

扩展的 Web Service 服务质量模型研究^{*}

牟玉洁 曹健 张申生 张建宏

(上海交通大学计算机集成技术研究室 上海 200030)

摘要 服务质量的描述和计算评价是服务描述、选择和运行监控的重要部分。目前 Web Service 服务质量研究主要集中在服务实现层的运行性能保障,而对用户更加重视的应用层服务质量则没有完整的描述和处理方法。本文分析 Web Service 服务质量综合性、模糊性、动态性和可配置性等特点,建立基于可配置综合模糊评价指标体系的扩展的 Web Service 服务质量模型,并提出基于服务上下文的服务质量动态计算方法。

关键词 服务性能,服务描述,综合模糊评价,上下文,Web Service

Research on Extended Web Service QoS Model

MOU Yu-Jie CAO Jian ZHANG Shen-Sheng ZHANG Jian-Hong

(CIT Lab. Department of Computer Science & Technology, Shanghai Jiaotong University, Shanghai 200030)

Abstract Quality of Service(QoS) is a key factor in Web service advertising, choosing and runtime monitoring. Current researches on Web service QoS focus on service implementation level performance assurance, ignoring application level criterions which are also important to service users. Web service QoS is shown to be synthetic, fuzzy, dynamic and configurable. An extended Web service QoS model is set up based on configurable fuzzy synthetic evaluation system in this paper. Web service QoS is evaluated dynamically according to the service context.

Keywords Quality of service, Web service description, Fuzzy synthetic evaluation system, Context, Web service

1 引言

Web Service 提供了一种有效促进 B2B 电子商务的手段。各组织可以将自己的业务过程作为对外提供的服务发布,令潜在的客户可以发现和使用。相比传统的较为固定的业务合作伙伴关系,在这种临时性的业务伙伴关系中双方都获得了更大的效益:服务使用方从更大范围内灵活地选择到了最适合业务需求的服务,从而以最佳的效果达到了业务目标;而服务提供方则将服务提供给了更大范围内的更多客户,从而获得了更多的利润。基于这样的共识,越来越多的组织开始考虑,或者已经开始将业务过程作为服务发布到网络上来^[1]。

随着越来越多的组织将可以提供的功能发布到网络上,针对同一功能需求用户可以选择的服务种类越来越多。相似服务的出现一方面可能是同行业竞争的结果,如网络上可能会有多个相互竞争的预定机票服务;同时,同一家组织为了最大限度满足不同类型用户的需求,也可能配置出按照不同方式实现相同功能,从而具有不同性能指标的系列服务。多个功能相似服务的共存使服务质量(Quality of Service, QoS)成为这些服务之间最显著的区别和用户选择的依据。QoS 描述和计算评价在 Web Service 描述、选择和使用过程中占据的地位会越来越显著。

Web Service 的 QoS 的主要用途在于^[2]:

- 服务描述。作为 Web Service 非功能性特征的描述, QoS 体现一个 Web Service 功能完成的性能指标,是服务提

供方业务逻辑优点的外在体现和相互竞争的利器,对 QoS 的描述是服务描述的重要组成部分。

- 服务查找选择。面对多个可以满足同样需求的 Web Service,用户更加倾向于从 QoS 出发作出选择。

- 服务监控。QoS 可以看作服务提供方与客户之间的一种协议。在服务运行期间和结束后计算 QoS,可以评定服务是否按照预定义的质量指标协议完成,并根据完成情况进行计费或惩罚。

- 服务改进。服务提供方可以通过对 QoS 进行统计和分析,对服务进行改进,以提高各项 QoS 指标,在不断变化的环境中提供更加符合客户需求的服务以赢得竞争。

目前的 QoS 研究、讨论和定义集中于服务实现层的性能指标,如响应时间、单位时间吞吐量等。从技术角度出发,这些指标是实现稳定可靠 Web Service 运行、调用和复合的重要保障,但是作为一般用户,其对 Web Service 的评价和选择更可能从业务目标实现性能出发,即更关心应用层的服务质量指标。同时,当前 Web Service 描述模型中 QoS 未得到足够重视,其描述采用属性名-值对集合这种简单的平面结构,而计算方式采用静态的统计方式。本文提出当前服务描述语言和语义模型中对服务质量平面的、静态的和简单的描述不能够满足 Web Service QoS 描述、选择、监控和改进的需求,主要原因在于^[1,3,4] Web Service QoS 是一个多侧面指标共同表征的层次化综合评价体系,且其指标取值存在模糊化、动态化特征,而基于不同出发点用户在 Web Service QoS 评价中

