2011 International Conference on Computer Science and Information Technology (ICCSIT 2011)
IPCSIT vol. 51 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V51. 113

Executable Petri Nets Model for Semantic Web Services

Xu Yinglei [a], Ma Bingxian[b,*]

[a]School of Computer Science, Shandong University of Finance, Jinan, 250014,China

[b]School of Information Science and Engineering,University of Jinan, Jinan, 250022,China

Abstract. A new approach that using Petri net to model composite semantic web service is put forward oriented to the implementation of semantic web services. Atomic semantic web service is described as a transition element and its input, output, pre-condition and effect as place elements of Petri net, each place element has its place id, name and semantic markup (domain ontology definition) according to parameter's domain ontology. Composite semantic web service Petri net model could be constructed through sharing composition operation of atomic web services Petri nets with ontology reasoning on place semantic markups and its Petri net should allow self-loop, be acyclic and 1-safe.

Keywords: SEMANTIC WEB SERVICE; PETRI net; DOMAIN ONTOLOGY; SEMANTIC MARKUP

1. Introduction

Web services have become more and more important for many companies to survive the massive competition created by online economy through putting their core business on the Internet as a collection of web services for more automation and global applications.

Many works have done for the effective use of web services in recent years, such as Web Service Definition Language (WSDL), Universal Description, Discovery and Integration (UDDI) and Simple Object Access Protocol (SOAP), especially web services compositions and related technologies. Web services composition will assemble web services loosely coupled into a composite web services or services flow to create more flexible dynamic business processes, especially in the areas of dynamic and on-demand business, which will help people to use applications on the web more effectively. The fast and dynamic composition of services is an essential requirement for organizations to adapt their business practices to the dynamic nature of the web, and semantic web service has been put forward to meet with the automatic and dynamic composition of services which has machine understandable description with ontology information.

Semantic web services composition includes services description, discovering, matchmaking, composition and calling of web services. In addition to describe web services with WSDL, OWL-based Web service ontology (OWL-S)[1], the foundation technology within web services composition is how to model web services operate semantic correctly and many formal tools have been selected and used.

In this paper, we set up the executable Petri net model for (semantic) web services according to their executing. There are two classes of web services, atomic and composite web services. Atomic web services has its input and output parameters. Atomic web service is modeled as a Petri net with just one transition element denotes the web service operation and pre and post place elements denote the input, precondition, output and effect. Each input or output of a semantic web service has its ontology definitions and the related place element has its semantic markup. Composite web service combined with more than one web services, its Petri net model could be constructed through sharing composition operation of atomic web services Petri nets based on the ontology reasoning on place semantic markups. And then, we point out that composite web service Petri net should allow self-loop, be acyclic, and 1-safe.

[*] Corresponding author.
E-mail address: ise_mabx@ujn.edu.cn.

The rest of this paper is organized as following: Section 2 briefly introduces semantic web services, OWL-S and Petri nets, section 3 discusses the method to model semantic web services with their executable Petri net models and related properties, section 4 briefly lists some related workings, Finally, Section 5 concludes the paper.

2. OWL-S and Petri Nets

Semantic web services add semantic markup to web services with ontology technology which is a synergistic confluence of the Semantic Web and Web Services. OWL-S uses the OWL language capabilities to represent the service and enable agents to make inferences about it. A semantic web service in OWL-S is described from three different but complementary, perspectives:

Service profile describes what the service does, by including a description of its inputs/outputs and applicability conditions (functional features), and some non-functional features such as quality of service.

Service grounding describes the details about how an agent can access to the service such as the communication protocol, message formats or the port numbers used in contacting the service.

Service model indicates a client how the service works, by detailing the semantic content of requests, the conditions under which particular outcomes will occur, and the step by step processes leading to those outcomes. Service model of OWL-S answers the question how a service is used. The main element of the service model is the process concept. A process is defined by its inputs (the hasInput relation), its outputs (the hasOutput relation), the preconditions that ensures that the process will be performed successfully (the hasPrecondition relation) and its effects in the environment (the hasResult relation). OWL-S defines three kinds of processes: atomic, composite and simple.

