A Semantic-Link-Based Infrastructure for Web Service Discovery in P2P Networks

Jie Liu[1, 2] and Hai Zhuge[1]
China Knowledge Grid Research Group, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, 100080, China[1]
Graduate School of the Chinese Academy of Sciences[2]
(lj@kg.ict.ac.cn, zhuge@ict.ac.cn)

ABSTRACT
An important issue arising from P2P applications is how to accurately and efficiently retrieve the required Web services from large-scale repositories. This paper resolves this issue by organizing services in the overlay combining the *Semantic Service Link Network* and the *Chord* P2P network. A service request will first be routed in the Chord according to the given service operation names and keywords. Then, the same request will be routed in the Semantic Link Network according to the service link type and semantic matching. Compared with previous P2P service discovery approaches, the proposed approach has two advantages: (1) produce more accurate and meaning results when searching for particular services in a P2P network; and (2) enable users and peers to discover services in a more flexible way.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – r*etrieval models*, s*earch process*, *selection process*.

General Terms: Algorithms, Management, Design

Keywords: Peer-to-Peer, Semantic Link, Web Service

1. INTRODUCTION
An important issue arising in P2P applications is how to accurately and efficiently discover the required services in P2P networks. Based on our previous work on multi-valued service specialization [5], this paper proposes a semantic-link-based service discovery approach by incorporating the Chord overlay.

2. GENERAL ARCHITECTURE
When a peer P_i joins a P2P network, it will publish its services in two ways: (1) associate each *Operation Name* and *Keyword* of *Service* (P_i) with a key using SHA-1 hashing functions and store index of *Service* (P_i) at the peer corresponding to that key on Chord; (2) establish semantic service links between itself and its neighbors.

As depicted in Figure 1, users can query services using GUI (Graphical User Interface) or SSQL (Semantic Service Query Language) — an SQL-like language for service discovery. Peers can communicate with their neighbors using SOAP messages.

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

Upon receiving a service request, P_i will first hash the given operation names or keywords using SHA-1 functions and search on the Chord overlay. In addition to the exact keyword-based lookup protocol, our approach supports two types of complex queries, i.e., *Wildcards Matching* and *Combinational Query*.

The keyword-based Chord lookup has no semantics [4]. Similar services that have slightly differences in operation names or keywords descriptions may be hashed to different peers on Chord.

To support semantic-based service discovery and get more accurate results, after routing in Chord, the same request will be forwarded to neighbors of P_i according to the service link types and the similarity degree between them.

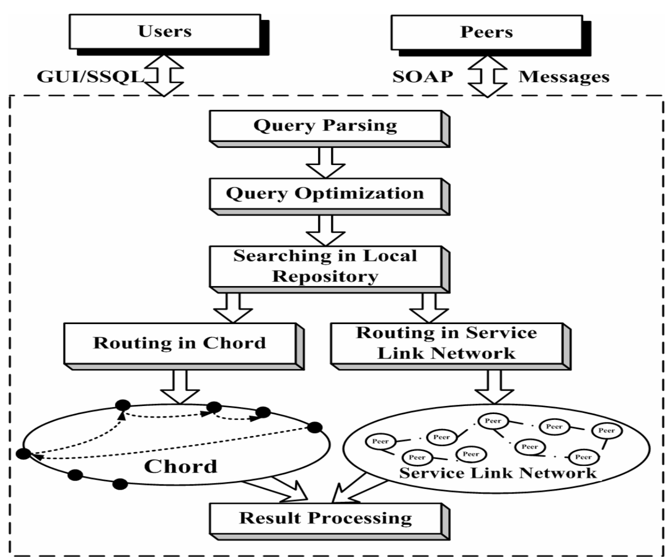

Figure 1. General architecture of the proposed approach.

As depicted in Figure 1, the semantic-link-based service discovery approach consists of the following steps:

(1) *Query Parsing* — To parse and extract the keyword description, the required operation names, the input parameters, and the required output of particular services.

(2) *Query Optimization* — To optimize the service requirement using query optimization techniques.

(3) *Searching in Local Repository* — To search the required services in its local repository first.

(4) *Routing in Chord Overlay*— To lookup the required services in Chord by hashing the operation names and keywords.

(5) *Routing in Service Link Network Overlay* — To forward the service requirement to its neighbors according to the semantic service link types and similarity between them.

(6) *Result Processing* — To analyze the service indices returned by peers in Chord or the service link network, and invoke the corresponding services.

3. SEMANTIC SERVICE LINK NETWORK

3.1 Semantic Service Link Model

As defined in [5], a Web service can be abstracted as *WS*(*OpSet*={op_1: $\sigma_1 \rightarrow \tau_1$, op_2: $\sigma_2 \rightarrow \tau_2$, …, op_n: $\sigma_n \rightarrow \tau_n$}, *CATEGORY*, *KEYWORD*), where *OpSet* is the operation set, i.e., the functionalities that a service can provide, σ is the input, τ is the output, *CATEGORY* is the classification type, and *KEYWORD* is the keyword set.

Input, output, category and keywords can be used to identify the semantic relationships between two web services. Six types of *Semantic Service Links* are defined as follows:

(1) *Equal-to Service Link*. If *Service* (P_i) and *Service* (P_j) has the same *OpSet*, the same *CATEGORY*, and the same *KEYWORD*, then P_i—*s-equu*→P_j and P_j—*s-equ*→P_i.