^{*} 基金项目:上海高校网络技术 E 研究院网络门户项目(编号:200305),上海市科委 2003 年重大科研攻关课题“信息网格及其典型应用研究”。
牟玉洁 博士研究生,主要研究领域为 workflow 系统、过程知识管理、动态过程管理等;曹健 博士,副教授,主要研究领域为 workflow 技术、网络技术、企业建模技术和软件工程等;张申生 博士,教授,博士生导师,主要研究领域为分布式计算、计算机集成技术、并行工程、知识工程等;张建宏 博士研究生,主要研究领域为 workflow 系统、软件代理技术。

对不同的指标关心的程度也不尽相同。这些特点在当前的服务描述语言和语义模型中都没有充分考虑,从而使服务描述、发布和选择等过程中都无法充分体现 QoS 的影响因素。

为解决以上问题,本文建立能够全面反映 Web Service 服务质量特点和应用需求的扩展的 Web Service QoS 模型,以及基于上下文的动态 QoS 计算评价方法。

本文第 2 节讨论 Web Service QoS 相关研究,提出其不足之处并与本文的方案做比较;第 3 节分析 Web Service QoS 综合性、模糊性、动态性和可配置性的特点,建立相应的扩展服务质量描述模型;第 4 节讨论基于上下文的 QoS 综合模糊评价指标体系计算方法,并展示了一个应用实例,最后是总结和展望。

2 相关研究

服务质量概念早期应用于网络服务、实时系统、中间件、多媒体信息服务等领域,用于衡量或配置服务的性能特征^[5,11]。Web Service 兴起之后,研究人员将服务质量概念引进用于 Web Service 非功能性信息的描述。IBM 的研究人员提出了 Web Service 服务质量的六个主要指标:可见性(Availability)、可访问性(Accessibility)、完整性(Integrity)、吞吐量和响应时间(Performance)、可靠性(Reliability)、标准符合程度(Regulatory)和安全性(Security)^[6]。从这个定义看,Web Service 服务质量的定义还局限在软件执行性能层面上的描述。

随着 Web Service 的发展,研究人员已经注意到包括 Web Service 在内的各种服务的 QoS 包括多侧面复杂结构且具有动态性。来自 workflow 研究领域的研究已经开始将流程 QoS 表达和计算方法应用于 Web Service 的 QoS 建模及复合服务的服务质量计算。文[ardoso 2004]建立了一个包括成本、时间和可靠性三个主要侧面的服务质量模型,利用基于历史流程执行统计信息预测单个活动的 QoS,并通过特定的归约算法预测整个流程的 QoS。关于流程 QoS 的作用和描述、使用在 CrossFlow 项目中也有研究[handrasekaran 2002, Ralph Busse 2000],然而这些研究建立的服务质量模型仍然局限在流程管理领域的通用服务质量指标,并没有建立包括特定应用领域服务质量指标在内的全面综合的 QoS 模型。而且 QoS 指标计算模型基于对历史数据的统计及流程模型本身结构的分析,并未考虑如业务目标、资源负荷等上下文对服务 QoS 产生的影响。

