

服务部署与发布绑定的基于 P2P 网络的 Web 服务发现机制

陈德伟 许 斌 蔡月茹 李涓子

(清华大学计算机科学与技术系 北京 100084)

摘 要 Web 服务技术的兴起为 Web 的使用提出了新的计算模型,构建基于面向服务的架构(SOA)的应用是未来分布式计算的发展方向.在 SOA 所要解决的技术问题中,服务的自动发现是难点和关键.文章提出了一种基于 P2P 技术和语义网技术的 Web 服务发现机制,将服务的部署与服务的发布无缝地绑定在一起,在处理用户查询的时候,提出了二阶段查询方法和两层搜索机制,提高了查询的效率.该机制由于采用了 P2P 技术,使得服务提供者可以自由地随时加入、变更和退出,不会影响最终用户的查询效果.文章还提出使用服务成长期和服务僵死期这两个指标,对服务发现机制进行性能评价.

关键词 Web 服务发现;语义 Web 服务;服务部署;服务发布;P2P 网络
中图法分类号 TP311

A P2P Based Web Service Discovery Mechanism with Bounding Deployment and Publication

CHEN De-Wei XU Bin CAI Yue-Ru LI Juan-Zi

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

Abstract Web Services constitute a new computing model for Web application. The application based on SOA is a promising trend of distributed computing. The key and most difficult problem in SOA is how to automatically discover the services according to the end users' query accurately and quickly. A new P2P and Semantic Web based service discovery mechanism is discussed in this paper in which the deployment and publication of a Web Service are bound together. When web service manager deploys his services, the embedded toolkit will create web service description files automatically and put them into the service metadata repository of the Peer. Profiting from using P2P network to exchange the metadata, the service providers are permitted to add and modify and delete services freely without conferring others. When a query is submitted by a service consumer, a two steps querying and two layers searching methods are used to improve the performance. Key words based matching and indexing in the Group layer are used during the first step, and semantic based Goal-Capability matching in the Peer layer is used during the second step of querying. Two metrics, service growth time and service death time are also introduced in this paper to evaluate the discovery performance.

Key words Web service discovery; semantic Web service; service deployment; service publication; P2P network

收稿日期:2004-12-05;修改稿收到日期:2005-02-23.本课题得到国家自然科学基金(60443002)资助.陈德伟,男,1977年生,硕士研究生,主要研究方向为 P2P 网络、语义 Web 服务. E-mail: chendw@keg.cs.tsinghua.edu.cn.许斌,男,1973年生,博士研究生,讲师,主要研究方向为 P2P 网络、语义 Web、Web 服务.蔡月茹,女,1947年生,副教授,主要研究方向为电子商务、Web 技术.李涓子,女,1964年生,博士,副教授,主要研究方向为语义 Web 与 Web 服务、文本挖掘和知识发现.

1 引 言

Web 服务技术的兴起为建立一个适合 Web 环境的、高度分散化、高度自治分布的应用提供了计算模型。这个模型包括了一系列开放的规范,包括 HTTP, XML, SOAP, WSDL, UDDI 等。一系列相互关联的 Web 服务的集合构成了面向服务的架构 (Service Oriented Architecture, SOA), 在 SOA 中有三个基本操作: 服务的发布、查找和绑定。从当前的发展来看, 服务的绑定技术相对比较成熟, 只要用户能获取到所需服务的 WSDL 文件, 就可以利用现有的工具包完成服务的绑定, 构建基于 Web 服务的应用。然而, 面对数量庞大、形态万端的服务群, 怎样描述和索引服务, 怎样查找用户需要的服务, 并且把多个服务动态组合以完成特定功能等, 这些问题成为制约 Web 服务发展的关键问题。为解决上述问题, 出现了服务建模 (Web Service Modeling)、服务发现 (Web Service Discovery)、服务组装 (Web Service Composition)、服务管理 (Web Service Management) 等一系列新的研究热点, 本文着重介绍一种解决服务发现问题的新方法。

Web 服务发现中的主要难题, 一是如何准确、细致地刻画服务能力 (capability), 从而支持用户需求与服务描述之间更精确的匹配操作; 二是如何存储、索引、交换服务元数据, 既保证服务发现的搜索广度, 又将搜索时间限定在用户可接受的范围内。Web 服务发现的研究者为解决第一个问题, 引入了语义网 (Semantic Web) 技术^[1,2], 借助于本体和描述逻辑等逻辑推理系统的使用, 加强服务描述信息的机器可理解性, 支持用户需求和服务能力之间的逻辑推理匹配。为解决第二个问题, 引入了 P2P 技术来处理服务元数据的交换, 克服传统 UDDI 技术中服务元数据集中注册、集中存放对搜索广度带来的限制。

本文在分析各种发现机制的特点, 借鉴国际相关研究项目的经验基础上, 提出将 P2P 网络嵌入到 Web 应用服务器上, 作为服务发现的网络基础。同时通过内置于 Web 服务部署工具内的服务描述文件辅助生成工具, 将服务的部署与服务发布绑定起来。为提高查询性能, 本文设计了 P2P 网络上的二阶段查询方法和两层搜索机制, 由各个 Peer 上的查询代理 (Querying Agent) 和查询响应代理 (Query Answering Agent) 按这种查询方法和搜索机制处

理用户的查询操作。

本文第 2 节介绍系统的总体结构以及服务部署与发布的绑定方法, 第 3 节介绍二阶段查询处理方法, 第 4 节介绍两层搜索机制, 第 5 节对系统的性能进行分析, 第 6 节介绍与本文所述内容相关的主要论文和研究项目, 最后是总结和下一步的研究设想。

2 服务发现系统总体结构

本文所述的服务发现系统在服务部署的 Web 应用服务器上, 嵌入了 P2P 网络的核心模块 (例如, JXTA), 使得 Web 应用服务器在完成 Web 应用服务, 提供 Web 服务运行环境的同时, 还加入到 Web 服务发现的专用 P2P 网络中, 成为 P2P 网络中的一个 Peer。在系统运行期, Web 服务的调用仍然使用现有的基于 Web 的网络环境, 但服务的描述信息通过 P2P 网络进行交换。借助于 P2P 网络的自组织能力, 将分布在世界各地的服务描述信息汇集在一起, 形成丰富的、可用性强的 Web 服务候选集。系统结构如图 1 所示。

同时, 对服务部署工具 (例如, AXIS) 进行改装, 集成服务描述文档辅助生成工具, 当服务提供者通过服务部署工具部署其 Web 服务时, 内置的服务描述文档辅助生成工具将被自动激活。该工具根据服务部署人员填写的描述信息, 按照服务建模本体 (例如, WSMO) 规范的格式生成服务描述文档 (例如, WSMML)。该文档被自动保存到 P2P 网络环境下的服务描述信息库, 作为服务查询处理时的候选服务子集。

在上述操作过程中, 服务提供者在完成部署服务 (deploying) 的同时, 已经完成了在服务查询系统上发布服务 (publishing) 的过程。在传统的服务发现机制中, 这两个过程是完全不同、各自独立的操作过程。例如, 在 UDDI 中, 服务提供者必须手动登录到 IBM, Microsoft 或者其他 UDDI 网站, 手动填写发布信息才能完成服务发布工作。本文所述的服务发现机制, 通过内建 P2P 网络核心模块和服务描述文档辅助生成工具的办法, 省略了显式的服务发布过程, 实现了服务部署与服务发布的绑定, 大大简化了服务提供者的工作。

