

基于 UDDI 扩展的 Web 服务 QoS 模型

孙素云

(广东轻工职业技术学院计算机系, 广州 510300)

摘要: 在现有 UDDI 规范中增设 QoS 认证组件, 提出一个 Web 服务 QoS 评估模型。定义一组描述 Web 服务 QoS 属性分类的 tModel, 在引入 QoS 量化概念的基础上, 结合 QoS 认证中心主动监视和服务消费者 QoS 反馈机制, 对 UDDI 中注册的 Web 服务 QoS 进行动态评估和调整, 保证了 Web 服务 QoS 信息的公平性、可信性和实时性。

关键词: Web 服务; 评估模型; 服务质量

QoS Model of Web Service Based on Extending UDDI

SUN Su-yun

(Department of Computer, Guangdong Industry Technical College, Guangzhou 510300)

【Abstract】 This paper brings forward a QoS evaluation model of Web service by adding a QoS authority part on the existing UDDI criterion. It defines a set of QoS categorization tModels, and introduces the concept of QoS quantification. It uses a monitor mechanism initiatively and a feedback mechanism from service consumers for evaluating and adjusting QoS value of Web service dynamically in UDDI. The impartiality, trustworthiness, and real-time characteristics of QoS information are ensured.

【Key words】 Web services; evaluation model; Quality of Service(QoS)

1 概述

随着 Web 服务技术的快速发展, 服务质量问题受到越来越多的关注, 对于关键的商业应用来说, 如果缺乏一套有效的 QoS 管理机制, 可能会严重影响服务的使用效率。在目前的 Web 服务发现机制中, UDDI 数据模型和应用开发接口规范没有涵盖任何类型的 Web 服务 QoS 属性, 通常是由服务请求者应用简单的关键字在 UDDI 中查找功能匹配的 Web 服务, 服务请求者无法指定其他如花费或可靠性等非功能性服务质量需求, 而且 UDDI 规范也没有考虑服务过滤和选择的问题, 导致用户从 UDDI 获得候选服务之后, 无法根据 QoS 信息对服务做进一步的判断, 因此, 一旦服务提供者提供错误或恶意的 Web 服务, 系统将无法对其进行判断和惩罚, 无法实现有效的服务调用。文献[1]研究了 2003 年~2004 年公共 Web 服务的使用情况, 其数量并没有明显增加, 只有大约 34% 的服务可用, 而且每周大约有 16% 的已注册可用的 Web 服务已经失效。文献[2]指出 48% 的 UDDI 中的注册信息是无效的, 这就导致从 UDDI 中查找到的服务信息实际可用性差。因此, 研究如何从数量众多、功能相同或相近、服务质量等非功能特性各异的服务中, 根据服务质量等应用需求动态地选择出最能满足用户需求的服务, 实现应用程序的“按需服务”机制, 已成为计算机领域中一个亟需解决的问题。

文献[3]提出了一种扩展的 UDDI 模型, 文献[4]对网格中的基于 QoS 服务调度策略作了具体描述。本文以这些研究为基础, 参考 QoS 量化的方法^[5], 通过对现有的 UDDI 规范进行扩展, 建立了一个 Web 服务 QoS 评估模型。

2 Web 服务 QoS 评估因素

在 Web 服务的服务质量研究中, 不同的研究工作者提出了不同的服务质量参数体系, 这些服务质量参数体系的内容大同小异, 基本上涵盖了服务质量的不同方面, 包括性能、

可靠性、可提供性、正确性、完整性、费用、安全等, 有的是定量的, 有的是定性的。本文从服务质量参数的重要性程度和易于度量的角度出发, 对 Web 服务的服务质量参数进行选择, 建立了包括 5 个服务质量参数^[4]的参数体系:

(1)性能: 用来评测服务完成请求的速度, 用响应时间为性能的度量指标。响应时间是指工作流或者服务响应一次请求所需要的时间, 可以进一步划分为请求等待时间和请求处理时间。

