A Model for Composite Service Discovery Based on Data dependency

CHEN Zhi-gang LIU Li, SHEN Xiao-jian
School of Information Science and Engineering,
Central South University, Changsha, China 410083
E-MAIL: llyan_liu@163.com

LI Chang-Yun
School of Computer and Communication,
Hunan University of Technology,
Zhuzhou, China 412008

Abstract: Service discovery is the premise of service composition. The existing service discovery methods only consider individual service functions and static properties, less considering the inherent dependencies between the composite services. The data dependencies between services reveal the logic correlation between the composite services, but it is of great significance for the service composition and service discovery. Propose a domain ontology services dependency graph (DOSDG) model of composite service discovery based on the data dependence relations. This model consists of two layers composed of the lower from the domain services dependency graph (DSDG) and the higher domain ontology agent (DOA). Each DOA manages a DSDG, and all of the DOA forms a P2P structure. Reasoning the service domain context and the correlation between the services from user's composite request can improve the efficiency in runtime. This approach takes into account the service domain features and the correlation between services, so it not only greatly reduces the services search space and can amend the user's request.

Key words: composite service discovery; data dependency; composite request, domain service ontology

1 Introduction

With the proliferation of Web services, Web services composition become a research hotspot. Services composition has two main processes: Service composition process control and the binding of atomic services, the former problem to be solved is how to find and select the composite service which the user request and the order of service composition. Service composition mainly uses two class method implementation: a workflow model is that the establishment of service composition process, and then an alternative way to process The service template in turn replaced with specific services, such as BPEL [1]. The workflow solution may be mature but require manual labor involved, resulting in services composition degree of automation and flexibility is inadequate; another method is based on semantics as the core [2-4], is the representative METEOR-S [5] the framework proposed in WSDL-S, OWL-S. Because this method is based on all registered services as the search space resulting in a lower search efficiency and a smaller number of network environments. For semantic-based Web services focused on two elements: one is to create a computer language could be more mutual understanding between the computers, The second is based on the language in which Web services can be achieved between the automatic discovery, selection, integration and interaction model or framework structure [6-7]. Most of the existing composite service discovery methods think that the optional services are mutually independent, without considering the linkages between services. In practice, service-oriented applications, a follow-up service discovery often depend on other associated optional service, there is often
among the variety of associated [8], such as data association, time sequence relationship. A self-help tourism going to a scenic area need to know the appropriate air routes, scenic hotel reservations, as well as room and board situation, according to a user's queries need to return to user an appropriate mix of travel services, which are closely related between the services rather than independent of each other. In short, the main problems of the existing composite service discovery methods is: searching in all of the registered service as a search space resulting in relatively low efficiency, no response correlation between the composite services, not fully utilize the service requests and the existing Web Services Description Language can not describe the correlation messages between services.

Service dependency graph (SDG) is a directed graph, which reflects the input and output data dependencies correlation of the registered services. To construct SDG is based on the relationship between input and output interfaces of the WSDL-defined according to the different domain. This process is automatically implemented by the intelligent agent when the service registry. In the SDG construction process, two tables are used to store the dependencies relationship between services. One, called operation-data table, used to record input and output data required to service operation. The other, called data-operation table, used to record the consumption of these data operations. Propose a two layered composite services discovery model DOSDG based on domain ontology SDG. This model consists of two layers composed of the lower from the domain services dependency graph (DSDG) and the higher domain ontology agent (DOA).Each DOA manages a DSDG, and all of the DOA forms a P2P structure. To find composite services has mainly two steps: the first step is to reason logically the user service domain context and the correlation between the services from user’s composite request at run time and locate the query in several limited domain ontology agent, the second is to find the specific services using the heuristic search within the domain ontology SDG based on the dependency relationship between the services. This approach takes into account the service domain features and the correlation between services, so it not only greatly reduces the services search space and can amend the user's request. In this way the composite service found has a strong correlation in the business logic and its output play a very good assistance and guidance in the application logic.

The article is structured: Section II describes system model DOSDG and the basic concepts and definitions. Section III describes the construction algorithm of domain ontology SDG, Section IV describes the composite service discovery algorithm based on DOSDG model, Section V presents related work, the Section VI is the conclusions and prospects.

II System Model

A. Concepts and Definitions

Definition 1. A services $S = (OP, I, O, P, E, D)$. Where $S.I = \bigcup Q.I$, and $S.O = \bigcup Q.O$ is the operation set and input /output data set of $S$ respectively sets; $S.OP = (Q | Q$ is the S operation), $P = \square S.P$, and $E = \square S.E$ is the pre-conditions and produce of $S$, $D$ is the domain of $S$.

