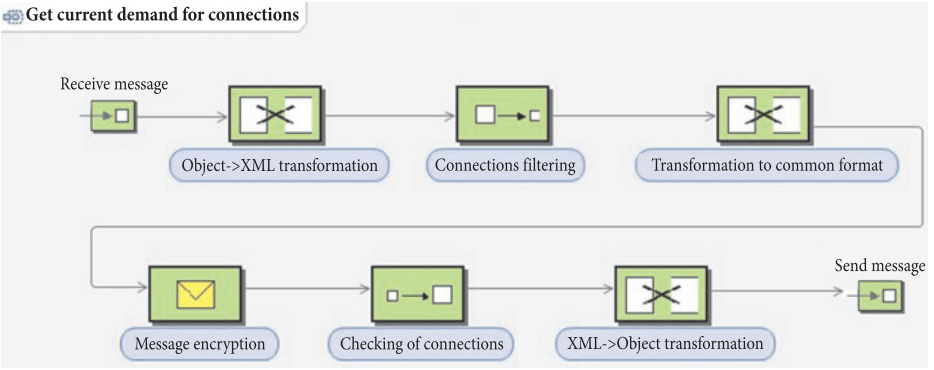


Fig. 5. Mediation flow for the service "Get current demand for connections"

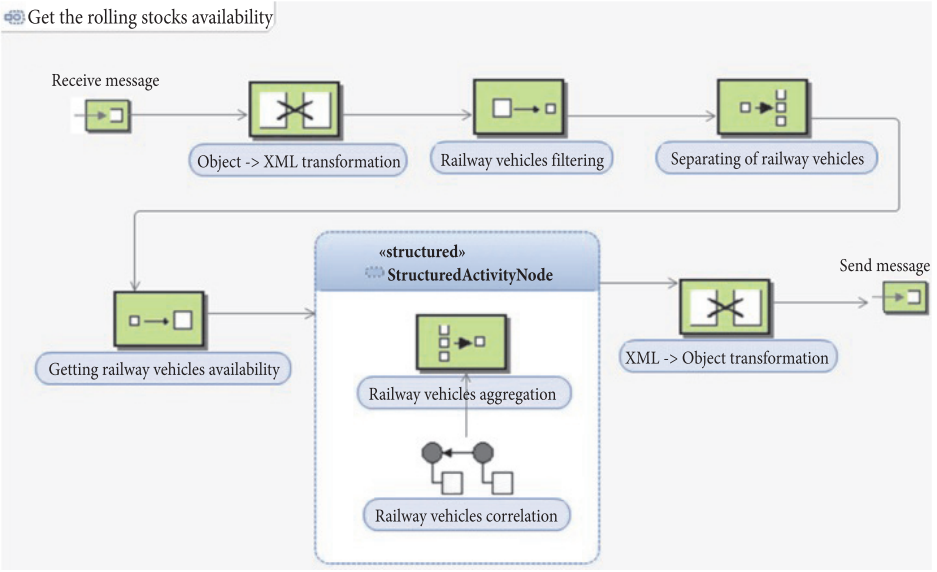


Fig. 6. Composite mediation flow for the service "Get the rolling stock availability"

Thanks to using UML activity diagrams there is also possible to model composite flows. Modeling multilevel UML activity diagram is possible with using StructuredActivityNode UML metaclass. Figure 6 presents composite mediation flow for the service "Get the rolling stock availability".

Moreover, one of the advantages of UML activity diagram is partition. Using partitions it is possible to clearly define responsibilities for performing specific mediation mechanisms in mediation flow.

6. Conclusion and further work

The paper presents two UML profiles: *UML Profile for Integration Platform* and *UML Profile for Integration Flows*. In the first profile, stereotypes relating to the structural elements of the integration platform were placed. The second profile contains stereotypes representing mediation mechanisms which are used in mediation flows. Both profiles were designed and implemented in an IBM Rational Software Architect 8.0. In that fashion, there was obtained environment ready for describing architecture of an integration platform and mediation flows. Both profiles have been designed according to requirements of "1+5" architectural views model [13, 15] which suits the process of an integration platform designing. *UML Profile for Integration Flows* ensures the ability to model complete mediation flows. Moreover, each stereotype in the profile has its own icon which can be easily recognized.

Furthermore, the paper presents a new semantic extension of an activity diagram to model the mediation flows. It is a proposal of a new UML diagram: mediation flows diagram. Thanks to partitions, there can be clearly defined responsibilities for performing activities in a mediation flow. In the paper, as a business case, a railway company was chosen and area of railway transport. Application of stereotypes from both profiles was shown in the paper.

Further work is planned in the area of automation of architectural description of integration platform. This is the field for application of transformations. Both kinds of transformations: model-to-model and model-to-code are considered. The use of transformation is an element of the Model-Driven Engineering approach (MDE). The topic of model-to-model transformations is the subject of many studies [2, 4, 26]. An up-to-date, comprehensive overview of the model transformation verification problems is given in [4]. In recent publications, a quality of models itself is also emphasized [2]. The available analysis of IT systems shows that where model-driven engineering is applied, the software development process duration is recorded to be 3 times shorter [17]. It is planned to analyse impact of transformations on development time of integration solutions. In further studies, both profiles will be used to fully semantically describe architecture of an integration platform.

The article has been prepared on the basis of the paper presented at the Conference Systems Engineering 2012, Warsaw.

REFERENCES

[1] M. Abi-Antoun, J. Aldrich, N. Nahas, B. Schmerl, D. Garlan, *Differencing and merging of architectural views*, Automated Software Engineering, 15, 2008, 35-74.

[2] T. Arendt, G. Taentzer, *A tool environment for quality assurance based on the Eclipse Modeling Framework*, Automated Software Engineering, 20, 2013, 141-184.