本文所述系统还在 Web 应用服务器上加载服

Hao He. What is Service-Oriented Architecture. <http://webservices.xml.com/pub/a/ws/2003/09/30/soa.html>

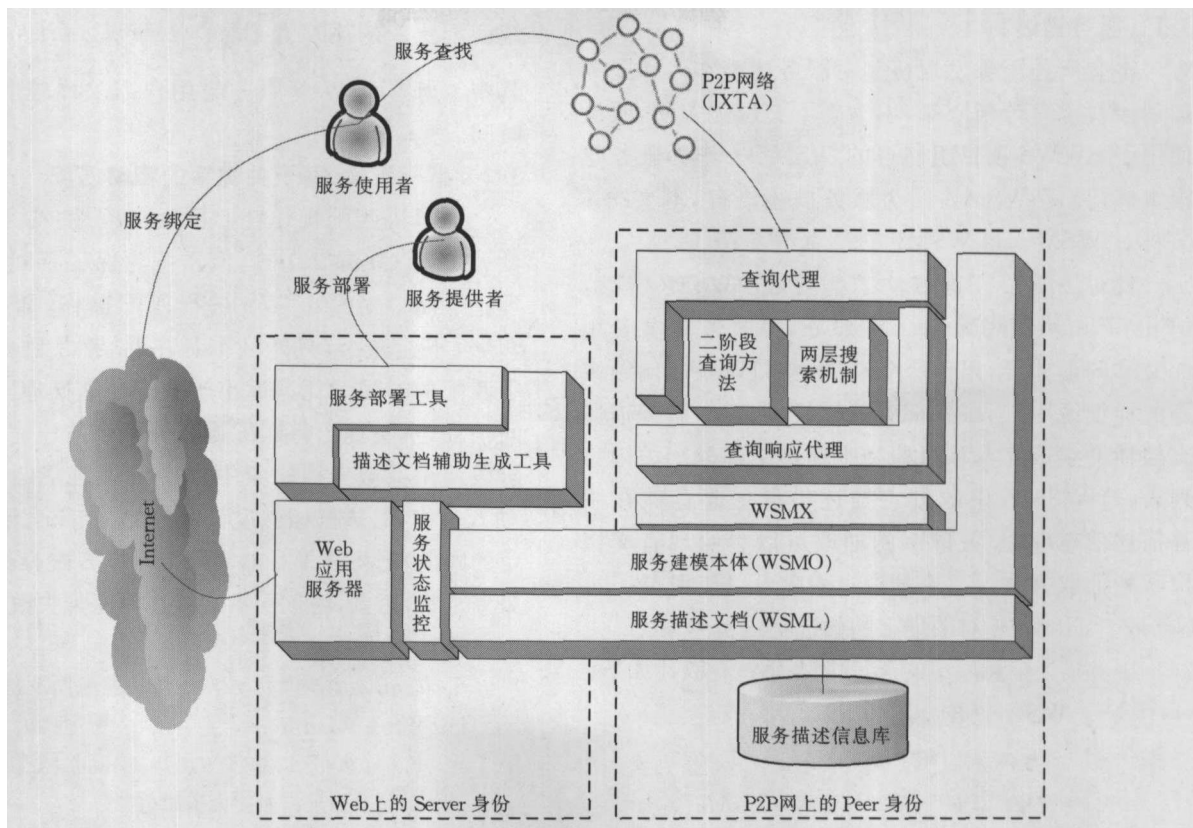


图 1 系统总体结构

务状态监控模块, Web 服务的状态发生改变时, 服务状态监控模块可以动态地更改相应的服务描述文档, 使该服务的描述信息与服务状态时刻保持一致。当 Web 服务出现故障, 不能继续提供服务的时候, 该模块将相应的服务描述文档从服务描述信息库中删除, 从而使该服务从候选服务子集中退出, 实现了服务的自动撤销 (un-publishing), 大大简化了服务管理的难度。

P2P 网络上的所有 Peers 共享相同的 service 建模本体, 并在该本体规范的格式之上建立查询代理、查询响应代理和服务描述信息库。查询代理主要完成三项操作: (1) 将服务使用者提交的服务需求按 service 建模本体规范的格式生成查询请求; (2) 按两层搜索机制确定的搜索方案将查询请求在 P2P 网络上转发; (3) 将 P2P 网络各个 Peer 反馈的查询结果进行汇总和后处理, 然后提交给用户。查询响应代理主要完成两项操作: (1) 接收 P2P 网络上其它 Peer 转发的查询请求, 将用户需求与服务描述信息库中的候选服务子集依次进行匹配推理操作, 将满足用户需求的结果反馈给相应的查询代理; (2) 按照两层搜索机制确定的搜索策略将所接收到的请求在 P2P 网络上进一步转发。根据需要, 某个 Peer 可以只实现查询代理或者查询响应代理中的一个, 但 Web 应用

服务器所在的 Peer 必须实现查询响应代理。

3 二阶段查询方法

Web 服务发现与文档检索相比, 存在两个重要的差别: (1) 由于只有 Web 服务与用户需求可以精确匹配时, 该 Web 服务才能被成功调用, 所以 Web 服务发现对查准率的要求大大高于文档检索。 (2) 在用户需求与 Web 服务描述之间进行精确的匹配比较难度很大, 文档检索中常用的关键字匹配方法达不到查准率要求。在文档检索中, 可以用关键字概括全文信息, 而 Web 服务不能只用关键字来概括, 必须要有精确的服务能力和接口描述。现在主流的匹配推理是采用基于语义的匹配推理, 这种匹配推理机制, 支持复杂的语义描述, 匹配精确度高, 但最大的缺点是匹配推理过程耗时巨大, 在某些情况下推理过程不可判定。在 Web 服务发现系统中, 服务的匹配比较是系统性能瓶颈, 粗糙的匹配比较会导致系统查准率过低, 而精细的匹配比较又会导致匹配耗时过长。本文综合了基于关键字的匹配和基于语义的匹配推理两种查询机制的优点, 提出了用于 Web 服务发现的二阶段查询方法, 该方法在不降低查准率的前提下, 减少了匹配耗时。

3.1 服务描述

本文所述服务发现机制对服务建模本体和服务描述语言没有严格限制,但为了下文叙述方便,假设使用了 SWWS 项目组制订的 WSMO^[3]作为服务建模本体,使用 WSML 作为服务描述语言,本文第 6 节将对 WSMO 和 WSML 进行简介.

针对二阶段查询方法的需要,在 WSMO 规范的 nonFunctionalProperties 部分,由服务提供者用纯文本的形式,给出该服务功能、接口以及所属领域等的详细说明.当服务部署的时候,内置的服务描述文档辅助生成工具从这些文本描述中抽取出关键字列表,与 WSML 中的其它描述信息一起存储在服务描述信息库中.关键字的抽取可以参照现有文档检索的抽取方法^[4,5],在抽取过程中可以使用 WorldNet 或者 HowNet 对关键字的候选集进行约束.