Definition 2. User's composite service request $CSR = (S, I_r, O_r)$. Where: $I_r = \bigcup S.P.I$, and $O_r = \bigcup S.P.O$ that CSR provides a collection of input and output requirements, $P$ is the operation set of composite service.

Definition 3. For service $s_i$ and $s_j$, If $\square O_i \supseteq I_j$, $\square E_i \supseteq P_j$ ,we say $s_i \rightarrow s_j$ if the $s_i$ and $s_j$ satisfy ①and ②. Just to satisfy ①,there is
s_i \rightarrow s_j$. Define the data dependence degree (SDDD) between the two services as the following equation

$$SDDD(S, S) = \alpha C_e(O, I) + (1 - \alpha) C_e(P, E)$$  \hspace{1cm} (1)

$$C_{\cdot \cdot} = \begin{cases} 1 & \text{if } i = j \\ \frac{|\rho \cap \eta|}{|\eta|} & \text{other} \end{cases}$$  \hspace{1cm} (2)

$0 \leq \alpha \leq 1$ is the weights. The definition $C_{\text{on}}(E, P)$ is same as $C_{\text{on}}(O, I)$.

Definition 4. Define the domain service relationship ontology, DSRO = (T, S, R_T, R_S), where T is the subject of S, S is the available services; R_T is the relationship between T and S; R_S \in (T \times T, S \times S, T \times S). Rs indicates the data dependencies between services.

Definition 4 Composite Service Satisfaction, CSSAT Describe the extent of the query results to satisfy user requests CSR. CSSAT defined as follows:

$$CSSAT = \prod_{i=2}^{n} \prod_{j=2}^{n} SDDD(S_i, S_j, S_{i+1}, S_{j+1}, R)$$  \hspace{1cm} (3)

CSSAT reflects the transmission of data dependency between composite services.

B. Data Dependency Description

The data dependency relation should be defined in the service description, but the existing service description frameworks have been no effective way to define such data relation. We use the following template to describe this relationship in the service WSDL file. $S = (O, I, O, P, E, D)$ descript as follows:

$\text{<definitions>}$
$\text{<message name=\"S_input\">}$
$\text{<part name=\"input\" element=\"T\">}$
$\text{</message>}$
$\text{<message name=\"S_output\">}$
$\text{</message>}$
$\text{<message name=\"S_input\">}$
$\text{<part name=\"input\" element=\"O\">}$
$\text{</message>}$
$\text{<message name=\"S_output\">}$
$\text{</message>}$
$\text{<message name=\"S_pre\">}$
$\text{<part name=\"pre\" element=\"P\">}$
$\text{</message>}$
$\text{<message name=\"S_post\">}$
$\text{<part name=\"post\" element=\"E\">}$
$\text{</message>}$
$\text{<message name=\"S_domain\">}$
$\text{<part name=\"dom\" element=\"D\">}$
$\text{</message>}$
$\text{<message name=\"S_input\">}$
$\text{<input message=\"S_INPUT\">}$
$\text{</inputmessage>}$
$\text{<output message=\"S_OUTPUT\">}$
$\text{</outputmessage>}$
$\text{<precondition message=\"S_pre\">}$
$\text{</preconditionmessage>}$
$\text{<postcondition message=\"S_post\">}$
$\text{</postconditionmessage>}$
$\text{<domain message=\"S_dom\">}$
$\text{</domainmessage>}$
$\text{<operation>}$
$\text{</operation>}$
$\text{</portType>}$
$\text{</definitions>}$

C. The Model DOSDG

We use the domain service ontology as the shared semantic model of domain service community. Those services with data dependency who share the same ontology form a domain ontology service dependency graph (DOSDG). In order to response the cross-cutting services request, each DOSDG should assign a domain ontology agent (DOA) to manage. The services within DOSDG follow a unified ontology model and the services can communicate through DOA. All of the DOSDG forms a P2P structure. DOSDG model frame is shown in Figure 1.
When the user sent to composite services request to DOSDG, the user's scenario and data dependency relation between services can be identified by reason engine from user’s CSR. Query results return to the user, and the user can confirm based on the satisfaction criteria (see equation (3)) or modify the query input and output conditions.

