

基于语义的动态服务组织模型

李 杜, 陈松乔

(中南大学 信息科学与工程学院, 湖南 长沙 410083)

摘 要: 随着互联网环境下服务的大量出现, 如何在动态异构的环境下将它们进行有效地组织和管理以便更有效的满足用户的不同需求是一个亟待解决的问题。针对这一问题展开讨论, 提出了基于语义的动态异构服务组织模型, 实现一个服务的分级组织管理模型分别通过纵向和横向建立关系。它分析各种异构的服务, 按照服务的功能操作、初始状态集和结束状态集信息将它们映射到不同的组织单元, 提高了服务查找的效率, 实现了灵活的服务组合和服务替换。

关键词: 服务组合; 服务组织模型; 语义; 组合服务挖掘

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2008)11-0032-04

Dynamic Organization Model of Services Based on Semantic

LI Du, CHEN Song-qiao

(School of Information Science and Engineering, Central South University, Changsha 410083, China)

Abstract: As there are large sums of services on the Internet, it is urgent that finding some method of organizing and managing the services in the dynamic heterogeneous environment, in order to satisfy more effective user's different demand. Discusses such problem, and presents a dynamic organization model of service based on the semantic, which builds relationships among services through vertical and horizontal. It analyses the various heterogeneous services, in accordance with the functionality of the service operation, the initial state set and the end state set, and services will be mapped to different organizational units. This model heightens the efficiency of service search, as well as implements flexible mechanism for service composition and replacement.

Key words: services composition; services organization model; semantic; composed services mining

0 引 言

随着服务技术的发展, 大量的服务资源被发布出来。由于这些资源都是异构的、异地的, 并且可能是处在不同的网络环境, 为了能够有效地利用这类资源, 将这些服务资源统一并且有机组织起来显得尤为重要。研究工作者对这个问题做了不少的研究, 提出了很多不同的方案: 有的根据物理存储的地域特点, 利用分布式数据库将服务资源信息按照区域来组织; 也有的将服务资源抽象出来统一管理^[1]。

为了将多个独立自治的服务按照其语义及逻辑关系“拼装”起来, 实现服务的粗粒度和自包含性^[2], 以达到面向用户需求的目标。仅仅将服务资源有效组织起来, 已经不能满足用户的需求, 特别是在一词多义、一

义多词的情况下。因此, 在服务资源信息组织的时候引入语义也就成为了必要。研究者在这方面也做出了大量的努力。Benatallah B 等人提出了服务区域(service community)的概念, 考虑到灵活高效的服务替换, 按照接口参数来组织服务^[3]; 也有研究者提出了服务簇(service cluster)的概念, 从服务的功能角度来组织服务^[4]。虽然他们从不同角度对服务资源重新进行了组织, 但是并没有考虑跨区域的异构服务组合和服务库中服务资源的动态的、自学习、自增长的变化。

因此, 文中提出了一个支持语义知识库自增长的组织模型, 它分析各种异构的服务, 按照服务的功能操作、初始状态集和结束状态集信息将它们映射到不同的组织单元, 实现一个服务的分级组织管理模型, 通过映射过程的转换实现了对服务异构性的屏蔽; 并通过初始状态集和结束状态集将功能作深层次的表达, 从而避免了关键词匹配^[5-7]和服务组合模板^[3, 8, 9]匹配带来的不足; 它能够自行推断, 组合出使用频率高的、新的复合服务, 并能通过语义聚合产生新的、更细化的服务类别。

收稿日期: 2008-02-10

基金项目: 国家教育部博士点基金资助项目(20030533011)

作者简介: 李 杜(1982-), 男, 湖南邵阳人, 硕士研究生, 研究方向为软件工程技术、服务组合; 陈松乔, 教授, 博士生导师, 研究领域为软件工程技术、服务组装等。

