Web Service Aggregation Platform Implementation Based on Join Operation

Jianxiao LIU[1], Da NING[2], Kai XING[1], Zonglin TIAN[1], Panbiao LIU[1], Feng LIU[1,a]

[1]College of Informatics, Huazhong Agricultural University, Wuhan, 430072, China

[2]CSIC NO.722 Research & Development Institute, Wuhan, 430074, China

[a]email: liufeng@mail.hzau.edu.cn

Keywords: Web Service; Join; Clustering; Aggregation

Abstract. In the era of service computing, it needs to develop web service aggregation platform to organize and manage services, and thus to help users to discover the atomic and a set of services with correlations to meet their functional and non-functional requirements. This is an important research point in the service-oriented software engineering. This paper develops a web service aggregation platform based on the join operation in relational database (RDB). Firstly, it uses the self-join operation in RDB to cluster service to form different service clusters. Then it organizes different service clusters in further according to the service execution dependency relationship, and thus to realize service aggregation. Finally, the service selection function is implemented to meet users' personal requirements.

Introduction

In service computing environment, the emergence of a large number of Web services on the internet presents new challenges to service discovery. How to discover services quickly and efficiently to meet users' needs is a key problem to be solved. In order to solve this problem, it needs to realize service organization and management efficiently, and select services for users' personal requirements accurately and efficiently. It can help to enhance the quality of on-demand service.

Service clustering is a common service organization method. There are all kinds of services which realize different functions, and users often need a set of services with correlations. Therefore, it needs to organize the clustered services furtherly based on service clustering. Then users can find the correlated services quickly and accurately to meet their needs. At present, there are some research work about Web service aggregation and organization. But the Web service aggregation platform is relatively lack of research, and there is no authoritative Web service aggregation platform to be used.

This paper designs and implements a Web service aggregation platform based on the join operation in relational database. The platform mainly includes the following three function modules: Web service clustering, service clusters organization and Web service selection. Based on the relational database theory and ontology concept semantic relationships, it mainly aggregates and selects Web services from the aspect of interface and capability. Finally, some screenshots of the developed platform are elaborated.

The Web service aggregation platform architecture will be introduced in section 2. The related work will be described in section 3. We will elaborate the screenshot of the developed platform. The conclusion and next step work are given finally.

Platform Architecture and Modules

Platform architecture

The platform main includes the following function modules: database development, Web service clustering, service cluster organization, personal service selection. The architecture is shown in figure 1.

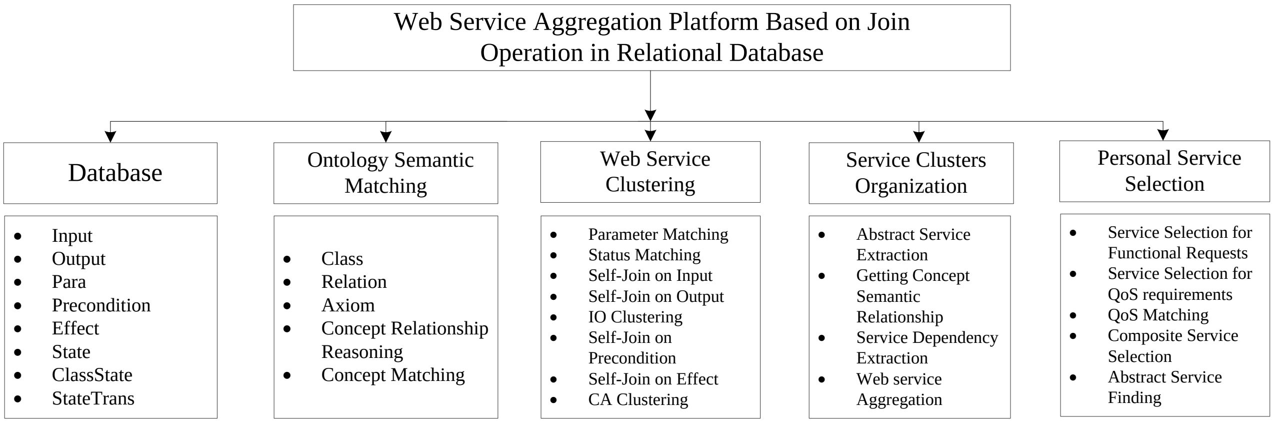

Web Service Aggregation Platform Based on Join Operation in Relational Database
Database
Input
Output
Para
Precondition
Effect
State
ClassState
StateTrans
Ontology Semantic Matching
Class
Relation
Axiom
Concept Relationship Reasoning
Concept Matching
Web Service Clustering
Parameter Matching
Status Matching
Self-Join on Input
Self-Join on Output
IO Clustering
Self-Join on Precondition
Self-Join on Effect
CA Clustering
Service Clusters Organization
Abstract Service Extraction
Getting Concept Semantic Relationship
Service Dependency Extraction
Web service Aggregation
Personal Service Selection
Service Selection for Functional Requests
Service Selection for QoS requirements
QoS Matching
Composite Service Selection
Abstract Service Finding