服务描述信息库中的 Web 服务形式化表述如下:

$$\begin{aligned} WS &= \{ K_{ws}, C, I, O_{ws}, M_{ws} \}, \\ K_{ws} &= \{ k_1^{ws}, k_2^{ws}, \dots, k_n^{ws} \}, \\ C &= \{ \text{pre}, \text{ass}, \text{post}, \text{eff}, O_c, M_c \}. \end{aligned}$$

其中, K_{ws} 为关键字列表, k_i^{ws} 为关键字, C 为服务能力, I 为接口, pre , ass , post , eff 分别是 WSMO 规范中描述服务能力的 Pre-conditions, Assumptions, Post-conditions, Effects, M_{ws} 和 M_c 是仲裁器 (Mediators), 用来解决本体和服务描述的异构问题. WSMO 在对 WS 和 C 的表述中使用了本体 O_{ws} 和 O_c , 从而支持基于语义的服务匹配推理. 本文第 6 节将介绍 WSMO 规范中对服务能力的描述方法.

多个 Web 服务的集合称为候选子集 $SubCandidates = \{ WS_1, WS_2, \dots, WS_i \}$.

全 P2P 网络上所有候选子集的并集称为候选集 $Candidates = \bigcup_i SubCandidates_i$.

相应的,用户需求也由两部分组成:一部分是用户对需求的纯文本描述,另一部分是按照 WSMO 中对 Goals 的规范进行的精确描述. Query Agent 从文本描述中抽取出关键字列表. 对用户需求的公式化表述如下:

$$\begin{aligned} R &= \{ K_R, G \}, \\ K_R &= \{ k_1^R, k_2^R, \dots, k_m^R \}, \\ G &= \{ \text{post}, \text{eff}, O_G, M_G \}^{[8]}. \end{aligned}$$

服务发现的目的是从 $Candidates$ 中找到与 R 一致的 WS, 将这样的 WS 列表反馈给查询提出者. 在一个可运行的服务发现系统中, $Candidates$ 集的势十分巨大, 如果采用穷举法进行匹配比较, 系

统匹配比较的耗时为 $T = \sum_j t_j \times |Candidates| \times t$, 其中 t 为单个 Web 服务与用户需求比较所消耗的时间.

3.2 第一阶段, 基于关键字的粗略匹配

为解决匹配操作耗时巨大的问题, 必须尽量降低 t 或者 $|Candidates|$, 但系统为保证召回率要求, 理论上需要对每个 WS 进行匹配操作, 即 $|Candidates|$ 不能减小, 只能从 t 上着手. 考虑到基于关键字匹配的时间消耗远远小于基于语义推理匹配的时间消耗, 本文提出将匹配操作分两个阶段完成, 第一阶段使用类似文档检索中常用的关键字匹配技术, 对 K_{ws} 和 K_R 进行相似度计算, 如果结果大于某个规定的阈值, 则将该 WS 加入第二阶段匹配操作的候选服务子集, 否则, 将该 WS 直接作为匹配失败项丢弃.

算法如下:

```
SubCandidates = ∅ // 第二阶段匹配操作的候选子集
For WSi ∈ Candidates
  If similarity( Kwsi, KR ) > threshold
    // 如果相似度大于阈值
    Then SubCandidates. Add( WSi )
    // 则将该 WS 加入到候选子集中
  End if
End for
Return SubCandidates
```

第一阶段算法简单, 没有复杂的逻辑推理过程, 计算量主要在相似度计算上. 相似度计算可以采用文档检索中成熟的算法, 例如, 向量模型^[6]、TFIDF^[7]等. 阈值可以根据系统性能要求来灵活选取, 如果系统时间消耗超过预期指标, 可以加大阈值, 缩小第二阶段匹配的候选子集; 如果系统召回率低于预期指标, 可以降低阈值, 加大第二阶段匹配的候选子集. 一般而言, 经过第一阶段匹配操作之后, 与用户需求相差很大的 Web 服务将被过滤掉, 减少了基于语义的推理匹配的次数, 根据经验, 这部分 Web 服务往往在候选服务集中占有很大的比例.

3.3 第二阶段, 基于语义信息和逻辑推理的精确匹配

在第一阶段大大缩小候选服务子集的前提下, 第二阶段重点利用服务描述中的语义信息, 用逻辑推理系统对 Web 服务功能和用户目标描述进行精确的匹配. 在使用 WSMO 和 WSML 的情况下, 可以直接使用 SWWS 项目组开发的 Web 服务建模运行环境 (WSMX) 来做这方面的工作. 匹配推理过程主要是利用服务描述中的能力描述 C 和用户需求描述中的目标描述 G , 用一阶谓词逻辑推理机判断

C 是否能满足 G 的要求, 该操作称为目标能力匹配 (*Goal_Capability_Matching*)^[8]. 推理方法为:

将目标 G 定义为基本事实 (Facts), 也就是不含变量的逻辑表达式:

将能力 C 视为一系列规则 (Rules):

推理判断 Facts 是否能让 Rules 成立, 如果能则匹配成功, 否则匹配失败.

形式化描述为:

给定目标 $G = \{ \text{post}, \text{eff}, O_G, M_G \}$,
和能力 $C = \{ \text{pre}, \text{ass}, \text{post}, \text{eff}, O_C, M_C \}$.

求证:

$$PO_{\text{gcm}}^*(G, C) \{ O_C, O_G, M_C, M_G \} =$$

$$CL \exists (\exists i_m, \dots, i_n : (\text{pre} (\text{post} \text{ post})$$

$$(\text{eff} \text{ eff})))$$

整个第二阶段的匹配算法为:

```
Matched = ∅ // 匹配成功的 Web 服务集
For WSi: SubCandidates // 第一阶段匹配操作的结果
  If Goal_Capability_Matching(Ci, G) == True
    // 进行目标能力匹配
  Then Matched.Add(WSi)
End if
End for
Return Matched
```

4 两层搜索机制

如第 3 节所述, 在处理某个用户的查询需求的时候, 系统需要对候选服务集与用户需求进行逐一比较, 如果使用集中式的服务信息存储和匹配计算, 那么全部计算量都将集中在服务器上. 如果说第一阶段查询操作的计算量还在服务器的计算能力范围之内, 那么第二阶段查询操作的计算量将远远超出现有计算设备的计算能力. 因此, 语义 Web 服务的发现必须使用分布式的计算方法, 构建在分布式的计算环境下. 最适合的分布式计算环境就是 P2P 网络. Gartner Group^[9]对 P2P 网络定义为:“以通过虚拟命名空间进行直接连接为特点, 没有上下级的层次控制, 一系列地位平等, 互相提供分布式的运算能力、内容和应用支持的节点构成的网络.”基于 P2P 网络构建 Web 服务发现系统既是系统性能的需要, 同时也符合 Web 服务分布式的自然特点.