1 基本术语

定义 1 服务用一个三元组来表示,即 $S = \{Iset, P, Oset\}$ 。三者之间的关系如下:

$$Iset \xrightarrow{P} Oset$$

其中, P 表示服务 S 中封装的操作所能实现的功能; $Iset$ 为服务 S 执行 P 操作所需要的所有初始条件的集合; $Oset$ 为服务 S 执行 P 操作后的结束状态的集合。其中不同服务中的 $Iset$ 和 $Oset$ 中的参数个数和参数类型都不一定相同。如果两个服务的初始条件集和结束条件集分别相同,则两个服务的功能是一致的。如图 1 所示: S 表示初始条件集, E 表示结束状态集,从 S 出发有两条路径 W, T 都到达 E 。由于用户关心的是服务的结果和质量,至于服务采用何种方式和过程显得并不重要。因此可以认为图 1 中的 W, T 在功能上是等价的。

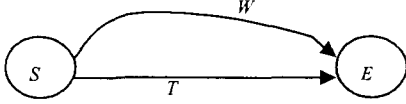


图 1 与过程无关的功能等价

定义 2 基本服务:根据描述信息能够在网络上找到实体服务的服务。

定义 3 复合服务:由 2 个或者 2 个以上的基本服务根据一定的结构(并行、顺序、循环)组合成新的服务,以满足更大的用户需求。

定义 4 类别服务:具有某一特征、功能的一组基本服务的集合的代表性服务,是虚拟的服务,体现了该组服务集合中的共性。

定义 5 相似关系(similar-to):两个同一层的服务之间的相似度达到一个给定的指标(如 0.8)后,将两个服务联系起来所形成的关系。如服务 a 相似于 b ,表示为 $a \sim b$ 。

相似度指两个服务在功能相近的基础上接口的相似程度。如 A, B 两个服务,且 $A.p \approx B.p$,它们的相似度计算如下:

$$S = \frac{A.Iset \cap B.Iset}{A.Iset \cup B.Iset} \times \frac{A.Oset \cap B.Oset}{A.Oset \cup B.Oset}$$

定义 6 类别关系(kind-of):子类与父类的关系。如 C 是 A 的子类,则表示为 A 包含于 C 。 C 的存在与否不影响 A 。

定义 7 组合关系(part-of):子服务与复合服务的关系。

一个或多个服务组装成一个复合服务的基本方式有三种:顺序连接、并行连接和循环连接。用符号表示分别为: $X = A + B$, $X = A \parallel B$, $X = A + A$ 。其中 X 表示要组装的复合服务, A, B 为参与组装的服务。涉

及到的操作其实只有两类,循环连接是顺序连接中的一种特殊情况,任何一个复合服务都可以描述成两种操作的不同组合,如 $X = (A + B) \parallel (C + C) + D$,子服务中缺少一个则复合服务也将不存在。

定义 8 本体用一个四元组表示: $O = \{C, R_c, F_u, r_u\}$ 。其中, C 表示服务相关概念的集合; R_c 表示服务之间关系的集合(如 similar-to, kind-of, part-of); F_u 表示服务函数表达式的集合(如 $X = (A + B) \parallel (C + C) + D$); r_u 表示服务组织模型中涉及的相关规则的集合(如规则 1 和规则 2)。

规则 1 若 $X = A + B$,使得等式成立的必要条件为:

$$\begin{cases} X.Iset = A.Iset \cup (B.Iset - A.Oset) \\ X.Oset = B.Oset \cup (A.Oset - B.Iset) \\ A.Oset \cap B.Iset \neq \emptyset \end{cases}$$

规则 2 若 $X = A \cup B$,使得等式成立的必要条件为:

$$\begin{cases} X.Iset = A.Iset \cup B.Iset \\ X.Oset = B.Oset \cup A.Oset \end{cases}$$

规则中 X 为复合服务, A, B 为基本服务或复合服务,符号 \cup 、 $-$ 、 \cap 分别表示两个集合的并、差、交操作。

2 服务组织模型 DOMSBS

2.1 纵向组织

DOMSBS 服务组织模型分三层。第一层和第二层存放的是类别服务和复合服务,第三层存放的只有原子服务。类别服务和复合服务可以互为父类,每个基本服务必属于一个类别服务,但不一定属于某个复合服务。

2.1.1 类别关系

构造思想:当一个新的基本服务加入模型时,首先将其与模型中第一层的所有类别服务进行相似度比较,是否存在一个服务相似度满足第一层的要求,有则加入到该类别服务中;无则以该基本服务为模板创建一个新的类别服务并加入其中。在服务进入第二层时,以同样的方式进行操作。具体过程见算法 1。