Petri net[2] is a formal tool for the description of concurrent and distributed systems. A Petri net is a directed bipartite graph, in which the nodes represent transitions and places elements. The directed arcs describe which places are pre- and/or post-conditions for which transitions (signified by arrows). Petri net has many structural and dynamic properties(such as reachability, liveness, fairness)and analysis methods(such as reachability graph,incidence matrix, state equation) that could be used to describe and analyze system static structure and behaviors properties, for further knowledge of Petri net , reference [2] is recommended .

3. Executable Petri nets Model for Semantic Web Services Composition

To accomplish the automatic composition of semantic web services, firstly, it is necessary to get the correctly operate semantic of them. According to the implementation of web services, each web service has its operations with Application Program Interface (API), User could invoke web service and receive the result of it through SOAP messages, while running composite Web services, each sub web service can be considered autonomous and the user has no control over these services.

Based on its input, output, precondition and effect, A atomic semantic web service is described as a transition element, its input and precondition as pre-place elements, output and effect of it as post place elements of the transition, each place element has its place id，name and semantic markup (domain ontology definition) according to the parameter’s domain ontology definition which might be class or individual and is described with OWL.

The Petri net model of a atomic web service is defined as following:

Definition 1 The Petri net model of a atomic semantic web service is a quadruple $\sum$=(S, T; F, L), where:

1) S is the set of place elements; a place might denote an input, output, precondition or effect parameter of the web service;

2) T is the set of transition elements, an atomic semantic web services has just one element denotes it;

3) $F\subseteq(S\times T)\cup(T\times S)$ is the set of arcs that describe the relationships among place and transition elements;

4) L: $S\rightarrow D\cup\{\tau\}$, D is the set of domain ontology classes and individuals while τ is a null set.

For composite web services which might be services orchestration or services choreography [3], their Petri nets model is defined as following:

Definition 2 The Petri net model of a composite semantic web service is a quadruple ∑=(S, T; F, L), where:

1) S is the set of place elements; each place denotes a input, output, precondition or effect parameter of a web service;

2) T is the set of transition elements; each transition denotes an atomic semantic web services or composite web services which is services orchestration;

3) F and L are same as defined in defintion1.

In definition2, a transition denotes a atomic semantic web services or composite web services which is services orchestration, Because of W3c definition, "orchestrations can be represented as a "white–box service": "service" because it basically defines a public interface. [3]". Users could interact with the "white–box service" instead of sub web services composed within it through the interface, at this point , services orchestration is same as a atomic web services. On the other hand, services choreography describes just the interact protocol of web services within it and users should interact with each sub web service to invoke it.

According to implementation of web services, the Petri net model of composite semantic web service has following propositions.

Proposition 1 :(**Self-loop**) The Petri net model of a composite semantic web service should allows self-loop .

Within a composite web services, two atomic web services might have common input parameters and each parameter has a corresponding place element of the composite web service Petri net model. Token within the place should supports executing of this two atomic web services, and then, self-loop [2] should be allowed between the place and the transition corresponding to each atomic web service.

Proposition 2: (**Acyclic**) The Petri net model of a composite semantic web service must be acyclic.

According to implementation of web services, users should make sure the input and precondition parameters values firstly while invoking a web service. For two atomic web services A and B, suppose one input of services A is *x* and one output is *y*, at the same time, *x* is one of outputs and *y* is one of inputs of service B. Obviously, service A and B would be circular-wait and neither could get correctly result, then, Petri net model of composite web services must be acyclic.

Proposition 3: (**1-safe**) The Petri net system of a composite semantic web service must be 1-safe.

Adding marking to the Petri net model, we can get composite semantic web service Petri net system, it must be 1-safe[2] because of token within each place is according to a parameter of web services.

For reasons of length, we will discuss the executing mechanism of composite semantic web service Petri net system in another paper.