但是, 如果不对 P2P 网络进行结构化设计, 每次的查询操作都在 P2P 网络上对所有节点进行遍历和穷举, 网络通信量的爆炸式增长以及网络延时的累积必将使查询操作的时间消耗急剧增加. 因此,

必须对 P2P 网络进行结构化处理, 并应用相应的搜索机制来降低消息转发的复杂度, 减小网络通信量和网络延时. 为此, 本文提出了两层搜索机制来解决上述问题.

4.1 P2P 网络的结构化

参考文献[10~13]中所述 P2P 网络组织方法和 JXTA 规范, 在本文所述系统中, 用组 (Group) 将 P2P 网络划分成两个层次.

如第 3 节所述, 每个 Peer 构建有自己的服务描述信息库, 使用向量 VP 作为该 Peer 服务信息的抽象描述:

$$VP = \{ k_1, w_1^P, k_2, w_2^P, \dots, k_p, w_p^P \},$$

$$K_P = \{ k_1, k_2, \dots, k_p \} = K_{WS},$$

$$w_i^P = \text{count}(k_i).$$

式中 k_i 为关键字, w_i^P 是关键字 k_i 的权重, 上式中以关键字出现的次数作为该关键字的权重, 实际系统中可以根据需要设计更复杂的权重计算方法.

定义 $Peer_1$ 与 $Peer_2$ 的相似度为:

$$\text{Sim}(Peer_1, Peer_2) = \text{Similarity}(VP_1, VP_2)$$

函数 *Similarity* 的具体算法可以参考文档检索中的成熟算法.

相似度大于某个阈值的 Peers 组成一个组 (Group), 一个组内至少有一个 Peer 充当组服务器 (Group Server) 的角色, 组服务器接收并汇总组内所有 Peer 关于自身 VP 的广告信息, 生成组内全部服务描述信息的抽象描述. 该组的抽象描述用向量 VG 表示:

$$VG = \{ k_1, w_1, k_2, w_2, \dots, k_g, w_g \},$$

$$K_G = \{ k_1, k_2, \dots, k_g \} = K_P,$$

$$w_i = w_i^P.$$

相应的, 定义 $Group_1$ 与 $Group_2$ 的相似度为

$$\text{Sim}(Group_1, Group_2) = \text{Similarity}(VG_1, VG_2);$$

定义组 ($Group_1$) 与 Peer ($Peer_1$) 的相似度为

$$\text{Sim}(Group_1, Peer_1) = \text{Similarity}(VG_1, VP_1);$$

定义用户需求 (R) 与组 ($Group$) 的相似度为

$$\text{Sim}(R, Group) = \text{Similarity}(K_R, VG),$$

$$K_R = \{ k_1^R, 1, k_2^R, 1, \dots, k_m^R, 1 \}.$$

每个组的组服务器除维护向量 VG 之外, 还维护一张组间路由表, 路由表结构如下:

$$\text{Route Table} = \{ \text{GroupName}_1, \text{GroupID}_1, VG_1, \dots, \text{GroupName}_i, \text{GroupID}_i, VG_i \},$$

如图 2 所示.

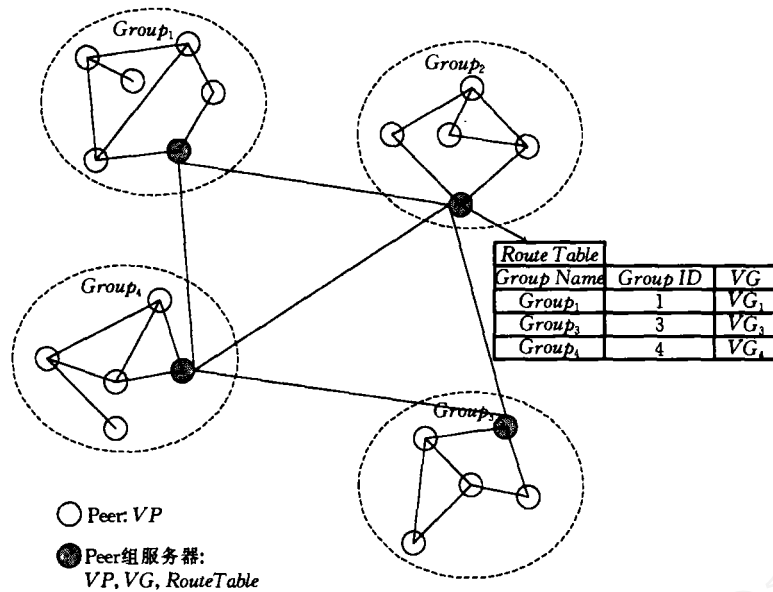


图 2 P2P 网络拓扑结构

为限制路由表的大小,在服务发现系统规模扩大,组的数量较大时,组服务器可以只保留与该组相似度最大(或大于某个阈值)的 N 个组和相似度最小(或小于某个阈值)的 M 个组的路由信息。

P2P 网络结构的维护,包括组的建立、Peer 的加入与退出、信息的更新管理等在本文中不进行详细介绍,有关内容可以参考文献[12,14]。

4.2 第一层搜索,定位到可能含有目标服务的组

当某个 Peer (PeerReq) 的查询代理接受用户的查询请求之后,将用户提供的信息抽象成第 2 节所述的 R ,在自身所属的组内广播消息 PeerReqID, R 。组内的各个 Peer 上的查询响应代理将 R 与本身的服务描述信息按二阶段查询方法进行匹配,如果存在匹配成功的服务集 (Matched),按 PeerAnsID, URL, Matched 的消息格式将结果反馈给 PeerReq。组服务器在完成上述操作的同时,按如下策略将消息在 P2P 网络上转发:

将 R 与路由表中的各个组分别计算相似度,得 $SIM = \{Sim_1, Sim_2, \dots, Sim_i\}$ 。

如果 SIM 中存在 Sim_j 大于规定的相似度阈值 $Sim_{threshold}$,则按 GroupID, hops, PeerReqID, R 的格式将消息转发到路由表中第 j 项记录对应的组。

如果 SIM 中不存在所述情况,则取相似度最小的 Q 个路由记录对应的组转发消息。

若 R 是其它组服务器转发的消息,则当第 j 项记录包含在源组列表中时,不转发消息。

转发消息时,将自身所在组的 GroupID 添加到源组列表 GroupID,同时置 $hops = hops + 1$,若 hops 已经达到跳数上限 MaxHops,则消息不再转发。

这种消息转发机制利用了关键字列表,对 Peer 组进行了一次筛选,从而使得网络上消息转发数量大大降低。第 和 第 条规则保证消息不会循环和无限转发。根据要求,可以通过调整 $Sim_{threshold}$, Q , MaxHops 等参数来调整系统性能。

4.3 第二层搜索,定位到含有目标服务的 Peer

当组服务器接收到 P2P 网络转发的查询请求时,除完成上面所述的转发任务外,还将原始的查询消息 PeerReqID, R 从收到的消息中解析出来,在组内以广播的方式转发。组内各 Peer (包括组服务器本身)在收到查询消息之后,查询响应代理启动二阶段查询方法,将 R 与自身服务描述信息库中的服务进行匹配操作,如果存在匹配成功的 Matched,则直接按 PeerAnsID, URL, Matched 的格式将结果反馈给 PeerReq。

5 性能分析

5.1 性能指标

Web 服务发现机制的评价目前没有公认的一致的评价标准,在绝大多数论文中,大家仍然借用信息检索中使用的查准率和召回率指标作为评价标准。

$$\text{查准率 } Precision_{ws} = \frac{|A \cap B|}{|B|};$$

召回率 $Recall_{ws} = \frac{|A \cap B|}{|B|}$. 其中 A 为标准结果集, B 为返回结果集, 在本文所述系统中 B 为各 Peer 返回的匹配结果集的并集, $B = Matched$.