算法 1. 类别关系生成算法

输入:一个新的基本服务 a

输出:NULL

步骤:

1) 设服务组织模型的根目录为 R , GetInto(R , a);

2) 计算 R 目录下各服务与 a 服务的相似度,并得到与 a 相似度最大的一个服务 K , Similar(K, a) =

GetMax(R, a);

3) 如果 $\text{Similar}(K, a) > \alpha$, 则 a 服务进入到类别服务 K 的目录下, $\text{GetInto}(K, a)$; 否则, goto 步骤 6;

4) 计算 K 目录下各服务与 a 服务的相似度, 并得到与 a 相似度最大的一个服务 D , $\text{Similar}(D, a) = \text{GetMax}(K, a)$;

5) 如果 $\text{Similar}(D, a) > \beta$, 则将 a 服务加入到类别服务 D 的目录下, $\text{AddTail}(D, a)$; 否则创建一个新的类别服务 $A = \text{Creat}(K, a)$, 并将 a 服务加入, $\text{AddTail}(A, a)$;

6) 创建一个新的类别服务 $A = \text{Creat}(R, a)$, 进入 A 的目录下 $\text{GetInto}(A, a)$, 在类别服务 A 下创建 $A' = \text{Creat}(A, a)$, 并将 a 服务加入到 A' 中, $\text{AddTail}(A', a)$ 。

说明:

(1) 算法中的 α, β 分别表示第一、二层相似度匹配要满足的最小阈值; $\text{Similar}(X, s)$ 为基本服务 s 与服务 X 的相似度。

(2) 函数 $\text{GetInto}(X, s)$ 表示基本服务 s 进入服务 X 的目录下。

(3) 函数 $\text{GetMax}(X, s)$ 获得 X 的目录下所有服务与基本服务 s 相似度最大的值。

(4) 函数 $\text{Creat}(X, s)$ 表示以服务 s 为参照在服务 X 的目录下创建一个新的类别服务。

(5) 函数 $\text{AddTail}(X, s)$ 将服务 s 加入到服务 X 的目录队列的末尾。

2.1.2 组合关系

组合关系描述的是几个功能相对较小的服务通过一定结构的组合, 组装成一个功能较大的服务。其中这个功能较大的服务就是复合服务, 而功能较小的这些服务可以是基本服务也可以是复合服务。

组合关系的形成与整个系统的运行有着密切的关系。从服务库中提取出复合服务是一个动态的过程, 组合关系利用数据挖掘技术提取出联系紧密的服务, 对大量的用户请求结果进行分析, 将出现次数多的组合结构提取出来组装成一个新的复合服务。

当有一个用户请求, 通过对这个用户请求的分析、分解, 挖掘一组服务按照某种结构组合, 能够满足用户的需要。如 $X = (A + B + C) \parallel (D + D) + E$ 。首先, 对这个虚拟的请求服务 X 进行分解, 得到最基本的基本服务(在服务库中能找到的基本服务或者复合服务)的组合以及这些组合出现的频率。

过程说明: 在集合 $C1$ 中列举了所有构造请求服务 X 的两项服务组合, 并标记了每项组合的出现频率; 通过对集合 $C1$ 的剪枝, 去掉了出现频率较低的组

合, 留下的组合则可以组合成一个新的复合服务, 如集合 $L1$ 中; 再根据集合 $L1$ 中的服务组合分别向左右延伸一个操作, 得到新的组合方式集合 $C2$; 对集合 $C2$ 中的组合方式剪枝, 得到复合服务集合 $L2$, 依此循环, 直到出现的复合服务集合 L 为空或者延伸的服务组合左右都没有操作, 组合服务挖掘过程就中止。