服务的描述框架和语言在 Web Service 技术出现后成为研究的重点,先后出现了多种服务描述语言。Web Service 描述语言 WSDL 描述了服务可以接受和发出的消息格式,以及可以被访问的 port 信息,其对服务的描述主要侧重调用方式,不包括任何服务质量描述信息^[7]。后续出现的服务描述语言 DAML-S 及 OWL-S 从语义出发对 Web Service 的三个基本内容进行描述:用于描述服务功能、适用范围等一般信息和 IOPE 语义的 profile,用于描述服务流程和调用步骤的 Service Process,以及用于描述服务数据交换协议的 Grounding。其中仅在 Service Profile 部分中以名-值对的形式提供对服务质量的描述,且仅限于服务可靠性、响应时间或某种评定标准下的服务质量等级信息等静态属性的简单集合,另一些服务质量指标,尤其是领域相关的指标,如货品价格、运输时间等,则散落在其他的服务描述语义中,而无法构成一个统一的服务质量指标体系用于综合 QoS 计算^[8,9]。这种简单的

服务质量数据对服务选择过程提供的参考价值极其有限^[1]。在目前基于 Web Service 语义模型的服务描述和查找等研究中,也没有体现出 QoS 的重要影响作用^[10~13]。这主要是因为当前 Web Service 的相关研究目标还处于“找到”阶段,而在将来 Web Service 数量激增、相互竞争激烈的情况下,QoS 必将作用于服务描述查找和使用的各阶段。

以上分析表明,目前 Web Service QoS 描述和计算方面的研究无法体现其综合、多样、动态等特性。面向 QoS 的服务描述需要一个相应的 QoS 描述语义框架,用于描述包括通用服务质量指标和领域特有服务质量指标在内的综合服务质量指标体系;同时,QoS 的计算评价还需要考虑到 QoS 在不同上下文中的动态特性,以及不同用户对 QoS 评价的不同侧重点等因素。本文的主要目标即建立这样一个 Web Service 的 QoS 描述模型及其相应计算方法。

3 Web Service QoS 分析和扩展的服务质量模型

本节通过分析 Web Service 服务质量的特点建立其服务质量模型。

3.1 Web Service QoS 特点

3.1.1 综合

服务质量是对服务非功能性属性的描述,用于描述服务完成业务目标的效果,它是一个包含多侧面指标的复杂结构。本文认为,为了在服务选择过程中提供足够的参考信息,服务质量描述需要包括以下几个方面的内容^[1,2]:

- 成本(cost)。描述服务使用方需要为使用服务支付的价格。服务成本可能由多部分组成,因此整体成本是多个子指标按照整体-部分关系形成的树形结构。成本指标不仅是服务使用方选择时的重要依据,服务提供方也利用该指标进行服务配置和监控。

- 时间(time)。描述服务从开始到结束所需要的时间。不同类型的服务,其时间指标可能有很大差别,如一般的天气预报服务仅消耗服务响应时间,而货物运输服务则可能需要几个小时甚至几个月。除了服务选择阶段的用途,在服务运行期间使用方和提供方都可以利用时间指标监控服务的进度。

- 运行性能(Operation Performance)。包括可见性、可访问性、完整性、吞吐量、可靠性、标准符合程度和安全等级等服务运行性能描述。运行性能在为了服务复合而进行的服务选择中起到重要的参考作用。

- 一般性能(General Performance)。是服务目录(如 UDDI 服务器)维护的统计性服务质量指标,如服务被调用次数、成功次数、平均用户评价等。这些由客户做出或目录服务统计的客观性服务质量描述在服务选择中起到一定的参考作用。

- 领域相关性能(Domain QoS)。是服务所属的分类(或应用领域)所特有的服务质量指标,功能上相似的服务拥有相似的领域相关性能指标集合。如天气预报类服务有“预报准确率”指标,而股票信息类服务则有“信息刷新间隔”指标。服务应当具有的领域质量指标是由该服务所属领域的领域本体决定的。

从上面的分析可以看出,QoS 由多侧面指标构成。每个指标又可能是多个子指标共同构成的,因此服务的 QoS 体现为一个树形综合指标体系。从提供服务选择和优化等角度出发,用户更加关心的是综合指标体系所体现出的综合性能,而

不是单个指标取值。

3.1.2 模糊

Web Service QoS 不同指标具有不同的评价方式。有些指标可以量化,如服务可访问性(在某个时间通过服务请求成功初始化一个服务的概率^[6])为 98%。但是另外一些指标则可能表示为一个可浮动的区间估计,如国内货运服务到达时间为 2~3 天。还有的质量指标采用某种评定等级表达,如软件外包服务,其软件开发过程按照软件能力成熟度划分为 5 个等级。服务质量的描述需要允许多种表达方式和一定的模糊性。本文提出的 QoS 指标分为以下几种表示方式^[4]:

- 数值型。可以是整数或浮点数,是最简单的指标值表达方式。

- 区间型。表达为 [Left, Right], 是由两个数字组成的取值区间,表明该质量指标可能取 Left 和 Right 之间(包括 Left 和 Right)的任何数值。

- 语言型。用一个有限的评价性词语集合中的元素表达该质量指标。如服务用户反馈评价为“优秀”、“一般”、“差”三种描述。各等级之间划分可能会出现模糊和相互交叉的情况,其取值在更大的意义上代表了一种概率。

- 等级型。也是用一个有限的评价性词语集合中的元素表达质量指标,不同的是等级型指标拥有明确的客观评定标准且等级之间是清晰离散的。如软件外包服务中的软件能力成熟度(CMM)等级就是等级型质量指标的例子。

模糊表达类型的指标值在计算综合服务质量的时候需要通过去模糊算法(见第 4.2 节)转换成为对应的数值。各种质量指标又分别属于两种类型:成本型指标,其对应的数值越小越好,如过程执行时间即属于成本型指标;效益型指标,其对应的数值则越大越好,如 CMM 等级越高说明软件开发过程越规范。

模糊表达类型的评价值可以从各自的定义集合中选择,模糊表达类型及其可选值集合可以单独定义并用于多个服务 QoS 的定义。

3.1.3 动态性

Web Service 的服务性能指标的取值不是固定不变的,可能随服务用户的需求、服务提供方的状态等上下文产生变化。按照不同的取值方式可以将服务性能指标分为以下三类:

- 固定型。由服务提供方设定,必要的时候由服务提供方进行更新。用于描述服务提供方的业务策略的 QoS 指标,如天气预报服务的提前时间指标即此类指标的例子。

- 统计型。由服务提供方或中介服务根据服务运行历史统计获得。例子有服务调用次数、平均执行时间等指标。

- 计算型。这种类型的服务质量指标与具体的服务运行环境及服务目标等上下文之间有固定的函数依赖关系。例如,物流货运服务会按照单价×里程×货物体积(或质量)的计算公式确定货物运价。依赖函数通常是一种基于经验和统计的预测性函数,用于在服务执行之前进行指标预测,随着业务历史数据的积累可以不断对依赖函数进行修正,以提高其预测的准确性。

3.1.4 可配置性

Web Service 的 QoS 本身是一个多指标构成的综合评价体系,然而对该综合评价指标体系进行评价计算时,不同用户从不同使用角度出发可能关心的指标集合及各指标所占的权重也有所不同。以网上订餐服务的选择为例,如若当前时间为中午 11:00~13:00 之间,则客户对服务 QoS 指标中的

“送餐所需时间”指标权重设置较高,因为该段时间是各餐厅送餐繁忙时段。而若当前时间为 17:30~22:00,客户可能对服务质量中的“顾客评价等级”一项权重设置较高,而“送餐所需时间”指标则权重很低甚至不在考虑之列,因为晚间可以有比较充裕的时间在餐馆享受比较正式的晚餐。从此可以看出,QoS 需要可以根据上下文信息进行动态配置。用户在配置 QoS 的时候,对各指标的权重也可能采用模糊的方式。

我们将 QoS 配置过程定义为在服务 QoS 指标集合中选择部分指标,并为这些指标赋予不同权重的过程。配置之后的 QoS 指标体系称为一个 QoS 配置。QoS 配置过程中需要符合以下规则:

- 所有被选择指标都必须来自该 Web Service 的 QoS 指标集合,必须为每个被选指标定义权重。

- 若选择了某个 QoS 指标,则该节点的所有上层节点都必须选择。系统可以自动实施该规则,以维持指标之间的层次关系。

- 若某非原子指标节点被选择,且此节点没有自身节点值,则其子节点集合中至少有一个节点需要被选择。这是为了在综合指标计算过程中保证每个指标节点都能够计算评价值。

- 同一个非原子指标节点的所有子节点的权重必须来自同一个权重取值模糊集合,以便在综合 QoS 计算时可以进行权重的归一化计算(见 4.2 节综合模糊评价计算过程)。

