

基于优先级算法的替代 Web 服务检索机制

周 霞, 曹宝香

(曲阜师范大学 计算机科学学院, 山东 日照 276826)

摘 要: Web 服务可替代性是指服务间相互替代的能力, 为避免服务请求者的应用由于调用的服务不可用而发生异常, 在对 Web 服务可替代性进行深入研究的基础上, 提出了一种基于优先级算法的可替代 Web 服务检索机制。该机制采用对等结构的形式, 利用所提出的第一、第二优先级排序对等节点并构造节点树, 在方法层上找到最优可替代服务集合。这种方法可以有效地节省遍历节点树的时间, 从现有的服务集中选择给定服务的替代服务, 保证高效的 Web 服务替代过程, 从而保证服务请求者的应用正常运行。

关键词: 服务检索; 替代 Web 服务; 优先级算法; 方法层

中图分类号: TP301.6

文献标识码: A

文章编号: 1673-629X(2010)10-0090-04

Substitutive Web Service Retrieval Mechanism Based on Priority Algorithm

ZHOU Xia, CAO Bao-xiang

(Department of Computer Science, Qufu Normal University, Rizhao 276826, China)

Abstract: Web service substitutability is the ability to substitute between services. In order to avoid exception occurrence when the application of the service requestor calls an unavailable service, with deeply studying the substitutability of Web services, propose a substitutive Web service retrieval mechanism based on priority algorithm. The mechanism used peer-to-peer architecture, and the proposed first and second priority to sort of peer nodes and to construct a tree of nodes, to find the optimal set of alternative services on operation layer. This method can effectively save time when traversing the node tree and select the alternative service for the given service from the existing service set, to ensure efficient alternative for the process of Web services, thus ensuring the service requestor's application be normally replied.

Key words: service retrieval; substitutive web service; priority algorithm; operation layer

0 引 言

Web 服务是一种新兴的应用模式和分布式计算模型, 是 Web 上业务数据和信息集成的有效机制^[1]。服务请求者的应用可能会因为网络故障、服务过期等原因不能得到正确响应, 即产生异常。为减少这种异常的产生, 可以采用替代服务来代替发生故障的服务, 从而保证服务请求者的应用正常运行。在服务描述文档(WSDL)的学习中可以发现, 一个 Web 服务可以抽象成一个方法集, Web 服务就是通过这个方法集提供多种功能的。假定具有相同名称的方法提供相同的功

能, 那么, 服务间的替代关系可以通过方法集之间的包含程度来度量。一个服务要替代另一个服务, 它必须能够提供另一个服务能够提供的的一个或多个方法^[2]。

实现服务替代过程的关键是高效的替代服务检索技术。已经提出的服务检索方法有: 基于文档结构和接口匹配的方法^[3]、对服务名及方法参数进行聚类的方法^[4,5]、基于语义的 Web 服务发现方法^[6,7]、对 Web 服务采用基于本体集的语义描述的方法^[8-11]等, 以及 Peng 等人提出的一种服务组织方式——服务格, 并从方法层面研究了替代服务检索^[3]。

基于服务格的替代 Web 服务检索有较好的检索性能, 但没有考虑动态服务检索的情况。文中在此基础上, 采用现流行的对等结构的思想来组织 Web 服务, 提出一种基于优先级算法的可替代 Web 服务检索机制, 能有效地找到替代服务, 从而保证服务请求者的应用不受到影响。

收稿日期: 2010-01-20; 修回日期: 2010-04-22

基金项目: 国家自然科学基金项目(60072014); 山东省自然科学基金项目(ZR2009GM009); 山东省公关项目(2009GG100001014)

作者简介: 周 霞(1986-), 女, 硕士研究生, 研究方向为企业信息化与系统集成、Web 服务; 曹宝香, 硕士研究生导师, 教授, 研究方向为企业信息化与系统集成、Web 服务。

1 基于对等结构的替代服务检索机制的体系结构

首先,给出有关替代服务的一些相关概念。

定义 1(Web 服务可替代性):是指服务间相互替代的能力。

定义 2(替代服务):当所调用的服务由于网络故障、服务过期等原因而发生异常时,可以用来代替发生故障的服务提供相同功能,从而保证服务请求者的应用正常运行的服务就是替代服务。用来代替发生故障的服务的服务集合称为替代服务集。