Fig. 1. Platform Architecture

Platform modules

(1) Database development

We design the following tables to store service basic information: parameter table of *Para*, service execution prerequisite table of *Precondition*, service execution postcondition table of *Effect*, service nonfunctional information table of *QoS*, status information table of *State*, the class status information table of *ClassState*, status transformation table of *StateTrans*, etc. We store the service basic information into the above tables, and it can lay the foundation of service aggregation and organization.

(2) Ontology semantic matching

In order to express Web service *IOPE* information from the semantic level, the ontology information is also stored. And we correlate it with the *IOPE* properties of services. We design the following tables: ontology concept table of *Class*, concept relationship type table of *Relation*, concept relationship table of *Axiom*. The interface and capability of Web services will be expressed by the ontology concepts and the concept status. We mainly consider the following semantic relationships: Exact/Plugin/Subsume/Intersect/Fail [1]. They refer to the relations of equivelantOf, subClassOf, superClassOf, intersection and fail respectively. On the basis of the stored ontology information, we realize the function of concept relationship reasoning, concept matching, etc.

(3) Web service clustering

Based on the concept semantic reasoning relationships and concept status path, the service matching value is calculated from the aspects of Web service interface (Input and Output) and capability (Precondition and Effect) using the self-join operation in RDB, and thus to realize service clustering. This approach can help to enhance the service clustering efficiency and accuracy. Firstly, it does the self-join operation on the table of *Input* and *Output* separately. The input and output matching matrix can be got, and the matching matrix of the interface level can be calculated. The services will be clustered according to the service matching values. Similarly, the services are clustered from the capability level. Finally, it combines the matching result from the above two levels and gets different service clusters. It can lay the foundation of service aggregation and organization.

(4) Web service clusters organization

This module mainly realizes the following functions: abstract service extraction, getting abstract service execution relationships and service aggregation based on join operation.

Abstract service extraction: it uses abstract service extraction algorithm to extract abstract services in different service clusters from the aspects of *Input*, *Output*, *Precondition* and *Effect*. Abstract service execution dependency relationship extraction: the abstract service execution relationships are got from aspects of service interface and capability. Service aggregation based on join operation: it uses the join operation in RDB to organize different service clusters according to the service execution dependency relationships, and thus to realize service aggregation.

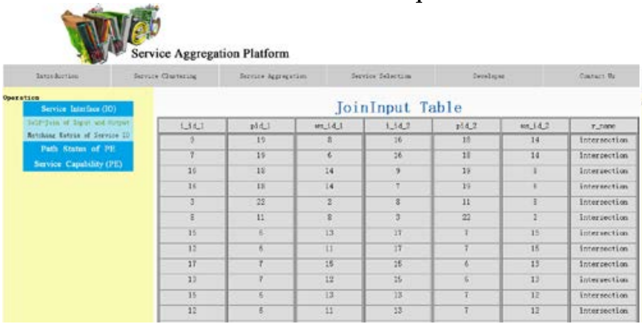
(5) Personal service selection

On the basis of service aggregation, it selects abstract service type from aspects of interface and capability, and thus to construct service execution path. Then it selects a set of services with correlations to meet users' functional requirements. According to users' QoS requirements, it tailors all the service execution paths and gets the optimized service execution path. Then it selects services with proper QoS in further in different service clusters. The atomic service and a set of services with correlations can be selected to meet users' functional and non-functional requirements.

Platform Implementation

On the basis of designing the platform, we use the struts2 architecture and MVC pattern to develop the Web service aggregation platform based on the join operation in RDB. It firstly analyzes and stores the information of Web service, and uses the pellet reasoning engine to analyze ontology files. The concepts and semantic relationship between concepts in ontology are stored. The modules of service clustering, aggregation and selection are developed.