(2) *Partial-Specialization Service Link*. If *OpSet* (P_j) ⊆ *OpSet* (P_i), *CATEGORY* (P_j) ⊆ *CATEGORY* (P_i), and *KEYWORD* (P_j) ⊆ *KEYWORD* (P_i), then P_i—*s-partial*→P_j.

(3) *Extension-Specialization Service Link*. If *OpSet* (P_j) ⊇ *OpSet* (P_i), *CATEGORY* (P_j) ⊇ *CATEGORY* (P_i), and *KEYWORD* (P_j) ⊇ *KEYWORD* (P_i), then P_i—*s-extension*→P_j.

(4) *Revision-Specialization Service Link*. If *OpSet* (P_j) ∩ *OpSet* (P_i)≠NULL, *CATEGORY* (P_j) ∩ *CATEGORY* (P_i) ≠NULL, or *KEYWORD* (P_j) ∩ *KEYWORD* (P_i) ≠NULL, then P_i—*s-revision*→P_j.

(5) *Non-Specialization Service Link*. If there is no semantic relationship between *Service* (P_i) and *Service* (P_j), then P_i—*s-∅*→P_j and P_j—*s-∅*→P_i.

(6) *Unknown Service Link*. If the semantic relationship between *Service* (P_i) and *Service* (P_j) is unknown, then P_i—*s-N*→P_j and P_j—*s-N*→P_i.

When a peer P_i joins a P2P network, it will randomly take a peer P_j as its neighbor. Then P_i will send SOAP messages to get the services provided by P_j and establish the service link between P_i and P_j.

3.2 Similarity between Services

Let s_1, s_2, s_3 be the *operation-similarity*, *category-similarity*, and *keyword-similarity* of *Service* (P_i) and *Service* (P_j), which could be calculated using approaches in [5]. Let $w_1, w_2, w_3 \in [0, 1]$ be the weights of s_1, s_2, s_3, which satisfy $w_1+w_2+w_3=1$. The progressive weight distribution function $HW(s_i)=\sum_{k=1}^{i} w_k$ is 0-fuzzy metric. By using fuzzy integral approach, the similarity between *Service* (P_i) and *Service* (P_j) can be calculated by *Similarity*= $\bigvee_{i=1}^{3}[s_i \wedge HW(s_i)]$, where "∧" means by the minimum operation, and "∨" means by the maximum operation.

3.3 Reasoning Rules for Semantic Service Links

The heuristic reasoning rules for deriving the semantic relationships between P_i and other peers in P2P networks are listed in Table 1, where $\alpha \in$ { *s-equ*, *s-partial*, *s-extension*, *s-revision*, *s-∅*, *s-N*}.

Table 1. Reasoning Rules for Semantic Service Links

No.	Rules
Rule 1	P_i—*s-equ* →P_j, P_j—*s-equ* →P_k ⇒ P_i—*s-equ* →P_k
Rule 2	P_i—*s-equ* →P_j, P_j—α→P_k ⇒ P_i—α→P_k
Rule 3	P_i—*s-partial* →P_j, P_j—*s-partial* →P_k ⇒ P_i—*s-partial* →P_k
Rule 4	P_i—*s-extension* →P_j, P_j—*s-extension* →P_k ⇒ P_i—*s-extension* →P_k
Rule 5	P_i—*N*→P_j, P_j—α→P_k ⇒ P_i—*N*→P_k
Rule 6	P_i—∅→P_j, P_j—α→P_k ⇒ P_i—*N*→P_k

4. COMPARISON

Current P2P Web service discovery methods are either on unstructured P2P systems or on structured P2P systems. In [2], a service discovery approach based on Gnutella P2P protocol was proposed. However, this flooding-based approach suffers from problems such as traffic and load balancing. Structured P2P service discovery approaches were proposed in [1] and [3]. Compared with approaches in [1, 2, 3], our approach uses semantic service links to represent semantic relationships between services, and provides a more efficient and flexible way for service discovery in P2P networks.

5. CONCLUSION

This paper proposes a semantic-link-based service discovery approach on top of Chord overlay. Our contribution concentrates on a semantic-based infrastructure for flexible service lookup in P2P networks. Simulation results have demonstrated the efficiency and effectiveness of the proposed approach.

6. ACKNOWLEDGEMENTS

The research work is supported by the National Basic Research 973 Program (2003CB317001) and the National Science Foundation of China (No.60273020 and No.70271007). We thank all the team members of China Knowledge Grid Research Group (http://kg.ict.ac.cn, http: //www.knowledgegrid. net).

7. REFERENCES

[1] Emekci, F. et al. A peer-to-peer framework for Web service discovery with ranking. In Proceedings of ICWS'04, California, USA, July, 2004.

[2] Paolucci, M. et al. Using daml-s for p2p discovery. In Proceedings of ICWS'03, Erfurt, Germany, September, 2003.

[3] Schmidt, C. and Parashar, M. A peer-to-peer approach to Web service discovery, World Wide Web, 7 (2) (2004) 211-229.

[4] Stoica, H. et al. Chord: a scalable peer-to-peer lookup service for Internet applications. In Proceedings of SIGCOMM'01, San Diego, CA, USA, August, 2001.

[5] Zhuge, H. and Liu, J. Flexible retrieval of Web services. Journal of Systems and Software, 70 (1-2) (2004) 107-116.