服务可替代性和替代服务的检索对保证基于 Web 服务的松耦合应用稳定性具有重要意义^[2]。Web 服务的组织方式又是影响服务检索效率的关键因素。随着对等网络的发展,对等的思想得到了广泛的应用。文中将采用对等结构的思想来组织 Web 服务^[12],如图 1 所示。

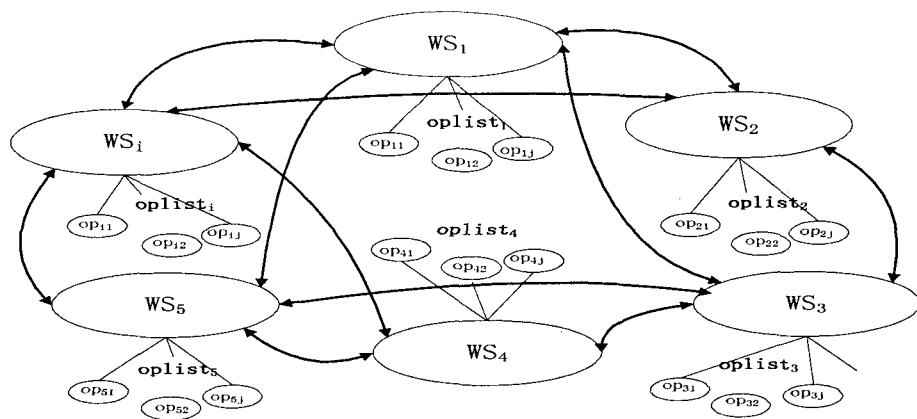


图 1 体系结构

每一个服务对应一个区域,每一个区域对应一个超级节点(Super-Peer, SP)即服务节点(WS_i 代表第 i 个服务),所有服务节点处于对等的位置。属于同一区域的方法集($Oplist_i$ 代表第 i 个服务的方法集)作为子节点(OP_{ij} 代表第 i 个服务的第 j 个方法)也构成一个区域。当服务请求者所请求的服务由于网络故障、服务过期等原因不可用时,就将该服务节点的对等节点按优先级排列作为该服务节点的子节点出现,利用优先级算法找到最有效的替代服务集,从而透明地完成服务请求者的请求。超级服务节点间的替代服务检索如图 2 所示(假设 WS_1 发生故障),这时,所请求的服务节点(WS_1)作为根节点,子节点是其他服务节点,每一个服务节点仍是所对应区域的超级节点。

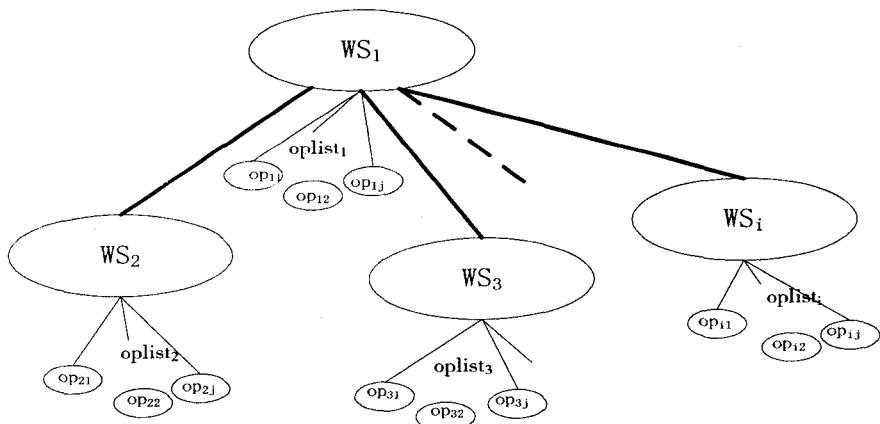


图 2 SP 节点间的服务检索

2 利用优先级算法的替代服务检索机制

现给出关于优先级的规定:

定义 3 (第一优先级 $P1$):令 $P1 = |oplist_i \cap oplist_j|$,即不同服务中相同方法的个数。所得到的数值作为第一优先级,值越大优先级越高。其中 $oplist_i$ 是所请求服务的方法集, $oplist_j$ 是其他对等服务的方法集。只有当 $P1$ 不为 0 时, WS_j 才有可能成为 WS_i 的替代服务。如果 $P1 = 0$,说明 WS_j 代中没有与 WS_i 相同的方法, WS_j 自然也就不可能成为 WS_i 的替代服务。

定义 4 (第二优先级 $P2$): $P2 = |r \cap oplist_j| / |oplist_j|$,即服务请求者所请求的方法数目在可能成为替代服务的的方法集

中所占的比例。其中 r 是服务请求者的所请求的方法集合, $oplist_j$ 是指可能成为替代服务的的方法集。 $P2$ 值越大,优先级越高。只有当 $P1$ 不为 0 时,才计算它的 $P2$ 。

表 1 给出了服务实例及其方法集。

r_1 、 r_2 、 r_3 、 r_4 是对服务 TempService (服务标识为 WS_1)的服务请求,假设 TempService 不可用,现通过优先级来有效地检索 TempService 的替代服务。其中 $r_1 = \{WF, RG\}$ 、 $r_2 = \{WF, RG, TP\}$ 、 $r_3 = \{RG\}$ 、 $r_4 = \{PN, TP\}$ 。

对非 TempService 服务的的服务节点 WS 计算第一优先级 $P1$,如表 2 所示。例如,对 WS_2 的第一优先级 $P1$ 的计算方法为:因为 $oplist_1 \cap oplist_2 = \{WF\}$,所以 $P1 = |oplist_1 \cap oplist_2| = 1$ 。

表 1 Web 服务实例及其方法

服务标识	服务名	方法名(缩写)
WS ₁	TempService	getWeatherForecast(WF) getRegion(RG) getTemperature(TP) getPostNumber(PN)
WS ₂	TravelService	getWeatherForecast(WF) BookTickets(BT)
WS ₃	SwimmingService	getTemperature(TP) PrepareEquipment(PE)
WS ₄	SportService	getWeatherForecast(WF) getRegion(RG)
WS ₅	PublicService	getRegion(RG) getHospital(HP)
WS ₆	TrainService	getTrain(TA) getHospital(HP)

(注:方法后面的大写字母对应方法的缩写)

表 2 各个服务节点所对应的第一优先级

	WS ₂	WS ₃	WS ₄	WS ₅	WS ₆
P1	1	1	2	1	0

根据服务请求 r_1, r_2, r_3, r_4 对 P1 非 0 的服务节点 WS 计算第二优先级 P2, 如表 3 所示。

表 3 P1 非 0 的服务节点所对应的第二优先级 P2

P2 \	WS ₂	WS ₃	WS ₄	WS ₅
r_1	1/2	0	1	1/2
r_2	1/2	1/2	1	1/2
r_3	0	0	1/2	1/2
r_4	0	1/2	0	0

对于表 3 的第三行第二列,表示请求 r_2 对 P1 非 0 的服务节点 WS₂ 的第二优先级。WS₂ 所对应的方法集为 $oplist_2 = \{WF, BT\}$, $r_2 \cap oplist_2 = \{WF\}$, 故 $|oplist_2| = 1$, $|r_2 \cap oplist_2| = 1$, 因而请求 r_2 对服务节点 WS₂ 的第二优先级为 $|r_2 \cap oplist_2| / |oplist_2| = 1/2$ 。从 P2 的计算方法中可以看出, $P2 = 1$ 所对应的服务是优先替代服务, $P2 = 0$ 所对应的服务不可能成为替代服务。

3 优先级算法的替代服务检索算法

当 WS_i 不可用时,就构造一棵以 WS_i 为根节点、非 WS_i 为同一层的子节点的树,其子节点具有以下特点:

- 1) P1、P2 都不为 0;
- 2) 左孩子节点的 P2 不小于右孩子节点的 P2;
- 3) 如果左孩子节点的 P2 等于右孩子节点的 P2, 则要满足左孩子节点的 P1 不小于右孩子节点的 P1。

