An Interactive Composition of UML-AD for the Modelling of Workflow Applications

Yousra BenDaly Hlaoui
Research Unit of Technologies of information and Communication
ESSTT, 5 Av. Taha Hussein, BP :56, Bab Menara
Tunis, Tunisia
Yousra.BendalyHlaoui@esstt.rnu.tn

Leila Jemni Ben Ayed
Research Unit of Technologies of information and Communication
ESSTT, 5 Av. Taha Hussein, BP :56, Bab Menara
Tunis, Tunisia
leila.jemni@fsgt.rnu.tn

ABSTRACT

In today's distributed applications, semi automatic and semantic composition of workflows from Grid services is becoming an important challenge. We focus in this paper on how to model and compose interactively workflow applications from Grid services without considering lower level description of the Grid environment. To reach this objective, we propose a Model-Driven Approach for developing such applications based on semantic and syntactic descriptions of services available on the Grid and abstract description provided by UML activity diagram language as well. As there are particular needs for modeling composed workflows interactively from Grid services, we propose to extend the UML activity diagram notation. These extensions deal with additional information allowing an interactive and semi automatic composition of workflows. In addition this specific domain language contains appropriate data to describe matched Grid services that are useful for the execution of the obtained workflows.

Key Words: Grid services, Interactive composition, Workflow application, UML-AD

1. Introduction

Today’s distributed applications are developed by integrating web or Grid services in a workflow. Due to the very large number of available services and the existence of different possibilities for constructing workflow from matching services, the problem of building such applications is often usually a non trivial task for a developer. This problem requires finding and orchestrating appropriate Grid services in a workflow. Therefore, we propose an approach that allows semi automatic and interactive composition of workflow applications from Grid services. To describe and model workflow applications we use UML activity diagrams.

Recently, several solutions were proposed to compose applications from Grid services such as works presented in [13, 12]. However, the proposed solutions need interaction with user and guidelines or rules in the design of the composed applications. In consequence, the resulting source code is neither re-usable nor it promotes dynamic adaptation facilities as it should. However, for applications composed of Grid services, we need an abstract view not only of the offered services but also of the resulting application. This abstraction allows the reuse of the elaborated application and on the other reduces the complexity of the composed applications. There are several architectural approaches for distributed

computing applications [14] which make easy the development process. However, these approaches need rigorous development methods to promote the reuse of components in future Grid development application. It has been proven from past experience that using structured engineering methods makes easy the development process of any computing system and reduces the complexity when building large Grid application.
To reduce this complexity and allow the reuse of Grid service applications, we adopt a model-driven approach [14]. Thus we introduce in this paper a new approach to build Grid services applications by following OMG(s) principals of the MDA [14] in the development process.
In this approach, our focus is to compose and model workflows from existing Grid services that represents the main aspect in the development of Grid services applications. The workflow modeling identifies the control and data flows from one depicted Grid service's operation to the next to build and compose the whole application. To model and express the composed workflow of Grid services, we use as abstract language the activity diagrams of UML [17]. The provided model forms the Platform Independent Model (PIM) of the proposed MDA approach. This model is more understandable for the user than an XML based workflow description languages like BPEL4WS [21] which represent the Platform Specific Model (PSM).
This paper is organized as follows. Section 2 presents the related work. Section 3 specifies different steps of the interactive composition process and section 4 introduces the relative composition system. Section 5 concludes the paper and proposes areas for further research.