2.2 横向组织

横向组织的关系只有两种, 即相似关系和同义关系。

同义关系是相似关系中的一个特例, 当相似度达到 1 时, 相似关系转变为同义关系。如服务 a 同义于 b , 表示为 $a = b$ 。

相似关系和同义关系都具有双向性, 即 $a \sim b$, 则 $b \sim a$; $a = b$, 则 $b = a$ 。但是相似关系不一定有传递性, 而同义关系具有传递性, 即 $a \sim b$ 且 $b \sim c$, 但不一定有 $a \sim c$; 而 $a = b$ 且 $b = c$, 则有 $a = c$; 若 $a \sim b$ 且 $b = c$, 则有 $a \sim c$ 。

以第三层的基本服务为例来说明横向的组织关系, 如图 2 所示。

图 2 中的箭头表示同义关系, 直线表示相似关系。根据图 2 描述可以得到一些关系, 如 $a \sim b$, $e \sim f$ 等。

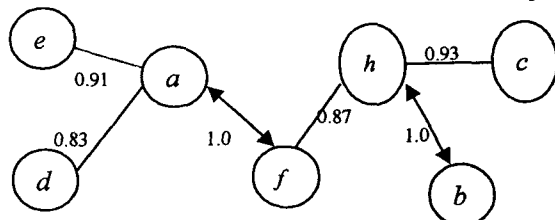


图 2 横向组织的局部关系

以第三层基本服务层为例来建立相似关系, 第一层与第二层相似关系的建立与此构造方法一致。

如图 3 所示, 当一个基本服务 s 进入到它所属的类别服务 K 中时, 将服务 s 与服务 K 中的所有子服务进行相似度比较(若有两个服务互为同义关系, 如 $a = b$, 则这两个服务视为同一个服务), 如果存在一个服务 a 与服务 s 满足相似度指标 γ (设 $\gamma = 0.8$), 则对服务 a, s 建立相似关系; 如果存在一个服务 h 与服务 s 的相似度达到 1 时, 则对服务 h, s 建立同义关系。具体过程见算法 2。

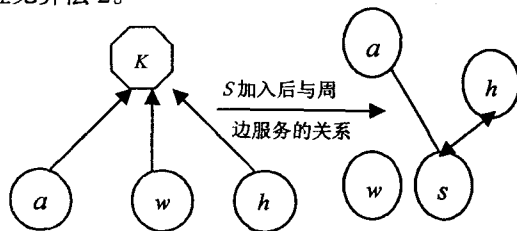


图 3 服务 s 的相似关系建立

算法 2 相似关系生成算法

输入:一个新的服务 s 进入 K 目录

输出:NULL

步骤:

1) K 目录下子服务以链表形式存储, s 进入 K 目录后, $\text{AddTail}(K, s)$;

2) 依此从 K 目录中取出一个非 s 的服务, 计算其与服务 s 的相似度;

3) 如果服务 w 与 s 的相似度 $\text{Similar}(w, s) > \gamma$, 则分别对服务 w, s 的相似关系队列进行修改, $\text{Insert}(w, \text{Similar}(w, s), s), \text{Insert}(s, \text{Similar}(w, s), w)$;

说明:

(1) $\text{Similar}(w, s)$ 为服务 w, s 之间的相似度, γ 为两个服务建立相似关系所要满足相似度的最小阈值。

(2) 函数 $\text{AddTail}(K, s)$ 将服务 s 加入到 K 的目录队列末尾。

(3) 函数 $\text{Insert}(s, \text{Similar}(w, s), w)$ 将服务 w 按照 $\text{Similar}(w, s)$ 值的大小插入到服务 s 的关系队列中。

3 实验和相关工作对比分析

为了在动态异构环境下对服务进行组织和管理, 高岩等人提出了一种支持动态服务组合的 Web 服务三层模型^[1], 简称为 OMDSC, 它按照功能建立起服务簇, 在服务簇中按照接口参数的不同建立起服务社区, 从而构建起 Web 服务三层组织模型。利用层次结构组织和管理服务, 在很大程度上提高了服务查找的效率。然而, 该模型忽视了用户需求是一个大的复合服务的问题: 一个大需求逐步分解成小需求的过程。其次, 该模型提到了服务替换的问题, 但没有根据语义将语义关系密切的服务绑定。因此, 该模型具有一定的局限性。