首先选取最左孩子节点作为最优替代服务节点,并将服务请求的方法集 r 更新为 $r' = r / (r \cap oplist_j)$, 接下来利用更新后的服务请求方法集 r' 重新计算其他孩子节点的 P2, 若 $P2 = 0$, 则删除此孩子节点, 直到 r' 为空, 或者所有孩子节点都已经搜索结束。当 r' 为空时, 说明找到了最优的替代服务集, 服务请求者的请求没有受到影响; 当所有孩子节点都已经搜索结束, r' 仍不为空时, r' 中剩余的方法将无法得到响应, 这说明所请求的服务中本来就不包含 r' 中的方法, 或者 r' 中的方法没有可以替代的。

具体算法描述如下:

Begin

当服务请求者向服务 WS_i 发出请求 r , WS_i 不可用时, 就构造一棵以 WS_i 为根节点的树。

For 对每个服务 WS_j ($j \neq i$) 计算 P1

If $P1 \neq 0$

计算请求 r 对 WS_j 的第二优先级 P2_j

If $P2_j \neq 0$

将 WS_j 作为 WS_i 的孩子节点

EndIf

对 WS_i 的孩子节点排序, 保证 $P2_j \geq P2_{j+1}$

If $P2_j = P2_{j+1}$ 则保证 $P1_j \geq P1_{j+1}$

EndIf

EndIf

EndFor

If WS_i 有孩子节点

将 WS_i 的最左孩子节点 WS_k 选作为替代服务节点

初始化 $r' = r / (r \cap oplist_k)$

While $r' \neq \emptyset$ && WS_i 的所有孩子节点没有遍历结束

从左到右依次计算 WS_i 的所有孩子节点关于 r' 的第二优先级 P2_j

If $P2_j = 0$

删除作为 WS_i 的孩子节点的服务 WS_j

EndIf

对 WS_i 的孩子节点排序, 保证 $P2_j \geq P2_{j+1}$

If $P2_j = P2_{j+1}$ 则保证 $P1_j \geq P1_{j+1}$

EndIf

EndWhile

返回以 WS_i 为根节点的树的所有孩子节点

Else 没有可以替代 WS_i 的服务集

EndIf

End

仍然以请求 r_2 为例: 利用上述方法构造一棵以 WS₁ 为根节点的树, 如图 3 所示, 根据表 3 可以知道 WS₄ 的 P2 = 1, 为优先替代服务, 因而也是 WS₁ 的最优替代服务节点。更新 $r'_2 = r_2 / (r_2 \cap oplist_4) = \{TP\}$ 。并依据 r'_2 更新剩余服务节点的 $P2 = |r'_2 \cap oplist_j| / |oplist_j|$, 得到表 4。

表 4 更新请求后服务节点所对应的第二优先级 P2

P2 \	WS ₂	WS ₃	WS ₅
r ₂	0	1/2	0

此时只有 WS₃ 的对应的 P2 值最大且不为 0,删除以 WS₁ 为根节点的树中 P2 为 0 的节点。于是得到图 4。选取 WS₃ 为 WS₁ 的替代服务节点,继续更新服务请求 $r''_2=r'_2/(r'_2 \cap \text{oplist}_3)=\emptyset$,说明找到了最优的替代服务集,WS₁ 的孩子节点就是 WS₁ 关于请求 r₂ 的替代服务集,服务请求者的请求没有受到影响。

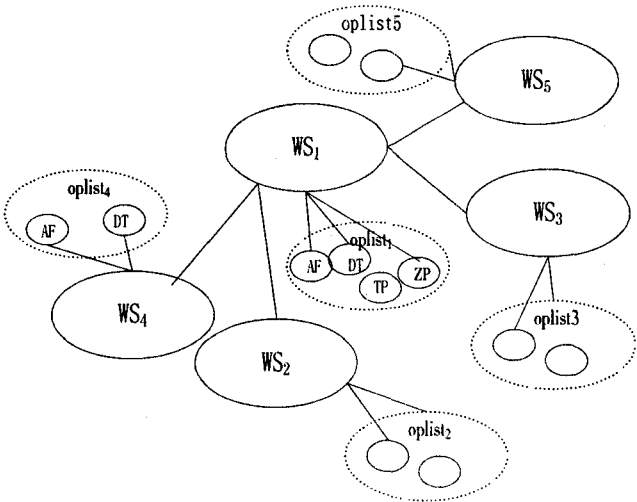


图 3 以 WS₁ 为根节点的树

分析上述算法,在搜索替代服务时,可以快速高效地找到最优替代服务节点,并且该结点只遍历一次,如果某个结点的第一优先级或第二优先级为 0,就不再对其进行搜索。因而可以在短时间内找到最优的替代服务。