2. Related Work

Many works were carried out in the field of Grid and Web services composition, like works presented in [1, 2, 3, 5, 6, 7]. In [2] authors were interested in the semi automatic composition of web services and proposed a validation approach based on the semantic descriptions of services and on a logic based language to describe and validate the resulting composite Web services. However, the resulting composed web service is not clear for user who is not familiar with logic based languages. In our contribution, we propose a solution not only to compose workflows from available Grid services, but also to provide graphical and comprehensive models of the resulting workflows. In the same framework, Pattak et al [1] proposed a composition approach of Web services based on Symbolic Transition Systems (STS). They developed a sound and complete logical approach for identifying the existence of available composition. They have emphasized upon the abstract representation of the composition request (the goal of the composition) and the representation of the resulting composite Web service. For the representation, authors have used UML state machine diagrams [17] which are suitable only to describe a sequence of component services without addressing the other forms of matching services in a workflow such as parallel branches or and-branches. On the other hand, UML activity diagrams that we use in our modelling approach support all kind of workflow composition patterns [23] such as parallelism, split and fork. The authors in [5, 6, 7] have proposed a Model Driven Approach for composing manually Web services. They were based on UML activity diagrams to describe the composite Web service and on UML class diagrams to describe each available Web Service. The user depicts the suitable Web service and matches it in the workflow representing the composite Web service using UML activity diagrams. This approach would have been better if the composition were automatically elaborated, since the number of available services is in increase with the existence of several forms and manners to compose such services.
In our approach, we propose an UML profile for composing systematically a workflow application from Grid services.

Based on domain ontology description, we lead the user through to the composition process. Also, we provide for this user a graphical interface based on a domain specific UML language for automatic grid service composition. This UML profile [20] is based on stereotypes, tagged values and workflow patterns [20] that we propose to ensure the automatic composition. In the field of Grid services composition the most related work is the work presented by Gubala et al in [3, 12, 13]. In this work, the authors have developed a tool for semi automatic and assisted composition of scientific Grid application workflows. The tool uses domain specific knowledge and employs several levels of workflow abstractness in order to provide a comprehensive representation of the workflow for the user and to lead him in the process of possible solution construction, dynamic refinement and execution. The originality of our contribution is that first we save the effort of the user from the dynamic refinement and execution as we propose a Model Driven Approach which separates the specific model from the independent model.
Second we use UML activity diagrams to deliver the functionality in a more natural way for the human user. The use of UML activity diagrams in the description of workflow application is argued in several works such as works presented in [10, 24, 25, 26, 27]. Thus, the advantage of UML activity diagrams is that they provide an effective visual notation and facilitate the analysis of workflows composition.

3. UML-based interactive composition of workflows from Grid services

In order to integrate different Grid services, we need to analyze constructs of workflows models at higher abstraction level. Since UML [12] is the core of the MDA [8], we use its activity diagram language to model composed workflows. Fig.1 presents the composition process of Grids services workflows. In the following, we illustrate the composition process through the example of the domain of the city traffic pollution analysis. This application, as presented in [19], targets the computation of traffic air pollutant emission in an urban area.

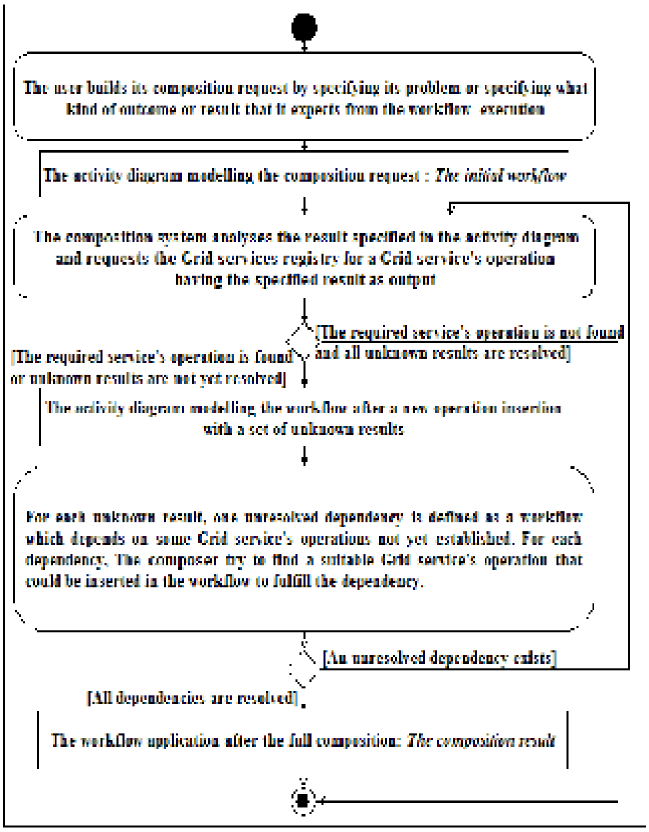

Fig. 1. The composition process of Grid services workflow

Step 1: Fig.2 shows an example of initial workflow that represents a composition request for the results of the pollutant emission due to the city traffic.