(2)花费: 用来衡量工作流或者服务一次执行所需要的费用。

(3)可靠性: 指服务或者工作流在一定条件下, 在特定的时间内执行所需要的功能的能力。可靠性可以用执行的成功率来衡量, 也就是服务或者工作流执行成功的次数与总的执行次数的比率。

(4)可提供性: 与可靠性相关, 是指服务或者工作流在一定的条件下, 在特定的时间内提供所需要的功能的能力。可提供性可以用服务或者工作流提供运行的时间与总的时间的比率来衡量。

(5)声誉: 用来描述一个服务的可信度的指标, 它依赖于用户对服务使用的体验。不同的用户对相同的服务可能有不同的体验, 会给予服务不同的声誉值。

3 Web 服务 QoS 评估模型

3.1 模型的体系结构

目前 UDDI 规范没有定义如何对 Web 服务的 QoS 进行描述, 因此, 它不支持基于 QoS 约束的 Web 服务发现; 同时

作者简介: 孙素云(1977 -), 女, 讲师、硕士, 主研方向: 分布式计算

收稿日期: 2007-08-17 **E-mail:** sunsy81@126.com

UDDI 规范对其实现细节没有进行严格的规定,也没有提出参考的实现规范。因此,本文在保证与 UDDI 规范兼容的前提下,对 UDDI 的描述能力进行必要的扩展,使其能提供对 Web 服务 QoS 的描述,并进一步提供基于 QoS 约束的 Web 服务发现。基于上述原则,本文在现有的 UDDI 规范上新增了一个 QoS 认证中心的角色,通过进行 QoS 量化、协商、反馈和监视,实现基于 QoS 约束的 Web 服务发现,其结构如图 1 所示。

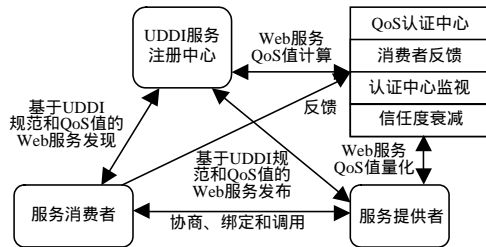


图 1 基于 UDDI 扩展的 Web 服务 QoS 评估模型

在基于 UDDI 的 QoS 评估模型中有 4 种基本角色: Web 服务提供者, Web 服务消费者, UDDI 服务注册中心以及 QoS 认证中心。因为 Web 服务在 UDDI 中注册为 1 个 tModel 对象,所以在 UDDI 中为每个 tModel 关联 1 个 QoS 对象,利用此对象存储服务的 QoS 值。Web 服务提供者在部署好服务之后,向 UDDI 服务注册中心发布携带 QoS 信息的服务描述信息; Web 服务消费者向 UDDI 服务注册中心提出服务发现请求,发现请求中可以包含对目标服务 QoS 指标的需求; UDDI 的 QoS 认证中心作为第三方权威机构负责主动监视并管理所有在 UDDI 中注册服务的 QoS 信息,实现对 Web 服务的 QoS 进行动态评估和调整,从而保证 Web 服务 QoS 信息的公平性、可信性和实时性。

3.2 模型工作机制

在基于 UDDI 扩展的 Web 服务 QoS 评估模型中,在进行服务发布的时候,服务提供者调用量化函数 *Quantify()* 对 Web 服务的 QoS 进行量化,返回一组 QoS 分类信息,并将携带 QoS 分类信息的 Web 服务描述信息在 UDDI 进行服务注册;在发现服务的时候,服务消费者调用 *Quantify()* 操作对用户的 QoS 需求进行量化,返回一组 QoS 分类信息,并携带 QoS 约束在 UDDI 中进行服务查找,然后调用 *Negotiate()* 操作与服务提供者进行 QoS 协商,服务提供者根据系统的当前负载来决定接受或拒绝服务请求;在服务消费者调用服务的时候,服务消费者启动监视过程,对服务调用过程的 QoS 进行测量,并调用 *Feedback()* 操作向 UDDI 的 QoS 认证中心提供有关 QoS 的反馈信息;在整个 Web 服务运行期间, QoS 认证中心负责定时对所有已注册的 Web 服务进行主动监视,并根据 Web 服务注册信息、客户反馈信息、主动监视信息和 Web 服务 QoS 衰减因子,对 Web 服务的 QoS 进行动态评估和调整,实现基于 QoS 约束的 Web 服务发现机制。