在文献[10]中作者针对基于 P2P 网络的文献检索的特点, 提出使用 Peer 数量、文档数量、文档分散状况、网络拓扑、广告方式、Peer 选择(用于消息转发)算法、最大跳数等 7 个输入参数, 文档的查准率、召回率, Peer 选择的查准率、召回率, 消息数量等 5 个输出参数作为指标来衡量系统性能. 在本文所述系统中, 文档分散状况即 Web 服务的分散状况, 基本上是随机分布, 网络拓扑、广告方式、Peer 选择算法都是固定的, 因此可以使用 Peer 数量、Web 服务数量、最大跳数 3 个量作为评价的输入参数. 输出参数使用 Web 服务的查准率、召回率、查询时间 3 个量进行评价. 因为 Peer 选择的查准率和召回率最终会影响到 Web 服务的查准率和召回率, 没必要单独拿出来作为输出参数. 同时, 与消息数量相比, 查询时间更能体现一个服务发现系统的性能优劣, 所以, 不采用消息数量, 而采用查询时间作为评价的输出参数.

为了突出分析本文所提出的服务部署与发布绑定机制的特点, 另定义如下两个评价指标作为输入参数:

定义 1. 服务成长期为从服务部署到服务可在全网的任一 Peer 被成功发现的时间间隔.

$Growing Time_i = \max_j \{ \min Time(WS_i, Candidates_j) \} - Time_i^{Deployment}$,
式中 $\min Time(WS_i, Candidates_j)$ 表示服务 WS_i 加入到 $Peer_j$ 上查询代理的候选服务集的时间, $Time_i^{Deployment}$ 表示服务 WS_i 部署的时间.

系统的平均服务成长期

$Growing Time = Average(Growing Time_i)$.

定义 2. 服务僵死期为从服务撤销或者失效到全网的任一 Peer 都不再视该 Web 描述信息为候选 Web 服务的时间间隔.

$dying Time_i = \max_j \{ \min Time(WS_i, \notin Candidates_j) \} - Time_i^{UnDeployment}$,
式中 $\min Time(WS_i, \notin Candidates_j)$ 表示服务 WS_i 从 $Peer_j$ 上查询代理的候选服务集中删除的时间, $Time_i^{UnDeployment}$ 表示服务 WS_i 撤销或失效的时间.

系统的平均服务僵死期

$Dying Time = Average(Dying Time_i)$.

由于本文所述系统将服务的部署与发布绑定在一起, 服务的部署时间即为服务的发布时间, 服务的撤销(失效)时间即为服务的反发布(Un-publishing)时间. 所以, 某个服务的成长期等于 Peer 所发出的关于该服务的广告送达最后一个组服务器的时间消耗. 相应的某个服务的僵死时间等于 Peer 发出的关于该服务的撤销广告送达最后一个组服务器的时间消耗. 可以得出如下公式:

$Growing Time = Dying Time = Average(Length) \cdot k$,
式中 $Average(Length)$ 表示 P2P 网络中消息转发路径的平均长度, k 是一个与网络传输速度有关的常数.

P2P 网络 $Average(Length)$ 是网络节点数的函数, CAN 系统中是 $O(N^{1/d})$ 的量级^[15], Chord 系统中是 $O(\log N)$ 的量级^[16]. 总之, $Average(Length)$ 随网络规模(即 Peer 的数量)的变化并不十分明显, 当网络状况相对稳定的时候, 可以假设它是一个常量. 所以,

$Growing Time = Dying Time = k$.

而在其他服务部署与发布不绑定的服务发现机制中, 由于含有人的手工操作, 系统的平均成长期和平均服务僵死期是不可预料的, 而且在相当规模的网络状况下, 一般都远远大于 k . 例如, 在 UDDI 系统下, Web 服务成功部署在应用服务器上之后, 部署人员要手动登录 UDDI 网站, 手动填写相关信息, 这个过程将算到该服务的生长期中, 更有甚者, 部署人员可能会在若干天后才做这些工作. 服务失效之后的情况往往更糟糕, 服务管理人员甚至可能会忘记要到 UDDI 网站上去撤销他登记的服务信息, 这将导致该项服务长时间处于僵死状态.

5.2 查准率分析

定理 1. 是否使用第一阶段查询不影响系统的查准率.

因为 $B = Matched$, 而某个服务 WS_i 是否加入到 $Matched$ 集合中是在第二阶段查询时决定的. 因此, 影响查准率的是基于语义和逻辑推理的匹配过程, 是否使用了第一阶段查询方法来缩小候选服务子集对系统查准率不造成影响. 有如下近似公式:

$Precision_{ws} = Precision_{2nd_Step_Query}$.

如果系统集成 WSMX 来进行第二阶段匹配操作, 则本文所述系统的查准率等于 WSMX 系统的查准率.

定理 2. 系统平均服务僵死期的增加会导致系统查准率的下降.

设服务撤销(失效)速度为 a , 即单位时间内有 $a \cdot 100\%$ 个服务被撤销或者出现故障, 服务功能失效. 设系统中平均有 $a \cdot k \cdot 100\%$ 的 Web 服务处于僵死状态. 处于僵死状态的 Web 服务 WS_{Dying} 具有如下特性:

$$WS_{Dying} \subseteq Candidates \quad (1)$$

$$WS_{Dying} \subseteq A \quad (2)$$

设系统中全部僵死状态的服务集合为 $Dying = \{WS_{Dying}\}$. 就某次查询而言, 设系统中存在僵死状态的 Web 服务时查询结果集为 B , 标准集为 A ; 系统不存在僵死状态的 Web 服务时查询结果集为 B , 标准集为 A . 根据特性(1)有 $B = B$, 根据特性(2)有 $A \subseteq A$, 而且 $A - A \subseteq Dying$. 从而有

$$|B| = |B|,$$

$$|A - B| \leq |A - B|,$$

$$Precision_{ws} = \frac{|A - B|}{|B|} \leq \frac{|A - B|}{|B|} = Precision_{ws}.$$

上式证明当系统存在僵死状态的服务时, 系统的查准率会降低.

下面考虑一种特殊情况, 假设: (1) $Candidates$ 集中的每个元素都具有相同的影响因子(定义服务 WS_i 的影响因子为 $\frac{Count(WS_i - A)}{Count(R)}$, 式中 $Count()$ 表示对某种情况进行次数统计); (2) $Candidates$ 集中每个元素撤销(失效)的概率都相同.