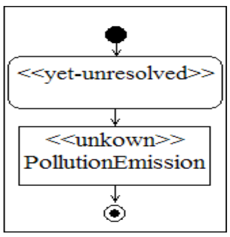

Fig. 2. Initial workflow as a composition request

The desired result is described by the rectangle representing the object node in the relative activity diagram.

Step 2*:** Fig.3 represents the workflow of the computation of traffic air pollution analysis after one step of composition. The service's operation, delivering the **PollutionEmission** result, is ***AirPollutionEmissionCalculator. This operation is the result of the composer query asked to the ontological Grid registry. The operation requires two inputs ***TrafficFlowFile*** and ***PathsLenght-File,*** thus it infers two unresolved dependencies in the activity diagram modelling the composed workflow.

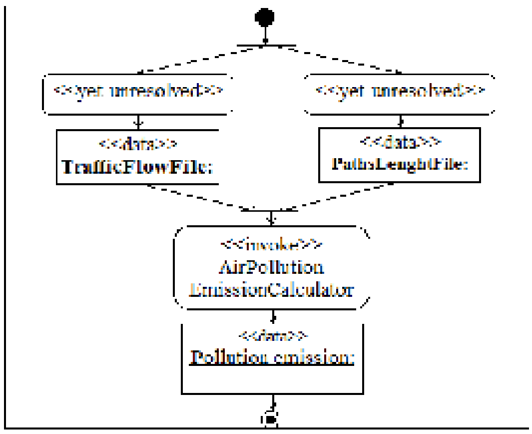

Fig. 3. An example of workflow after one step of composition

Step 3*:** For every dependency that needs to be resolved .i.e. a *yet-unresolved* activity, the composer contacts the ontological registry in order to find suitable service's operations that may produce the required result. The services are described in an ontological form with statements regarding the service operation's inputs, outputs, preconditions and effects (the IOPE set) [6]. Through these notions, the composer system is able to match different operations into a workflow following a reverse traversal approach. Thus, and by associating the required data with the produced output, the composer constructs a data flow between operation using workflow patterns and our UML profile. The composer may also use a specific notion of effect that may bind two operations together with non-data dependency. In [4], five basic control patterns were defined to be supported by all workflow languages and workflow products. These patterns are Sequence pattern, Parallel split pattern, Synchronization pattern, Exclusive Choice and Simple Merge patterns. Fig.4 represents the example of city traffic analysis Workflow after the full composition activity. It involves several Grid service operations, sequence branches, parallel split branches, simple merge branches and a loop. The loop is involved in the workflow diagram as the application iterates in order to analyze the possible traffic. The Figure shows also how UML activity diagrams support the five basic patterns in the composition specific domain of Grid services workflows. In the example, some of object node or input data, such as ***VehiculeType and ***StartZonzId***, are given by the user of the application; they do not have an operation provider. This illustrates the interaction between our composition system and the user.