4 关键技术的实现

根据 Web 服务 QoS 评估模型的设计结构,评估模型主要由服务提供者、服务消费者、UDDI 和 QoS 认证中心组成,下文对这 4 个组件的主要实现方法进行介绍。

4.1 服务提供者

服务提供者采用现有的标准格式(WSDL)来描述所提供的 Web 服务,为了对服务的 QoS 方面的信息进行了描述,本文对 WSDL 进行扩展,增加一个 `<QoSMetric>` 元素保存 QoS 信

息。因此,在 Web 服务发布之前,服务发布代理需要调用 QoS 认证中心提供的 *Quantify()* 操作,对 Web 服务的 QoS 通告进行量化,返回一组 QoS 分类信息,这些 QoS 分类信息将作为 Web 服务描述信息的一部分发布到 UDDI,从而允许服务消费者根据 QoS 分类信息进行服务查找。在服务被调用前,通过调用协商操作来确定是否接受或拒绝服务请求,确保 Web 服务不会因为大量的并发访问而造成性能下降。

4.2 服务消费者

服务消费者为了根据用户需求选择并调用服务,可以通过发出基于功能性和 QoS 属性的服务请求来实现。首先,服务代理调用 QoS 认证中心提供的 *Quantify()* 操作对用户的 QoS 需求进行量化,结果返回一组 QoS 分类信息,然后将这些 QoS 分类信息作为查询条件的一部分提交给 UDDI 进行服务查找,并绑定满足需求约束的 Web 服务。在 Web 服务调用的时候,服务代理对整个服务执行过程进行监控,对实际发生的服务质量参数进行测量,服务调用结束后,服务代理一方面将调用结果以适当的方式呈现给用户,另一方面调用 *Feedback()* 操作向 QoS 认证中心提供关于 QoS 的反馈信息。

4.3 UDDI

为了保证新模型与当前 Web 服务架构的兼容性,本文在现有的 UDDI 规范上新增了一个 QoS 认证中心的角色,实现 Web 服务 QoS 的评估和基于 QoS 约束的服务发布和查找的功能。为了对 Web 服务的 QoS 属性进行描述,定义了一组表示 QoS 分类属性的 tModel,如性能、花费和可靠性等;在 UDDI 规范中,分类属性是以 *attriName/attriValue* 形式表示的。*attriName* 是一个字符串,是对分类信息的描述;*attriValue* 是一个整数,唯一标识了分类 tModel 中的一个类别,因此,本模型所定义的 QoS 分类 tModel 中,只能表示一组离散值,需要在实际 Web 服务 QoS 属性值和分类 tModel 之间进行转换,在具体实现过程中,引入“量化”概念来表示这种 QoS 属性值与分类 tModel 之间的转换。

4.4 QoS 认证中心

QoS 认证中心作为 Web 服务 QoS 管理的第三方权威机构,主要有 QoS 量化服务、QoS 主动监视服务、消费者 QoS 反馈处理以及 QoS 计算服务功能,主要服务操作接口描述如下:

(1) QoS 量化

为了衡量 Web 服务提供的服务质量,引入 QoS 量化函数 *Quantify()* 量化服务 QoS,该函数返回介于 [0, 1] 之间的实数值来定量地表达 QoS 强度,1 表示最优的服务质量,0 表示无服务质量保证。在 QoS 量化具体实现时,参考文献[5]的量化方法,调用 *Quantify(String xml, bool publish)*;其中, *xml* 是以 XML 描述的服务质量通告或需求;*publish* 是量化类型,true 表示对 QoS 通告进行量化,false 表示对 QoS 需求进行量化,以 XML 描述的 QoS 分类信息返回 QoS 量化结果。