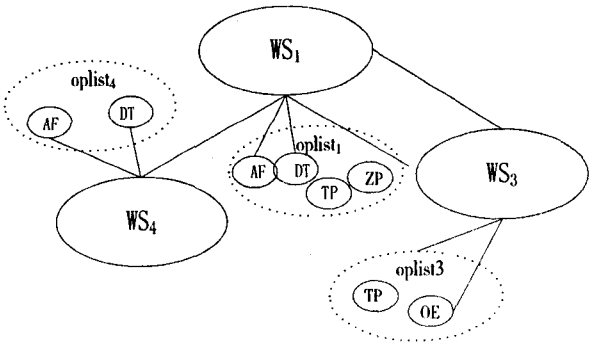


图 4 删除以 WS₁ 为根节点的树中 P2 为 0 的节点

4 结束语

为了更好地避免服务请求者的应用由于调用的服务不可用(如网络故障、服务过期等)而发生异常,用替

代服务来代替发生故障的服务,保证服务请求者的应用正常运行。文中采用对等网络(P2P)体系结构,将所有服务节点构成一个对等网络结构,当所请求的服务发生故障时,利用优先级排序其他对等的服务节点,优先级最高的服务节点将成为最优替代服务节点,并且只被遍历一次;优先级为 0 的服务节点将没有被遍历,从而减少了遍历节点树的时间,有效地从现有的服务集中选择给定服务的替代服务。

下一步,将尝试借助人工智能、搜索引擎等相关技术对优先级算法进行优化,并试着应用到服务搜索管理中。

参考文献:

[1] 李长云,阳爱民,满君丰,等.一种面向按需集成服务的业务模型构造方法[J].计算机学报,2006,29(7):1095-1104.

[2] 彭敦陆,周傲英.服务格-支持替代 Web 服务检索的服务组织结构[J].计算机工程与应用,2007,43(26):8-12.

[3] Wang Y,Stroulia E.Flexible interface matching for Web-service discovery[C]//Proceedings of Fourth International Conference on Web Information Systems Engineering. Roma, Italy:[s. n.],2003.

[4] Dong X, Halevy A, Madhavan J,et al. Similarity search for Web services[C]//Proceedings of the 30th VLDB Conference. Toronto, Canada:[s. n.],2004.

[5] 彭敦陆,周傲英.基于方法聚类的 Web 服务检索技术[J].计算机应用,2007,10(10):2365-2368.

[6] 张 左,左 春,王裕国,等.基于语义扩展的 Web 服务发现方法[J].通信学报,2007,28(1):57-63.

[7] 刘 奎,王一宾.基于语义的 Web 服务发现方法的研究[J].2008,29(19):5140-5142.

[8] Coalition D S.DAML-S:Web service description for the semantic Web[C]//Proceedings of the 1st International Semantic Web Conference (ISWC'02). Sardinia, Italy:[s. n.],2002.

[9] 党伟超,白尚旺.一种基于 OWL-S 的 Web 服务体系结构[J].计算机技术与发展,2007,17(5):195-198.

[10] 牟 帅,黄映辉,李冠宇.基于中文分词的 OWL-S/UD-DI 语义 Web 服务检索模型[J].计算机工程与设计,2009,30(3):705-708.

[11] Srinivasan N,Paolucci M,Sycara K.Semantic web service discovery in the OWL-S IDE[C]//Proceedings of the 39th Annual Hawaii International Conference on System Sciences. Washington:IEEE Computer Society,2006:109-119.

[12] 孟宪福,王迎春,史 磊.基于 P2P 的路由查询机制[J].计算机工程,2008,34(9):148-150.