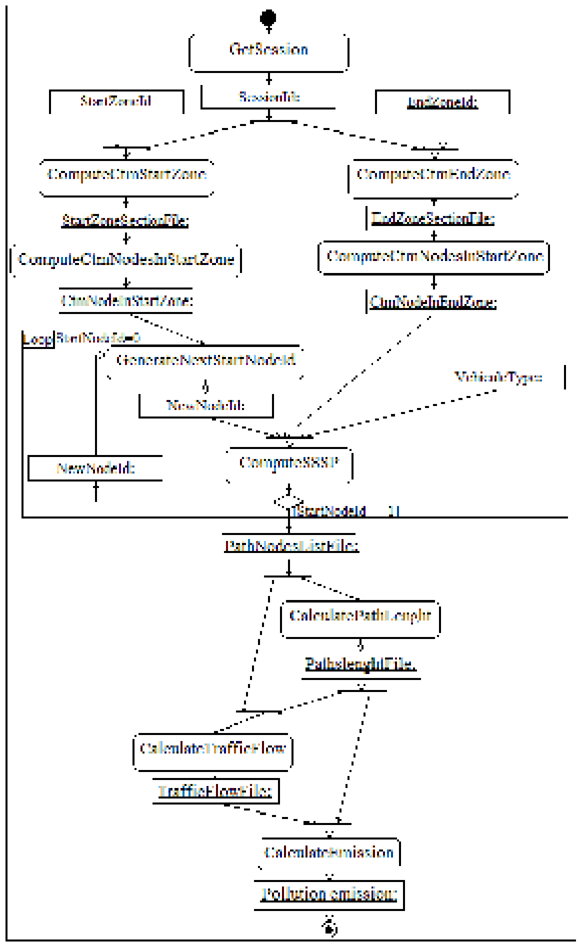

Fig. 4. The workflow application after the full composition

4. The interactive composition system

The system allows a semi automatic and semantic and interactive composition of workflows from Grid services. As shown in Fig.5, It is composed of three components: a Grid Services workflows composer, an ontological Grid Services registry and a workflows execution system also we call it activity machine.

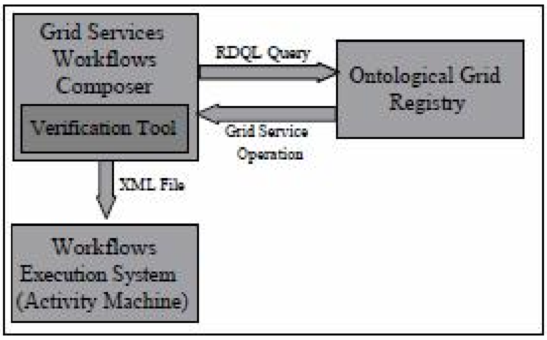

Fig.5: The components of the workflows composition system

4.1 The Grid Services workflows composer

The system provides a graphical interface in the form of UML activity diagrams editor allowing to the user an interactive and systematic workflow composition based on the composition process presented in section 3.

4.2 Verification tool

Once a workflow model is built, it should be validated and verified to ensure its reliability before being executed and reused as sub-workflow.

Consequently, workflow activity diagram models should be spotted and corrected as early as possible for activity diagrams; symbolic model checking has proven to be an efficient verification technique [24]. Thus, our verification tool is based on NuSMV symbolic model checker [31] that supports strong fairness property which is necessary to be verified in a workflow model to obtain realistic results. To support verification of workflow models described in UML activity diagrams, the tool translates the activity diagram into an input format for NuSMV according to a relative semantic. The details of these semantics may not be relevant to the topic for which present paper is submitted. However these details could be made available. With the model checker arbitrary propositional requirements can be checked against the input model. If a requirement fails to hold, an error trace is returned by the model checker. The tool translates systematically the error trace into an activity diagram trace by high-lighting a corresponding path in the given activity diagram.

4.2 Grid registry

The Grid registry is an ontological distributed repository of Grid services and sub-workflows. This registry is responsible for storing and managing documents which contains descriptions of syntax and semantics of services and their operations expressed in a WSDL file [15].

During the workflow composition process, the Grid registry provides the composer system with the description of services available at the moment and provides reasoning capabilities to enable proper matchmaking of services inputs and outputs. The knowledge in the Grid registry can be updated directly by expert users and indirectly through portal by regular users. The registry supports knowledge creation and import, knowledge storage and the knowledge accessing through OWL-S [9].