(2) 主动监视

由于很少有组织在发布服务信息后主动对服务信息进行更新,因此必须提供服务的主动监测机制。在本模型中,由 UDDI 中的 QoS 认证中心主动对注册的所有 Web 服务发起监测请求,监测所有服务的当前运行状态,以提高 UDDI 注册中心服务信息的可用性。本文参考网格 Web 资源监视的机制,由 QoS 认证中心定时向注册于其上的服务提供者发送监测请求,并根据监视返回信息更新服务注册库,动态调整相应 Web 服务的 QoS 值,保证 UDDI 上注册信息的实时有效性。

(3) 客户反馈

客户反馈服务接收来自 Web 服务消费者的 QoS 反馈信息并将其保存在 UDDI 的 QoS 认证中心的本地数据库中, 在充分信任客户的前提下, 调用 *Feedback()* 统计 QoS 反馈信息, 并依此分析得到服务的 QoS, 以便对其消费过的服务 QoS 进行评估和调整。

(4) Web 服务 QoS 管理

Web 服务 QoS 管理是由 QoS 计算服务组件完成, 在计算服务 QoS 的时候, 根据预先设定的 Web 服务 QoS 计算算法, 周期性地对本地数据库中存放的 QoS 测量值进行分析, 进而得出每个服务的 QoS 值, 并将结果更新到 UDDI 的 QoS 认证中心, 以保证 UDDI 中 Web 服务 QoS 信息的可信性和实时性。

5 Web 服务 QoS 计算

5.1 属性变换

由于模型中定义的 5 个服务质量指标中, 有些指标的值越大, 服务质量越低(负指标), 如性能、花费; 有些指标的值越大, 服务质量越高(正指标), 如可靠性、可提供性、声誉。因此, 需要对这些指标值进行变换, 在模型中, 对于负指标使用式(1)、正指标使用式(2)来进行具体变换:

$$V_i = \frac{Q_{i\max} - Q_i}{Q_{i\max} - Q_{i\min}}, \text{ 若 } Q_{i\max} - Q_{i\min} \neq 0, \text{ 否则 } V_i = 1 \quad (1)$$

$$V_i = \frac{Q_i - Q_{i\min}}{Q_{i\max} - Q_{i\min}}, \text{ 若 } Q_{i\max} - Q_{i\min} \neq 0, \text{ 否则 } V_i = 1 \quad (2)$$

其中, $Q_{i\max}$ 代表某项指标在 QoS 认证中心的最大值; $Q_{i\min}$ 代表某项指标在 QoS 认证中心的最小值, $1 \leq i \leq 5$ 。通过上面的变换, 可以建立一个 Web 服务的包含 5 个服务指标的质量参数向量。

5.2 计算

在得到服务质量参数向量后, 就可计算每个 Web 服务 QoS 分量值, 计算公式为

$$S(s)_j = \sum_{i=1}^5 (V_i \times W_i) \quad (3)$$

其中, $S(s)_j$ 表示根据主动监视、消费者反馈和服务者发布提供 3 种方式分别计算 Web 服务的 QoS 分量值, $j \in \{1, 2, 3\}$; V_i 表示利用式(1)和式(2)变换后的 QoS 属性值; W_i 是 Web 服务中各项服务质量指标的权重, 它满足: $W_i \in [0, 1], \sum_{i=1}^5 W_i = 1$ 。

用式(3)分别计算得到 $S(s)_j$, 结合相应的 QoS 衰减因子, 进行 Web 服务 QoS 的综合评估, Web 服务 QoS 计算公式可表示为

$$Q_v = \alpha \times S(s)_1 + \beta \times S(s)_2 \times E_i + \gamma \times S(s)_3 \times P_i \quad (4)$$