4. Related workings

Many works have studied the modelling and composition of web services. [4] used process algebra as an abstract representation means to describe, compose and reason web services. In [5], DAML-S service descriptions were automatically translated into linear logic axioms and the linear logic theorem proved the possibility to compose the required service from available atomic services. [6]used predicate logic and description logic to describe the state and semantic data type of semantic web services, [7]used Pi calculus to describe the service behaviour and the interaction among services ,but as [8]pointed out that Petri net is more suited to describe web services as alternative for Pi calculus. [9] proposed a Petri net-based algebra, used to model control flows, as a necessary constituent of reliable web service composition process, eight service algebra operators were used to model web service combinations semantic , such as sequence ,choice , iteration, etc. In [10], workflow nets, a class of Petri nets, have been introduced for the representation and verification of workflow processes. [11] translated BPEL to a Petri net semantic. Due to the mapping into Petri nets, several analysis methods are applicable to BPEL processes models, such as the verification of usability of one web service, the verification of compatibility of two Web services, the automatic generation of an abstract process model for a given web service. [12] Proposed labelled Petri net (LPN) models to describe behaviours of each partner cooperative systems (CSs) while a CS is modelled by the combination of all partners' LPN models. Also, we have proposed that adding semantic markups to Petri net to model semantic web services in [13, 14].

5. Conclusion

We have discussed the approach that using Petri net to model composite semantic web service oriented to web services implementation. Within the Petri net, each place element has semantic markup (domain ontology definition) according to the parameter’s domain ontology. Atomic semantic web service Petri net has just one transition element, composite semantic web service Petri net could be constructed through sharing composition operation of atomic web services Petri nets based on the ontology reasoning on place semantic markups, and its related properties are studied . This work proposes a practicable technical approach to carry out automatic semantic web services composition.

6. Acknowledgements

This work is supported by: National Natural Science Foundation of China (60903099), the Excellent Young and Middle-Aged Scientist Award Grant of Shandong Province of China (BS2009DX012), A Project of Shandong Province Higher Educational Science and Technology Program (J09LG14).

7. References

[1] OWL-S.http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/(2011.04)

[2] Murata T. Petri Nets: Properties,analysis and applications. Proc.of the IEEE, 1989, 77(4):541-580.

[3] S ́ebastien Mosser, Mireille Blay–Fornarino, and Michel Riveill, Web services orchestrations evolution:A merge process for behavioral evolution ,2nd European Conference on Software Architecture(ECSA'08), Paphos : Cyprus (2008).

[4] Gwen Sala ün, Lucas Bordeaux, Marco Schaerf, Describing and Reasoning on Web Services using Process Algebra, IJBPIM, 2006,1(2):116--128.

[5] Jinghai Rao, Peep K üngas, Mihhail Mat, Composition of semantic web services using linear logic theorem proving. Information Systems, Vol. 31, No. 4-5. (2006), pp. 340-360.

[6] Zixin Wu, Ajith Ranabahu, Karthik Gomadam, etc. Automatic Composition of Semantic Web Services using Process and Data Mediation.Technical Report, LSDIS lab, University of Georgia, February 28, 2007.

[7] Zhang Chi, "Web Service Interface Extending and Description Based on pi Calculus," iscsct, vol. 1, pp.203-206, 2008 International Symposium on Computer Science and Computational Technology, 2008

[8] W.M.P. van der Aalst,Pi calculus versus Petri nets: Let us eat “humble pie” rather than further inflate the “Pi hype”,BPTrends 3(5), 1-11,2005.

[9] Rachid Hamadi ,Boualem Benatallah, A Petri Net-based Model for Web Service Composition, In Proc. Fourteenth Australasian Database Conference (ADC2003), Adelaide, Australia. CRPIT, **17**. Schewe, K.-D. and Zhou, X., Eds. ACS. 191-200

[10] W.M.P. van der Aalst. “Verification of workflow nets”. In P.Azema and G. Balbo, editors, Proceedings of the 18th International Conference on Applications and Theory in Petri Nets, volume 1248 of Lecture Notes in Computer Science,pages 407-426, Toulouse, June 1997. Springer-Verlag.

[11] Martens. Distributed Business Processes – Modeling and Verification by help of Web Services. PhD thesis, Humboldt-Universit ̈at zu Berlin, July 2003.

[12] YuYue Du, ChangJun Jiang, MengChu Zhou. A Petri Net-Based Model for Verification of Obligations and Accountability in Cooperative Systems, IEEE Systems, Man, and Cybernetics Society ,39(2),299-308,2009.2

[13] Ma Bingxian, Xu Yinglei, Integrating PNML with OWL for Petri Nets, 2009 2nd IEEE International Conference on Computer Science and Information Technology, pp.228-230. 2009.8.

[14] Bingxian Ma, Nengfu Xie. From OWL-S to PNML+OWL for Semantic Web Services. ICCMS 2010, IEEE Computer Society, p326-328. 2010.1