### D. DOSDG Construction

The basic idea of construction DOSDG is find their Predecessor and successor for each new adding registration services in accordance with the equation (1) and domain using intelligent agents. Two data tables are used to indicate the data dependencies between services. DOSDG can constructed depend on data dependence in service ontology domain. Construction DOSDG algorithm is as follows:

**input:** $S=s_1, s_2, ..., s_n$, // $s_i$ $(1 \leq i \leq n)$ Is a registered service

**output:** DOSDG

1: Initialization

   \[ \text{DOSDG} = \varnothing, K = |S|, n = 0 /\text{. n is the number of field} \]

2: while $i, j < k$

   | If $s_i.d = s_j.d$ and \( \text{SDDD}(s_i, s_j) > M \); // Find the same service area of two to determine the degree of dependence between the data SDDD () , $M$ is a given threshold value
   | CreateNode \((s_i, s_j)\);
   | \(s_i \rightarrow s_j\), or \(s_j \rightarrow s_i\);

   | $i++$, $j++$;

   | end while

   \[ \text{DOSDG}.D = D_n \] // To add a domain ontology agent DOSDG

3: for $i = 0, i < N, i++$ \(//N\) is the number of domain

   Connectp2p(Di); // The formation of the domain agent P2P architecture

Endfor

4: return DOSDG

### E. User Request Pattern Recognition

We can understand the user behavior and data correlation between services by analyzing the user composite service requests through machine learning techniques from amount of user records. Understanding of patterns of user requests, you can not only find relationship between the requested service and a subject which user will search, but the domain the services belong to and the dependency between requested services. All of the information in discovering composite service will be inspired to take an important the role.

### □ Composite Service Discovery Based on DOSDG (CSD\_DOSDG)

Composite service discovery algorithm CSD\_DOSDG described as follows:

**Input:** CSR(S, D\_in, D\_out) // $S$ : service set , $D_{in}$, $D_{out}$ data set of input/output.

**Output:** CST // composite service in DOSDG satisfy the CSR.

CSD\_DOSDG(CSR, DOSDG). // find composite service
template in DOsdg
1. GetInfo(CSR) // Analysis and reasoning CSR to get information such as domain and context
2. Locate S.D in {DOA1, DOA 2,…, DOA m} // Determine the domain where the CSR
3. CST=ϕ ;
for (i,j=0;i,j<=k;i++,j++)
   If DSDG.Si.Din = CSR.Si.Din
   Then HeurSearch_ DSDG() // Heuristic Search DSGi depend on input and data dependency between services
      CST=CST+ DSDG.Si
   If CSR.Sj.D in DSDGj
      Then HeurSearch_ DSGj() // Heuristic Search DSGj depend on data dependency
      CST=CST+ DSGj.Sj
4. if DSDG.Si = CSR.S.Dout then return
5. else
   HeurSearch (DOA, {DSDG1, DSDG2,…, DSDGk})g //CSR in more than one domain
6. return CST;

Fig.3 algorithm CSD_DOsdg

The algorithm has the following characteristics.

Algorithm can determine the domain which user's concerned by analyzing the dynamic semantics of user's composite services dynamic semantics, filter out those domain not related to user request resulting in reducing the search space.

Hierarchical processing the registration service, it improves the efficiency of the service location.

The use of services dynamic semantics reflects the content the user concerned with recommendation relationship between the services.

Related Work

Composite service is composite of simple services, or synthesis of services. A few literatures related to service composition uses relationship between the services. Literature [4] proposed a hierarchical model supporting the selection of composite services from the quality of service (QoS) correlation between the available services, but its disadvantages are QoS itself of the services in dynamic environment is difficult to measure. Literature [5] use AND / OR graph to find the composite services, describing the dependencies between services have two types of nodes: operation nodes and data nodes, the data dependency of the services considered, but the disadvantage is not used the context of user composite services request. All of the registered services are adopted as the search space resulted in inefficiencies. Literature [10]Model the data correlation and put forward the concept of explicit data-links. The data-links used by static declarations directly expressed in the service description. However, the process does not reflect how to use the services
correlation. Literature [12] put forward the idea of organization the Registration services by domain. During registration the service are classified according to their domain. The method using P2P to search the services can improve the efficiency of the matching services, but it did not use the services domain characteristics to guide to match user composite query. Literature [13] proposed a generation method of interactive services based on service relationship ontology. From three point of view analysis the relationship of service and building the service relationship ontology. In the service generation, get the user intent through mining the dynamic semantics of the services and combined with service relationship ontology to guide follow-up services to match. [14] presents a double layer P2P semantic service discovery model based on ontology community, within the community, different registration center to form a layer P2P architecture, and inter-communal body and then form another P2P structure.

VI. Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 60873082,60903058,60773110. China Postdoctoral Science Foundation under Grant No. 20080440216, Doctoral Education Fund for New Teachers Grant No. 200805331019.

References