4.3 Workflow Execution System

Once a workflow model is verified it will be sent to the workflow execution system that produces implementation code for handling control flow and data flow. The activity diagram describing the workflow to execute is translated into a specific XML file which will be the input of the execution system. A workflow execution system executes different workflow activities specified in the workflow XML document in the correct order and with their required inputs and outputs data. The execution of an activity corresponds to the invocation of a grid service operation. The workflow execution system monitors these activities using the tagged values information expressed in the activities but does not perform them. An activity of the activity diagram modelling the workflow represents a state of the workflow execution system in which the system waits for an invoked grid service operation to complete its work. Hence, the defined semantics of activity diagrams for the verification describe the behaviour of the execution system. When the system enters a state relative to an invocation grid service node or activity *ai*, it invokes a piece of behaviour that is executed by the service or system environment. While the latter is in *ai* (activity *ai* is active), it waits for the termination event of the invoked piece of behaviour. When termination event occurs, the system reacts by executing the outgoing edge *E*: it leaves the *E*'s sources and enters the E's targets and the execution process continues for the other activity nodes until the final node is reached.

4. Conclusion

In this paper, we have proposed an approach for composing interactively Grid services workflows [20]. This composition is based on an UML profile for customizing UML activity diagrams to Grid services workflow semantic and interactive composition. The composition process was illustrated through the example of city traffic pollution analysis domain [12] We have developed and implemented the composition tool and we are working actually on the implementation of the workflow execution system that invokes and executes the depicted Grid service instances and manages the control and data flows in a run time environment relatively to our proposed activity diagram semantic. In parallel, we are developing the Grid registry.

References

[1] E. Sirin, J. Hendler, and B. Parsia, “Semi automatic composition of web services using semantic descriptions”. In proceedings of the ICEIS-2003 Workshop on Web Services: Modeling, Architecture and Infrastructure, Angers, France,. April 2003.

[2] J. Rao, P. Kungas, and M. Matskin. “Logic-based web service composition: from service description to process model”. In proceedings of the 2004 IEEE International Conference on Web Services, ICWS 2004, San Diego, California, USA, July 6-9 2004.

[3] T. Gubala, D. Herezlak, M. Bubak, M. Malawski. “Semantic Composition of Scientific Workflows Based on the Petri Nets Formalism”. In Proc. Of the Second IEEE International Conference on e-Science and Grid Computing (e-Science'06). 2006.

[4] W. Li, C. Huang, Q. Chen, H. Bian. “A Model-Driven Aspect Framework for Grid Service Development”. Proc. Of IEEE Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW). 2006, pp 139-146.

[5] R.Gronomo, I.Solheim. “Towards Modelling Web Service Composition in UML”. In The 2nd International Workshop on Web Services: Modelling, Architecture and Infrastructure. Porto, Portugal.2004

[6] R.Gronomo, MC.Jaeger.” Model Driven Semantic Web Service Composition”. In Proc. of the 12th Asia-Paci_c Software Engineering Conference(APSEC'05).2005.

[7] D.Skogan, R.Gronomo, I.Solheim. “Web Service Composition in UML”. In Proc. of the 8th Intl Enterprise Distributed Object Computing Conference (EDOC'04).2004.

[8] M.Smith, T.Friese, , Freisleben, B.: “Model Driven Development of Service-Oriented Grid Applications”. Proc. Of

IEEE Asia-Paci_c Conference on Services Computing(APSCC'06).(2006)

[9] OWL-S:Semantic Markup for Web Services. The OWL Services Coalition. OWL-S version 2.0. http://www.daml.org/services/oml- s /1.0/owl-s.html

[10] Pllana, S., Fahringer, T., Testori, J., Benkner, S., Brandic, I.: "Towards an UML Based Graphical Representation of Grid Workflow Application". In The 2nd Eu-ropean Across Grids Conference, Nicosia, Cyprus, (2004). Springer-Verlag.