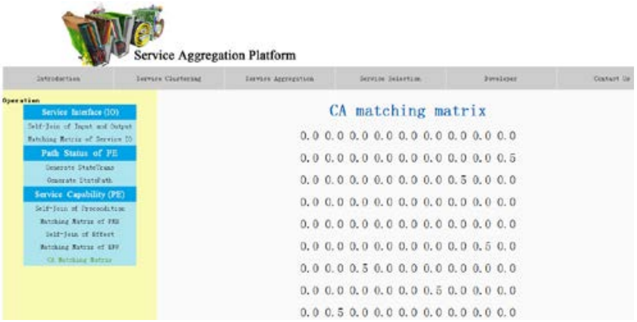
The platform includes the modules of system description, service clustering, service aggregation and service selection. Figure 2~5 show the screenshot of the developed modules. Figure 2 shows the self-join table of Input. Figure 3 shows the service execution capability matching matrix. Figure 4 shows the abstract service execution path of service aggregation. Figure 5 shows the service selection for users' functional requirements.



JoinInput Table

i_id_1	p_id_1	ws_id_1	i_id_2	p_id_2	ws_id_2	r_name
9	19	8	16	18	14	Intersection
7	19	6	16	18	14	Intersection
10	18	14	9	19	8	Intersection
16	18	14	7	19	6	Intersection
3	22	2	8	11	8	Intersection
8	11	8	3	22	2	Intersection
15	6	13	17	7	15	Intersection
12	6	11	17	7	15	Intersection
17	7	15	15	6	13	Intersection
13	7	12	15	6	13	Intersection
15	6	13	13	7	12	Intersection
12	6	11	13	7	12	Intersection

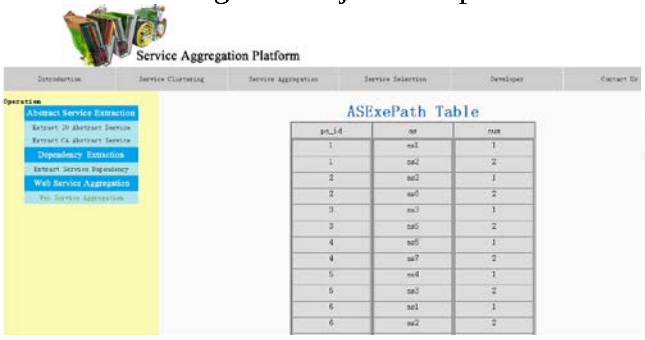
Fig. 2. Self-join of Input



CA matching matrix

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5
0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0
0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0
0.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0

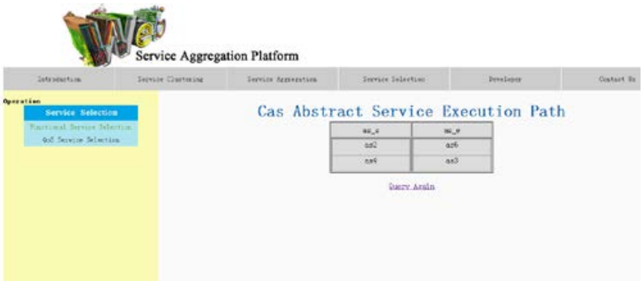
Fig. 3. Matching Matrix



ASExePath Table

pe_id	as	num
1	as1	1
1	as2	2
2	as2	1
2	as6	2
3	as3	1
3	as5	2
4	as6	1
4	as7	2
5	as4	1
5	as3	2
6	as1	1
6	as2	2

Fig. 4. Execution Path of Service Aggregation



Cas Abstract Service Execution Path

as_s	as_e
as2	as6
as4	as3

Fig. 5. Web Service Selection

Related Work

At present, there are some research work about Web service clustering, aggregation and selection. We mainly discuss the research methods and related platforms about service aggregation. Hu et al. have proposed a user-oriented service workflow constructing method [2]. The services are clustered and the spanning tree approach is used to represent the services in the same cluster firstly. Then service clusters are organized through the workflow business logic method. Liu et al. have organized Web services using the method of service group and service node [3]. Service group is similar to the service clusters that are formed through clustering, and service node is similar to the abstract service of specific service cluster in our method. The services are organized through building service aggregation model which uses business logic integration of service nodes. In [4], we have proposed a Web service aggregation method in the orientation of semantic interoperability. The service clusters are mainly organized from aspects of service execution capability and information interaction between services. And we also have organized service clusters in the consideration of users' features, like users' roles and goals [5]. Sellami et al. have used community to organize and manage Web services [6][7]. The fuzzy clustering algorithm is used to cluster services to form service community, and the service communities are organized from the point of functionality. Ding et al. have proposed a user-centric service composition method from synthesizing multiple views [8]. This approach starts from users' needs and it realizes service composition process in the exploratory manner. Han et al. have proposed a "starting form business and IT perspectives and meeting-in-the-middle" approach to realize service organization [9]. The users' requests are directly mapped into the virtual service resources and it realizes service discovery from the aspect of input and QoS constraints information. Chu et al. have used a personalized requirement oriented virtual service resource aggregation method [10]. This method uses the virtual resource with large-granularity to satisfy the personal requests. Then it models services to be autonomic entities which have certain intention and behaviors. And services can aggregate autonomously to meet users' requests. He et al. have proposed a requirements semantic-driven aggregation method for on-demand service [11]. They use the stakeholders-driven requirements semantics acquiring technique and use the connecting ontologies to realize service aggregation and organization.