则根据假设(1)可进一步假设:

$$\frac{|A|}{|Candidates|} = \frac{|A - A|}{|Dying|} \Rightarrow \frac{|A - A|}{|A|} = a \cdot k,$$

根据假设(2)可进一步假设:

$$\frac{|(A - A) - B|}{|A - B|} = \frac{|A - A|}{|A|} \Rightarrow$$

$$\frac{|(A - A) - B|}{|B|} = \frac{|A - A|}{|A|} \cdot \frac{|A - B|}{|B|},$$

则有系统查准率降低:

$$Precision_{ws} - Precision_{ws} = \frac{|A - B|}{|B|} -$$

$$\frac{|A - B|}{|B|} = \frac{|(A - A) - B|}{|B|} = Precision_{ws} \cdot a \cdot k.$$

上式说明 k 越大, 即系统平均服务僵死期越大, 系统查准率 $Precision_{ws}$ 将越小. 根据 5.1 节的分析, 由于本文所述系统将服务部署与服务发布绑定, 从而大大缩短了系统平均服务僵死期, 可以提高系统的查准率.

在一般情况下定理 2 的证明略.

5.3 召回率分析

定理 3. 两层搜索机制中第二层搜索不影响系统的召回率.

因为第二层搜索是在组内以广播的方式进行的, 对候选服务子集进行了穷举查询. 所以, 在第二层搜索过程中对召回率产生影响的是查询匹配方法, 与搜索机制无关.

虽然第二层搜索不影响系统的召回率, 但由于在第一层搜索时没有对候选服务集进行穷举, 系统过滤掉了认为最不可能属于标准集的服务, 但不能完全排除过滤掉的服务中含有标准集中的元素的可能, 所以, 两层搜索机制的第一层搜索会导致系统召回率的降低. 总的来讲, 两层搜索机制是通过牺牲系统召回率来达到缩短查询时间的目的. 有如下近似公式:

$$Recall_{ws} = Recall_{1st_Layer_Search} \cdot$$

$$Recall_{1st_Step_Query} \cdot Recall_{2nd_Step_Query}.$$

两层搜索机制对系统召回率的影响大小主要取决于三个方面的因素:

相似函数 $Similarity()$ 的设计;

第一层搜索时采用的相似度阈值 $Sim_{threshold}$ 的选取;

组服务器进行消息转发是最大跳数 $MaxHops$ 的选取;

当 $MaxHops = Numberof\ Groups - 1$, 且 P2P 网络平台可以保证 P2P 网络拓扑为连通图时, 若相似函数和相似度阈值的选取满足下式, 则第一层搜索也不会影响系统的召回率:

$$(Similarity(K_R, V G_i) < Sim_{threshold})$$

$$\forall x((x \in SubCandidates(Group_i)) \rightarrow (x \in A)).$$

虽然文中所述系统不能保证系统的召回率达到最优, 但提供了调节机制, 只要调整系统参数设置, 即能达到不同召回率指标的要求.

定理 4. 系统平均服务成长期的增加会导致系统召回率的降低.

对于处于成长期的 Web 服务, 有如下特点:

$$WS_{Growing} \subseteq Candidates \quad (3)$$

$$\exists R(WS_{Growing} \subseteq A(R)) \quad (4)$$

$A(R)$ 表示关于用户查询请求 R 的标准集. 上式说明, 存在某些情况, 标准集中的元素不属于候选服务集, 从而使得 $|A - B| < |A - B|$. 式中 A 和

B 分别表示没有处于成长期的服务时查询的标准集和返回集, $A = A$, 从而有

$$Recall_{ws} = \frac{|A \cap B|}{|A|} < \frac{|A \cap B|}{|A|} = Recall_{ws}$$

上式说明在某些情况下, 系统中存在处于成长期的服务时, 比不存在成长期的服务时的召回率要低。

定理 4 的详细证明略。

根据 5.1 节的分析, 本文所述系统将服务部署与服务发布绑定, 可以缩短系统平均服务成长期, 从而有利于提高系统召回率。

5.4 查询时间分析

对于某 Web 服务 $WS = \{K_{ws}, C, I, O_{ws}, M_{ws}\}$ 和某查询请求 $R = \{K_R, G\}$, 根据经验, 有 $T(\text{Similarity}(K_{ws}, K_R)) \ll T(\text{Goal_Capability_Matching}(C, G))$, 其中 $T(\text{Function})$ 表示执行 Function 所需要的时间。由于二阶段查询方法中的第一阶段查询使用 $\text{Similarity}()$ 过滤掉了很大一部分候选服务, 从而节约了匹配操作的计算时间。

相对集中式的服务查询系统(如 UDDI)而言, 本文所述 Web 服务发现系统构建在 P2P 环境之上, 某次查询的计算分散到了各个不同的 Peer 上, 达到了将作业并行处理的效果, 因此也可以缩短查询处理时间。但不足的是, 基于 P2P 的查询处理过程相对来说有更多的网络通信, 因此网络的延时比较大。

对于本文所述系统的平均查询时间, 有如下估算公式:

$$T_{\text{routing}} \cdot \text{Max Hops} + T_{\text{translating}} \cdot (\text{Max Hops} + 3) + T_{1\text{st_Step_Query}} + T_{2\text{nd_Step_Query}},$$

$$T_{\text{routing}} \cdot T(\text{Similarity}()) \cdot \text{Size}_{\text{Routing_Table}},$$

$$T_{1\text{st_Step_Query}} \cdot T(\text{Similarity}()) \cdot \text{Number}_{\text{Web_Services_in_Peer}},$$

$$T_{2\text{nd_Step_Query}} \cdot T(\text{Goal_Capability_Matching}()) \cdot \text{Size}_{\text{SubCandidates}},$$

其中 T_{routing} 为 Peer 组服务器上一次路由操作的平均时间消耗; $T_{\text{translating}}$ 为 Peer 到 Peer 的平均消息传递时间, 即平均网络延时; $T_{1\text{st_Step_Query}}$ 为 Peer 上一次第一阶段查询的平均时间消耗; $T_{2\text{nd_Step_Query}}$ 为 Peer 上一次第二阶段查询的平均时间消耗; $\text{Size}_{\text{Routing_Table}}$ 为路由表中路由记录个数; $\text{Number}_{\text{Web_Services_in_Peer}}$ 为平均每个 Peer 上的 Web 服务个数; $\text{Size}_{\text{SubCandidates}}$ 为第二阶段查询的候选服务子集中 Web 服务的平均个数。

6 相关工作

6.1 Web 服务建模

在 Web 服务发现研究领域中, 如何用机器可理解的方式清晰、准确地描述 Web 服务, 也就是如何对 Web 服务进行建模是一个难点。Web 服务可以看成是由某个定义域到某个值域的函数, 要准确描述 Web 服务, 就必须形式化地刻画出定义域、值域、映射方法等三个方面的内容。但遗憾的是, 这三个方面中的任何一个都难于用机器可理解的方式进行描述。当前, 在 Web 服务建模方面相对成功的模型(或称作上层本体 Upper Ontology)是 OWL-S 和 WSMO。