[3] Business Process Model and Notation (BPMN) 2.0, OMG 2011, http://www.omg.org/ spec/BPMN/2.0/, 23.04.2013.

[4] D. Calegari, N. Szasz, *Verification of model transformations a survey of the State-of-the-art*, Electronic Notes In Theoretical Computer Science, 292, 2013, 5-25.

[5] M. Chinosi, A. Trombetta, *BPMN: An introduction to the standard*, Computer Standards & Interfaces, 34, 2012, 124-134.

[6] F. Cicirelli, A. Furfaro, L. Nigro, *A service-based architecture for dynamically reconfigurable workflows*, The Journal of Systems and Software, 83, 2010, 1148-1164,

[7] D. Chappell, *Enterprise Service Bus*, O'Reilly, 2004.

[8] J.A. Cruz-Lemus, M. Genero, D. Caivano, S.S. Abrahão, E. Insfrán, J.A. Carsí, *Assessing the influence of stereotypes on the comprehension of UML sequence diagrams: A family of experiments*, Information and Software Technology, 53, 2011, 1391-1403.

[9] *Enterprise Integration Patterns*, http://www.eaipatterns.com/toc.html, 05.05.2013.

[10] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*, Prentice Hall, 2005.

[11] F.S. Esfahani, M.A.A. Murad, M.N.B. Sulaiman, N.I. Udzir, *Adaptable Decentralized Service Oriented Architecture*, The Journal of Systems and Software, 84, 2011, 1591-1617.

[12] R.Z. Frantz, R. Corchuelo, C. Molina-Jiménez, *A proposal to detect errors in Enterprise Application Integration solutions*, The Journal of Systems and Software, 85 2012, 480-497.

[13] T. Górski, *Architectural view model for an integration platform*, Journal of Theoretical and Applied Computer Science, 6, 1, 2012, 25-34.

[14] T. Górski, *Architektura platformy integracyjnej dla elektronicznego obiegu recept*, Roczniki Kolegium Analiz Ekonomicznych, zeszyt 25, Warszawa, 2012.

[15] T. Górski, *Platformy integracyjne. Zagadnienia wybrane*, PWN, Warszawa, 2012.

[16] T. Górski, *Projektowanie platform integracyjnych w architekturze zorientowanej na usługi*, Wiadomości Górnicze, 7.8.2012.

[17] J.F. Groote, A.A.H. Osaiweran, J.H. Wesselius, *Analyzing the effects of formal methods on the development of industrial control software*, IEEE ICSM 2011, USA, 2011, 467-472.

[18] G. Hohpe, B. Woolf, *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging*, Addison-Wesley, 2003.

[19] B. Kitchenham, *Procedures for performing systematic reviews*, Keele University Technical Report TR/SE-0401, UK, 2004.

[20] M. Nečaský, J. Klímek, J. Malý, I. Mlýnková, *Evolution and change management of XML-based systems*, The Journal of Systems and Software, 85, 2012, 683-707.

[21] Object Management Group, *UML Profile for Enterprise Application Integration*, http://www.omg.org/spec/EAI/1.0/, 04.05.2013.

[22] Object Management Group, *UML Profile for Modeling and Analysis of Real-time and Embedded Systems (MARTE)*, http://www.omg.org/spec/MARTE/1.1/, 04.05.2013.

[23] Object Management Group, *Unified Modeling Language Specification Version 2.4.1*, http://www.omg.org/spec/UML/2.4.1/, 04.05.2013.

[24] P. Potena, *Optimization of adaptation plans for a service-oriented architecture with cost, reliability, availability and performance tradeoff*, The Journal of Systems and Software, 86, 2013, 624-648.

[25] N. Rozanski, E. Woods, *Software Systems Architecture. Working with stakeholders using Viewpoints and Perspectives*, Addison Wesley, 2005.

[26] I. Santiago, Á. Jiménez, J.M. Vara, V. De Castro, V.A. Bollati, E. Marcos, *Model-Driven Engineering as a new landscape for traceability management: A systematic literature review*, Information and Software Technology, 54, 2012, 1340-1356.

[27] *Service oriented architecture Modeling Language (SoaML) Specification for the UML Profile and Metamodel for Services (UPMS) v. 1.0*, 2008.

[28] J. Zubcoff, J. Pardillo, J. Trujillo, *A UML profile for the conceptual modelling of data-mining with time-series in data warehouses*, Information and Software Technology, 51, 2009, 977-992.

T. GÓRSKI

Profile UML do opisu architektury platformy integracyjnej

Streszczenie. W artykule przedstawiono nowe profile języka Unified Modeling Language przeznaczone do modelowania architektury platformy integracyjnej. Artykuł zawiera opis notacji i języków modelowania architektury systemów informatycznych, takich jak: BPMN, UML, SoaML. Opisane notacje i języki modelowania nie dostarczają jednak pełnego zbioru konstrukcji semantycznych niezbędnych do przedstawienia architektury platformy integracyjnej. W związku z powyższym zaproponowano zbiór stereotypów języka UML opisujący wymagane dodatkowe konstrukcje znaczeniowe. Stereotypy te zostały pogrupowane w dwa profile języka UML: „UML Profile for Integration Platform" oraz „UML Profile for Integration Flows". W pierwszym profilu umieszczono stereotypy odnoszące się do elementów struktury platformy integracyjnej. W drugim profilu umieszczono stereotypy odpowiadające mechanizmom mediacyjnym. W artykule przedstawiono nowe rozszerzenie semantyczne diagramu aktywności dla modelowania przepływów mediacyjnych. Zaproponowano w ten sposób nowy diagram języka UML: diagram przepływów mediacyjnych.

Słowa kluczowe: integracja, profile UML, architektura system informatycznego, przepływy mediacyjne