3.1 实验环境

硬件环境: CPU 为 Inter Pentium IV 2.0GHz, 内存为 512MB, 操作系统为 Windows XP。算法实现工具为 JAVA。

实验数据: 采用随机生成的服务及服务之间的相互关系的数据作为测试用例。两个服务组织模型中的服务总数都为 1000, 分别对两个模型进行 500 次的服务请求, 每次请求包含 1~10 个基本服务。

3.2 实验结果

实验: 服务匹配效率实验。

由于 DOMSBS 随着模型的运行不断地进行复合服务挖掘, 在实验中根据复合服务在模型中的比重不

同进行匹配效率比较的实验。实验结构如图 4 所示。

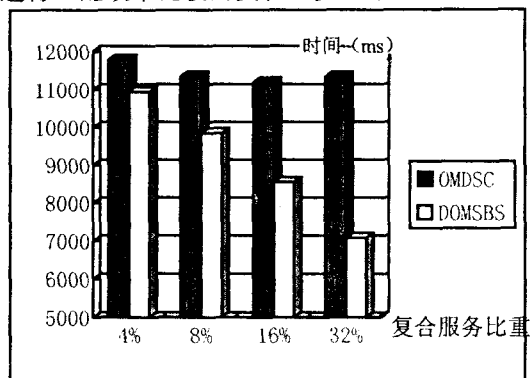


图 4 服务匹配效率的比较

由于 OMDSC 模型没有考虑复合服务在模型中的组织, 针对每次服务请求都要分解成基本服务并对每个基本服务进行匹配, 从图 4 中可以看出 OMDSC 在 4 次不同的服务匹配中所用的时间(单位 ms)差距很小。而 DOMSBS 随着模型中复合服务比例的增大, 匹配的效率明显提高, 与 OMDSC 在相同服务请求时, 匹配所用时间的差距也越来越大。说明 DOMSBS 模型随着系统运行时间越长, 效率越高。

3.3 服务执行中的服务替换比较

服务是动态多变的, 这使得某个具体的用户需求因为组合服务中某个服务的变化而无法正常工作。为了避免这类意外情况对服务组合带来的影响, 有必要提供良好的服务替换机制。由于 DOMSBS 模型在横向关系组织的时候, 采用了相似关系进行关联, 所以在服务执行时发现某个服务出现异常, 就可以及时地搜索该服务的横向关系, 找出与该服务相似度最高的服务用来替换, 仅仅只需要一次操作。而 OMDSC 模型在进行服务替换时, 需要在被替换的服务所处社区中选择一个满足其质量要求的服务, 因此找到替换服务最多需要 $K-1$ 次操作(设每个服务社区由 K 个基本服务组成)。

4 结束语

介绍了一种基于语义的动态异构服务组织模型, 该模型分别通过纵向和横向建立关系, 将异构服务有机地组织和管理。该模型根据不同的功能类别将服务组织起来, 提高了查找的效率, 同时可以自主地提取复合服务, 使得模型可以不断成长, 横向的相似关系使得服务匹配的替换机制得以实现。

参考文献:

- [1] 高岩, 那俊, 张斌, 等. 支持动态服务组合的 Web 服务三层组织模型[J]. 小型微型计算机系统, 2006, 27(10):

(下转第 40 页)

表 1 使用 BP 网络进行的汽车产量预测结果

预测日期	实际值(辆)	预测值(辆)	相对误差(%)
2006.1	133	132.23	-0.6
2006.2	180	171.12	-4.9
2006.3	170	180.14	6.0
2006.4	200	216.82	8.4
2006.5	290	284.53	-1.9
2006.6	220	220.64	0.3
2006.7	210	212.85	1.4
2006.8	165	163.89	-0.7
2006.9	118	127.39	8.0
2006.10	98	102.21	4.3
2006.11	85	85.07	0.1
2006.12	75	82.88	10.5
2007.1	140	129.60	-7.4
2007.2	185	178.11	-3.7
2007.3	189	181.60	-3.9
2007.4	220	216.04	-1.8
2007.5	302	296.83	-1.7
2007.6	250	230.37	-7.9
2007.7	220	219.25	-0.3
2007.8	160	166.85	4.3
2007.9	110	124.88	4.1
平均预测误差率: MAPE=3.9%			
预测准确率: ACCU=98.3%			