OWL-S 即 DAML-S^[17], 是由美国国防部信息开发办公室支持的 DAML 项目组, 在 DAML (DAML + OIL 是 W3C 推荐标准 OWL 的前身) 基础上, 针对 Web 服务的特点制订的关于如何描述 Web 描述的规范。OWL-S 从 ServiceProfile, ServiceModel, ServiceGrounding 三个层次描述一个 Web 服务。ServiceProfile 又叫服务能力广告, 回答“服务需要用户提供什么和能为用户提供什么”的问题; ServiceModel 回答“服务如何工作”的问题; ServiceGrounding 回答“如何使用该服务”的问题。OWL-S 使用 IOPEs (Inputs, Outputs, Pre-conditions, Effects) 来描述服务能力。

WSMO^[3] 是由欧盟 IST 支持的 SWWS 项目组制订的 Web 服务建模规范。WSMO 由 Ontologies, Goals, Web Services, Mediators 等 4 个部分组成。Ontologies 即本体, 为其它各个部分提供统一的、机器可理解的词汇集, WSMO 大量使用了本体, 对 Web 服务描述提供了强大的语义支持; Goals 用来描述用户需求; Web Services 用来描述 Web 服务; Mediators 用来解决异构和互操作问题。Goals 使用 Post-conditions 和 Effects 描述用户需求, 分别从信息空间和现实世界的角度描述目标服务执行后的结果。Web Services 使用 Capability 和 Interfaces 来描述一个 Web 服务, 分别表示服务的功能和如何获得这些功能。对 Web 服务发现来讲, Capability 是最重要的部分, WSMO 中的 Capability 由 Pre-conditions, Assumptions, Post-conditions, Effects 四个方

OWL-S 1.1 Release. <http://www.daml.org/services/owl-s/1.1/>

面表述. Pre-conditions 和 Post-conditions 分别表示信息空间中的前提和结果, Assumptions 和 Effects 分别表示现实世界中的假设和影响. WSMO 中一个突出的特点是在 Ontologies, Goals, Web Services 等 3 个部分大量使用 DefinedBy Logical Expression

的形式, 直接使用一阶谓词逻辑表达式对 Web 服务进行刻画, 这种方法准确性高, 便于机器理解, 使 WSMO 的描述能力得以大大得高. WSMO 使用 WSMML 作为其描述语言. 示例如图 3 所示^[18].

```

.....
importedOntologies { http://www.wsmo.org/ontologies/dateTime ,
                      http://www.wsmo.org/ontologies/trainConnection ,
                      http://www.wsmo.org/ontologies/purchase ,
                      http://www.wsmo.org/ontologies/location }
capability_#
precondition
axiom_#
nonFunctionalProperties
  dc:description hasValue "the input has to be a buyer with a purchase intention for an itinerary wherefore the
                           start- and end location have to be in Austria or in Germany, and the departure date has to be later than
                           the current Date. A credit card as payment method is expected. "
endNonFunctionalProperties
definedBy
  ?Buyer memberOf po:buyer and
  ?Trip memberOf tc:train Trip[
    tc:start hasValue ?Start ,
    tc:end hasValue ?End ,
    tc:departure hasValue ?Departure
  ] and
  (?Start.locatedIn = austria or ?Start.locatedIn = germany) and
  (?End.locatedIn = austria or ?End.locatedIn = germany) and
  dt:after( ?Departure ,currentDate) .
.....

```

图 3 WSMML 示例

虽然 WSMO 提供了强大的语义表达能力和推理能力支持, 但并没有明确服务发现系统的具体组织形式, 也没有明确如何对系统复杂度进行控制. 本文提出的二阶段查询方法和两层搜索机制, 对此是一个有力的补充.

6.2 MWSDI (METEOR-S Web Service Discovery Infrastructure)^[19]

METEOR-S 是美国 Georgia 大学的 LSDIS 实验室主持的关于语义网、工作流和 Web 服务技术的研究项目, MWSDI 是 METEOR-S 项目中关于 Web 服务发布与发现的子项目. 它在 UDDI 规范的基础上引入 P2P 技术, 在 UDDI 服务器之间建立 P2P 网络, 从而将多个服务注册中心连接起来, 共享服务描述信息. 同时, MWSDI 使用一个共享的注册本体 (Registries Ontology) 对各个 UDDI 服务器按知识领域 (例如, 旅游、运输等) 进行分类, 每个注册中心, 即 UDDI 服务器必须映射到注册本体上的一个或多个节点上. 注册中心与注册本体之间的映射关系用于组织 P2P 网络的拓扑结构, 映射到注册本体上相同节点的 Peers 组织成一个 Peer 组, 当用户提交查询时, 需要根据查询所在的知识领域在注册

本体中选定查询范围, 系统根据用户的选择将查询请求路由到相应的 Peer 组.

MWSDI 在处理服务实例与用户查询请求的匹配时, 使用一种简单的基于服务模板的匹配算法. 该算法需要事先定义一系列的服务模板, 服务描述文档和用户查询请求都遵循某个特定模板的格式书写, 模板中的各项都要进行相应的本体标注. 如果用户查询与服务实例使用了相同的服务模板, 则进一步对服务模板中的各项作基于本体的匹配推理.

MWSDI 存在的最大缺点在于, 整个 P2P 网络共享相同的注册本体, 维护注册本体的一致性变得十分困难, 从很大程度上限制了 P2P 网络的自组织能力. 本文所述系统由于采用关键字列表和相似度函数来组织 P2P 网络, 不存在类似的问题. MWSDI 的另一个缺点是基于服务模板的匹配算法过于简单, 不能支持复杂的逻辑推理和匹配比较, 本文所述系统借助于 WSMO 的强大逻辑表达能力和 WSMX 的推理能力, 克服了类似的问题.

7 结论及下一步研究

在借鉴当前国际上对语义 Web 服务发现和分

布式文献检索的研究基础上,本文提出了将服务部署和服务发布绑定,在 Web 应用服务器上内置 P2P 网络平台,通过 P2P 网络实现服务描述信息的自动交换的系统结构,省略了显式的服务发布工作,扩大了 Web 服务发现的候选服务集.同时,本文提出的二阶段查询方法,在服务发现的时间开销和精度约束之间取得了一定的平衡.通过两层搜索机制使 P2P 网络的消息转发复杂度和搜索广度得到了兼顾.

文中所述的服务发现机制中,各 Web 服务的描述信息都由 Web 应用服务器自身存储和维护,这种限制使用每个 Peer 上的服务描述信息库相对很小,从另一个角度来说,用户的一次请求将不得不历经多次消息转发才会到达目标 Peer.一个改进的策略是在 Peer 的服务描述信息库中缓存其它 Peer 上的服务描述信息,这样可以使得查询搜索过程历经较少的跳数就可以找到匹配的服务信息.但这样做的前提是必须较好地解决 P2P 网络环境下的数据一致性问题.

另一个可改进的地方是将两层搜索机制扩展为多层搜索机制,进一步减少广播消息的数量.但面临的问题是,过多的搜索层次必然会导致召回率的下降.