[11] Gronomo, R., Solheim, I.: "Towards Modeling Web Service Composition in UML". In The 2nd International Workshop on Web Services: Modeling, Architecture and Infrastructure, Porto, Portugal.(2004)

[12] Guballa, R., Hoheisel, A., First, F.: "Highly Dynamic workflow Orchestration for scientific Applications". CoreGRID Technical Report, Number TR-0101.(2007)

[13] Bubak, M., Guballa, R., Kapalka, M., Malawski, M., Rycerz, K.: "Workflow Composer and service registry for grid applications". Future generation Computer Systems 21, pp 79-86. (2005)

[14] Model Driven Architecture (MDA). Document nomber omrsc/2001-07-01.(2001)

[15] Web Services Descriptio Language (WSDL) 1.1. W3C Note 15 March 2001.

[16] I.Foster, C.Kesselman. "Grid Services for Distributed System Integration". IEEE Computer, vol.35, No.6.(2004)37-46.

[17] Object Management Group. UML 2.0 Superstructure Specification. July (2005).

[18] I.Foster, D.Berry, A.Djaoui, A.Grimshaw, B.Horn, H.Kishimoto, F.Maciel, A.Savy, F.Siebenlist, R.Subramaniam, J.Treadwell, J.Von Reich. "The Open Grid Services Architecture", Version 1.0. 2004.

[19] A.Seaborne. "Rdql-a query language for rdf". Technical report, Hewlett-Packard through W3C Consortium.2004.

[20] Y. Bendaly Hlaoui et L. Jemni Ben Ayed. Patterns for Modeling and Composing Workflows from Grid Services. In. Proceedings of the 11th International Conference on Enterprise Information Systems, ICEIS'2009, Milan, Italy, LNBIP 24, Springer-Verlag Berlin Heidelberg, pp.615-626, May, 2009.

[21] T.Gardner. "UML modelling of automated Business Processes with a Mapping to BPEL4WS". 17th European Conference on Object Oriented Programming (ECOOP). 2003.

[22] M.J.Young. "XML Step by Step". Microsoft Press. ISBN: 2-84082-812-X. 2001.

[23] A.Dumas, A. H. M.Hofstede. "UML Activity Diagrams as a Workflow Specification Language". Presented at UML 2001. 2001.

[24] R. Eshuis, "Semantics and verification of UML Activity Diagrams for Workflows Modelling". PhD thesis, University of Twente. 2002.

[25] R. Eshuis and R. Wieringa. "Comparing Petri net and Activity diagram variants for workflow modelling: A Quest for Reactive Petri Nets, Petri Net technology for communication based Systems", Lecture Notes in Computer Science, Springer Verlag", 2003.

[26] M. Dumas, and A. H. M. ter Hofsetde. "UML Activity Diagrams as a Workflow Speci_cation Language". In UML'2001 Conference, Toronto, Ontario, Canada, Lecture Notes in Computer Science (LNCS), Springer-Verlag, Heidelberg, Germany, October 1-5, 2001.

[27] R. Bastos, D. Dubugras, and A. Ruiz. "Extending UML Activity Diagram for Workflow modelling in Productions Systems", In 35th Annual Hawaii International Conference on System Sciences (HICSS'02), Big Island, Hawaii, IEEE Cs Press, Los Alamitos, CA, IUSA, Januray 07-10, 2002.

[28] C.Goble, D. de Roure. "The grid: an application of the semantic web". ACM SIGMOD Record-Special section on semantic web and data management, 31(4):65-70. 2002.

[29] M. Laclavik, E.Gatial, Z. Balogh, O. Habala, G. Nguyen, L.Hluchy. "Experience Management Based on Text Notes (embet)". In Proc. of eChallenges 2005 Conf.. 2005

[30] M. Masetti, S. Bianchi, G. Viano. Application of K-Wf Grid technology to Coordinated Traffic Management. http://grid02.softeco.it/site/project-info.html

[31] A. Cimatti, E. Clarck, F. G. M. P. M. R. R. S. and Tacchella, A." Nusmv version 2: An opensource tool for symbolic model checking". In CAV'02, International Conference on Computer-Aided Veri_cation. Lecture Notes in Computer Science, Springer Verlag.