While some of the above research approaches do not realize service organization from the semantic level, and this will influence the accuracy of service organization. Some approaches only consider the business logic execution relationship between services, but the service interface, capability and QoS information are lacking of consideration. Based on clustering services and extracting abstract services, this paper uses the join operation to determine the service execution dependency relations quickly in the view of service interface and capability. We have developed the platform to realize Web services aggregation and organization, and it can lay the foundation of realizing more personal service selection.

Conclusion

In the era of service-oriented software engineering, it needs to develop Web service aggregation platform to realize Web service organization and management, and help to select the atomic service and a set of services with correlations to meet users' functional and QoS requirements. Based on storing Web services and ontology information, this paper realizes the following functions: doing the semantic matching calculation from the interface and capability level; clustering Web services; extracting the abstract services based on service clustering; extracting the services execution dependency relationship; using the join operation to organize service clusters and realize service aggregation; realizing service selection for users' functional and QoS requirements. The next step of research work includes the following aspects: improving and optimizing the platform; considering users' features to realize service aggregation and selection in further and thus to meet the more personal requirements.

Acknowledgement

This research is supported by the National Training Programs of Innovation and Entrepreneurship for Undergraduates under grant No.201410504064, the Fundamental Research Funds for the Central Universities under grant No.2013QC020, 2013PY118.

References

[1] L. Li, I. Horrocks, A software framework for matchmaking based on semantic Web technology. Proc. of the 12th Int'l Conf. on World Wide Web. New York: ACM (2003) 331-339.

[2] C. H. Hu, M. Wu, G. P. Liu GP, et al, An Approach to Constructing Web Service Workflow Based on Business Spanning Graph, Chinese Journal of Software. 18(8) (2007) 1870-1882.

[3] S. L. Liu, Y. X. Liu, F. Zhang, A Dynamic Web Services Selection Algorithm with QoS Global Optimal in Web Services Composition, Chinese Journal of Software. 18(3) (2007) 646-656.

[4] J. X. Liu, K. Q. He, J. Wang, Z.W. Feng, D. Ning, A Semantic Interoperability Oriented Method of Service Aggregation, Chinese Journal of Software. 22(Suppl 2) (2011) 27-40.

[5] J. X. Liu, K. Q. He, J. Wang, D. H. Yu, Z. W. Feng, D. Ning, X. W. Zhang, An approach of RGPS-guided on-demand service organization and recommendatio, Chinese Journal of Computers. 36(2) (2013) 238-251.

[6] M. Sellami, W. Gaaloul, S. Tata, An Implicit Approach for Building Communities of Web Service Registries, The 13th International Conference on Information Integration and Web-Based Applications and Services. Ho Chi Minh City, Vietnam, December 5-7 (2011).

[7] M. Sellami, O. Bouchaala, W. Gaaloul, S. Tata, Communities of Web Service Registries: Construction and Management, Journal of Systems and Software. 86 (2013) 835-853.

[8] W. L. Ding, J. Wang, S. Zhao, A user-centric service composition method synthesizing multiple views, Chinese Journal of Computers. 34(1) (2011) 131-142.

[9] Z. F. Zhao, Y. B. Han, J. Yu, et al, A Service virtualization mechanism for business user programming, Chinese Journal of Computer Research and Development. 41(12) (2004) 2224-2230.

[10] D. H. Chu, X. Z. Wang, Z. J. Wang, X. F. Xu, Personalized requirement oriented virtual service resource aggregation method, Chinese Journal of Computers. 34(12) (2012) 2070-2380.

[11] B. Wen, K. Q. He, P. Liang, J. Wang, Requirements semantics-driven aggregated production for on-demand service, Chinese Journal of Computers. 33(11) (2010) 2163-2176.