其中, Q_v 为 Web 服务 QoS 值; $S(s)_1$ 为 QoS 认证中心作为第三方权威机构对 Web 服务 QoS 主动监视统计分析值, 因此, α 是评估服务 QoS 最大权重的影响因子; $S(s)_2$ 为 Web 服务消费者对该 Web 服务 QoS 反馈分析值, 由于它是对 Web 服务 QoS 评估最直接有效的评价因素, 因此 β 也应是比较大权重的影响因子; $S(s)_3$ 为服务提供者对提供的 Web 服务 QoS 值, 由于服务提供者可能夸大他们提供的 Web 服务能力来获取更大利益, 因此 γ 占的权重应比较小。服务行为的最大特点随时间衰减, 因此, 在评估 Web 服务 QoS 时, 引入 QoS 衰减因子 E_i 和 P_i 来反映 QoS 的衰减情况, 并对与时间相关的评估因素(如客户反馈和服务注册提供的 QoS 值)进行修正。

6 结束语

Web 服务 QoS 的评价为 Web 服务请求者正确选择服务提供了依据, 它通过对一个 Web 服务过去的“表现”进行综合, 为预期的行为作出评价。与现有的研究工作相比, 本文在目前仅支持功能性注册与发现的 Web 服务架构基础上, 通过对现有的 UDDI 规范进行扩展, 提出一种支持 QoS 属性的 Web 服务评估模型, 增强了 UDDI 服务的描述和发现能力。

参考文献

- [1] Kim S M, Rosu M C. A Survey of Public Web Services[C]//Proc. of International World Wide Web Conference. New York, USA: ACM Press, 2004.
- [2] Clark M. UDDI Weather Report[EB/OL]. (2006-11-28). <http://www.webservicesarchitect.com/content/articles/clark.asp>.
- [3] 郭得科, 任彦, 陈洪辉, 等. 一种 QoS 有保障的 Web 服务分布式发现模型[J]. 软件学报, 2006, 17(11): 2324-2334.
- [4] 王勇, 胡春明, 杜宗霞. 服务质量感知的网格工作流调度[J]. 软件学报, 2006, 17(11): 2341-2351.
- [5] 潘海军, 陆魁军, 吴朝晖. 基于网格系统的信任量化研究[J]. 计算机应用研究, 2005, 22(10): 49-53.

(上接第 128 页)

6 结束语

通过对该通信网络的综合交换技术体制进行设计、仿真和分析, 可以看到改进的 ATM 和 MPLS 技术在该通信系统中能够有效地结合在一起, 实现了系统新旧设备的兼容以及对于新技术的融合, 提高了系统的整体性能, 对业务的支撑能力能够满足该系统的要求。

参考文献

- [1] Kyas O. ATM 网络技术[M]. 辛再甫, 译. 北京: 中国水利水电出版社, 1998.

(上接第 131 页)

参考文献

- [1] Cesati B. 深入理解 LINUX 内核[M]. 陈莉君, 译. 北京: 中国电力出版社, 2004.
- [2] Rubini A, Corbet J. Linux 设备驱动程序[M]. 魏永明, 译. 北京: 中国电力出版社, 2002.

版社, 1998.

- [2] 李晓东. MPLS 技术与实现[M]. 北京: 电子工业出版社, 2002.
- [3] 信息产业部电信传输研究所. 多协议标记交换(MPLS)总体技术要求[M]. 北京: 人民邮电出版社, 2001.
- [4] 张金文, 王文博. OPNET Modeler 与网络仿真[M]. 北京: 人民邮电出版社, 2003.
- [5] 陈敏. OPNET 网络仿真[M]. 北京: 清华大学出版社, 2004.
- [3] Carlton R D. IPsec VPN 的安全实施[M]. 周永彬, 译. 北京: 清华大学出版社, 2002.
- [4] Krasnyansky M. Universal Tun Tap Device Driver[Z]. (2001-10-02). <http://vtun.sourceforge.net/tun/>.
- [5] 徐千洋. Linux C 函数库参考手册[Z]. 北京: 中国青年出版社, 2002.