一般来说,MAPE 在 5% 以下,ACCU 保持在 95% 以上,就可满足实际需要^[10]。由表 1 可以看出,预测值与实际值的变化趋势基本吻合。预测误差呈随机波动状态,说明预测误差无显著趋势,达到了良好的预测效果。

4 结束语

研究了基于 C#.NET 的 Matlab 引擎技术,成功调用 Matlab 神经网络工具箱,设计并实现 BP 网络对某汽车厂汽车产量的预测,使研究者从大量烦琐的数

学运算中解脱出来,极大地提高了编程效率。不仅如此,还可以优化 BP 网络,以达到更高的预测精度。此外,Matlab 环境之外的引擎技术和接口调用的效率问题也值得进一步思考。

参考文献:

- [1] 赵士伟,赵明波,陈平. 基于 COM 的 MATLAB 与 C#.NET 混合编程的实现与应用[J]. 山东理工大学学报:自然科学版,2006,20(4):28-29.
- [2] 王耀龙,梁小冰,黄萍. Visual Basic.NET 调用 Matlab 神经网络工具箱的实现方法[J]. 现代计算机,2005(1):78-80.
- [3] 陈涛,杨建国,杨江云. 基于 MATLAB 与 .Net 的神经网络推理应用[J]. 微机发展(现名:计算机技术与发展),2004,14(11):33-36.
- [4] 杨建刚. 人工神经网络实用教程[M]. 杭州:浙江大学出版社,2001:11-44.
- [5] 段登伟. 电力系统短期负荷预测的研究[D]. 成都:四川大学,1999.
- [6] 蒋良孝,李超群. 基于 BP 神经网络的函数逼近方法及 MATLAB 实现[J]. 微型机与应用,2004(1):52-53.
- [7] Matlab Neural Network Toolbox user's Guide[M]. US: Mathworks,2006.
- [8] 董维国. 深入浅出 MATLAB 7.x 混合编程[M]. 北京:机械工业出版社,2005:260-262.
- [9] 王俊忠,梁海刚,石旭斌. MATLAB 外部程序接口实现方法研究[EB/OL]. 2006-07-12. [2007-11-1]. <http://www.xinxijishu.org/Article/pc/kaifayingyong/200607/1576.html>.
- [10] 王剑俊. 基于 RBF 网络的城市供水短期负荷预测[J]. 西南给排水,2007,29(3):24-27.

(上接第 35 页)

- 1879-1882.
- [2] 麻志毅,陈泓婕. 一种面向服务的体系结构参考模型[J]. 计算机学报,2006,29(7):1011-1019.
- [3] Benatallah B, Dumas M. Declarative composition and peer-to-peer provisioning of dynamic Web Services[C]//The 18th International Conference on Data Engineering (ICDE'02). San Jose:[s.n.],2002.
- [4] Zhang Liang-jie, Li Bing, Chao Tian, et al. Requirements driven dynamic services composition for Web services and Grid solutions[J]. Journal of Grid Computing,2004,2(2):121-140.
- [5] Ponnekanti S R, Fox A. SWORD: A developer toolkit for Web service composition[C]//In: proceedings of International World Wide Web Conference. Honolulu, Hawaii, USA:[s.n.],2002:83-107.
- [6] Thkkar S. Dynamically composing Web services from on-line source[C]//In: Proceedings of AAAI Workshop on Intelligent Service Integration. Edmonton, Alberta, Canada:[s.n.],2002:1-7.
- [7] Benatallah B, Dumas M. The self-serv environment for Web services composition[J]. IEEE Internet Computing, 2003,7(1):40-48.
- [8] Tosic V, Mennie D, Pagurek B. On dynamic service composition and its applicability to e-business software systems[C]//In: Proceedings of Workshop on Object-Oriented Business Solutions. Budapest, Hungary:[s.n.],2001.
- [9] Li Jing-Shan, Liao Hua-Ming, Hou Zi-Feng, et al. A dynamic service composition method based on semantic interface description in pervasive computing[J]. Journal of Computer Research and Development, 2004, 41(7):1124-1134.