参 考 文 献

- Sivashanmugam K., Verma K., Sheth A., Miller J.. Adding semantics to Web services standards. In: Proceedings of the 1st International Conference on Web Services (ICWS '03), Las Vegas, Nevada, 2003, 395 ~ 401
- McIlraith S., Son T., Zeng H.. Semantic Web services. IEEE Intelligent Systems (Special Issue on the Semantic Web), 2001, 16(2): 46 ~ 53
- Roman Dumitru, Lausen Holger *et al.*. D2v1. 1. Web Service Modeling Ontology (WSMO). WSMO Working Group, Working Draft, 2004. <http://www.wsmo.org/2004/d2/v1.1/20041010/>
- Koller Daphne, Sahami Mehran. Toward optimal feature selection. In: Proceedings of the 13th International Conference on Machine Learning (ML), Bari, Italy, 1996, 284 ~ 292
- Lai Yu-Sheng, Wu Chung-Hsien. Meaningful term extraction and discriminative term selection in text categorization via unknown-word methodology. ACM Transactions on Asian Language Information Processing (TALIP), 2002, 1(1): 34 ~ 64
- Bookstein A.. Explanation and generalization of vector models in information retrieval. In: Proceedings of the 5th Annual ACM Conference on Research and Development in Information Retrieval. West Berlin, Germany, 1982, 118 ~ 132
- Salton G., Buckley C.. Term weighting approaches in automatic text retrieval. Information Processing and Management, 1988, 24(5): 513 ~ 523
- Uwe Keller. Inferencing support for semantic Web services: Tools for semantic support. SDK WSMO Working Group, Working draft: D5.2v0.1, 2004
- Gartner Consulting. The emergence of distributed content management and peer-to-peer content networks. Gartner Inc. Stamford, CT, USA, White Paper, 2001
- Haase P., Siebes R.. Peer selection in peer to peer networks with semantic topologies. In: Bouzeghoub M., Goble C., Kashyap V., Spaccapietra S. eds.. Semantics of a Networked World. Semantics for Grid Databases. Paris: Springer France, 2004, 108 ~ 105
- Srinivasan N., Paolucci M., Sycara K.. An efficient algorithm for OWL-S based semantic search in UDDI. In: Cardoso J., Sheth A. P. Eds.. Lecture Notes in Computer Science 3387, Semantic Web Services and Web Process Composition. USA: Springer-Verlag GmbH, 2005, 96 ~ 110
- Shen Heng Tao, Shu Yan-Feng, Yu Bei. Efficient semantic-based content search in P2P network. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(7): 813 ~ 826
- Haase P., Broekstra J., *et al.*. Bibster-A semantics-based bibliographic peer-to-peer system. In: Proceedings of ISWC2004, Hiroshima, Japan, 2004, 122 ~ 136
- Stephanos Androutsellis-Theotokis, Diomidis Spinellis. A survey of peer-to-peer file sharing technologies. Electronic Trading Research Unit (EL TRUN), Athens University of Economics and Business, White paper, 2002
- Ratnasamy Sylvia, Francis Paul, Handley Mark, Karp Richard, Shenker Scott. A scalable content-addressable network. In: Proceedings of ACM SIGCOMM2001, San Diego, California, USA, 2001, 161 ~ 172
- Ion Stoica, Robert Morris, David Karger, Frans Kaashoek M., Hari Balakrishnan. Chord a scalable peer-to-peer lookup service for Internet applications. In: Proceeding of ACM SIGCOMM 2001, San Diego, California, USA, 2001, 149 ~ 160
- Ankolekar A., Burstein M., *et al.*. DAML-S: Semantic markup for Web services. In: Proceedings of the First International Semantic Web Working Symposium (SWWS): Infrastructure and Applications for the Semantic Web, Stanford University, CA, USA, 2001, 411 ~ 430
- Stollberg Michael *et al.*. D3. 3 WSMO Use Case "Virtual Travel Agency". SDK WSMO Working Group, Working Draft, 2004. <http://www.wsmo.org/2004/d3/d3.3/v0.1/20041008/>
- Verma K., Sivashanmugam K., Sheth A., Patil A., Oundhakar S., Miller J.. METEOR—S WSDI: A scalable infrastructure of registries for semantic publication and discovery of Web services. Journal of Information Technology and Management, 2004, 6(1): 17 ~ 40



CHEN De-Wei, born in 1977, MS candidate. His research interests are in P2P network, semantic Web service, etc.

XU Bin, born in 1973, Ph. D. candidate, lecturer. His research interests include peer-to-peer network, semantic Web and Web service.

CAI Yue-Ru, born in 1947, associate professor. Her research interests include Ecommerce and Web technology.

LI Juan-Zi, born in 1964, Ph. D., associate professor. Her interests include Semantic Web and Web Services, text mining and knowledge discovery.

Background

This paper introduces a new Web Service Discovery mechanism based on P2P network, in which the deployment and publication of services are bound, and 2 steps querying and 2 layers searching methods are used to enhance the discovery performance. This work is supported by the National Natural Science Foundation of China under Grant No.

60443002, namely, Research of Ontology Granularity in Distributed System. The authors have made researches on ontology and its application, such as ontology granularity, ontology interoperability, web service discovery. Currently, they have made some progress in ontology storage, semantic annotation, web service discovering and so on.

第五届全国虚拟现实与可视化学术会议(CCVRV '05)

征文通知

由中国计算机学会虚拟现实与可视化技术专业委员会和中国图像图形学会虚拟现实与可视化技术专业委员会主办、北京航空航天大学承办的第五届全国虚拟现实与可视化技术及应用学术会议将于 2005 年 9 月 24 日~25 日在北京举行。本次会议将集聚国内从事虚拟现实与可视化技术的研究人员和工程技术人员,广泛开展学术交流、研究发展战略、推动成果转化、共同促进虚拟现实与可视化技术的发展与应用。

本次大会录用的学术论文将在我国计算机领域的权威期刊《计算机研究与发展》(增刊)发表。会议将邀请国内外著名专家作专题报告,同时将举办科研成果和最新产品展示会,为各研究开发单位及有关厂商展示自己的成果、产品提供场所。欢迎大家积极投稿。

一、征文范围(包括但不限于)

建模技术	动画技术	可视化技术	多媒体技术	人机交互技术	虚拟制造	仿真技术
分布式系统	空间化声音	模式识别应用	图形平台	网络技术	遥操作技术	VRML 技术
逼真图形图像技术	增强现实	协同操作	数字博物馆	网络游戏	图象绘制技术	可视化地理信息系统
基于图像的视景生成技术	虚拟现实与可视化应用系统					

二、征文要求

- 1、论文未被其它会议、期刊录用或发表;
- 2、要求接受电子投稿(同时提交 Word 与 Pdf 格式文件)
- 3、论文包含:题目、中英文摘要、正文、参考文献等;
- 4、正式论文格式见论文录用通知;
- 5、投稿者请在论文最后务必写清姓名、单位、通信地址、电话及 E-mail 地址。

三、重要日期

征文截止日期: 2005 年 5 月 1 日(收到日期) 录用通知日期: 2005 年 5 月 25 日(发出日期)

四、来稿联系方式

联系单位: 北京航空航天大学 6863 信箱
 联系地址: 北京市海淀区学院路 37 号 邮政编码: 100083
 联系人: 周 忠 吴 威 陈小武
 电 话: (010) 82339332; 82317644; 82317610
 电子邮件: ccvrv05@vrlab.buaa.edu.cn

五、会议网站

http://vrlab.buaa.edu.cn

欢迎上网查询大会各项文件和最新通知。