QoS 配置过程需要花费一定时间,且用户在不同情况下可能选用不同的 QoS 配置,因此可以预先定义一系列 QoS 配置,并定义基于上下文的 QoS 配置选择规则。参考文[14]中提出的用于表达基于用户上下文的服务选择规则的语法可以加以修改后用于定义这种基于上下文的 QoS 配置规则。

3.2 Web Service 服务质量模型

根据以上分析建立 Web Service 服务质量模型如下:

Service 的服务质量 QoS 是由多个 QoS 指标 QoSItem 构成的树状结构综合指标体系 $QoS ::= \langle QoSItemSet, QoSRoot, RSet \rangle$, 其中 QoSItemSet 是一个质量指标 QoSItem 集合, QoSRoot 即该服务的综合服务质量指标,是树状结构的根节点, RSet 是各指标之间的构成关系 Part of 集合, Part of (QoSItem1, QoSItem2) 标明质量指标 QoSItem1 是质量指标 QoSItem2 对应的节点的一个子节点。根节点 QoSRoot 有五个分支节点: cost, time, OperationPerformance, GeneralPerformance 和 DomainQoS。

服务质量指标 QoSItem ::= $\langle ItemName, HasNodeValue, ValueInfo, NodeWeight \rangle$, 分别代表该服务质量指标的名称、是否有指标值、指标取值信息以及指标权重。设置 HasNodeValue 是由于在 QoS 综合评价指标体系中的非原子指标如总成本 Cost 具有有意义的指标值,而另一些综合性指标如综合运行性能仅能够通过其子指标的取值反映出来(此时是该综合指标的评价值),而不具有有意义的独立指标值。在这里,我们规定所有的原子指标节点都必须有指标值。

ValueInfo 仅对 HasNodeValue = True 的 QoS 指标有意义,用于定义指标取值类型和取值方法: ValueInfo ::= $\langle ValueType, PreferredValueType, EvaluateMethod, EstimateFunction, NodeValue \rangle$, 分别代表取值类型(数值型、区间型、语言型、等级型)、价值取向(效益型、成本型)、指标取值方式(固定型、统计型、计算型)、指标估算函数(仅计算型指标具有此属性)、指标值。此处我们以指标名称为唯一标识,因此各

服务质量指标之间不允许有重名。同时,基于同一个组织维护的 Web Service 采用统一的指标取值单位的假设,此处省略数值型指标的取值单位。

计算型 QoS 指标需要定义其指标值预测计算函数,该函数表达了指标取值与服务上下文之间的依赖关系,此处首先给出服务上下文定义:

ServiceContext 是一组用于定义服务查询或运行环境的上下文项目 ContextItem 的集合,ContextItem 有以下属性: Name, DataType, Value, Source, 分别代表该 ContextItem 的名称、数据类型、取值和来源。ContextItem 可以来自服务用户 consumer、服务提供方 provider,以及 QoS 计算所处系统自身的记录。其中来自服务用户的上下文又分为用户情况上下文 ConsumerStatus(包括用户所处位置、用户帐号等信息,这些上下文由 QoS 计算所在系统(如中介服务)自行采集),以及用户需求上下文 ConsumerRequest(包括服务持续时间、购买服务或货品的数量等信息),此上下文信息在用户服务请求中提交。来自 Provider 的 ContextItem 用于计算型 QoSItem 的取值,例子有 Provider 某种关键资源的当前负荷比例、某种货品的当前存仓量等。系统自身提供的 ContextItem 包括当前时间等信息。ServiceContext 是独立于 QoS 而单独存在的服务相关信息定义,多个服务 QoS 定义可以引用同一个 ServiceContext 信息。

```

(QoS)
  (QoSItem ItemName="QoSRoot" hasNodeValue="false" NodeWeight="重要")
    (SubItems)
      (QoSItem ItemName="Cost" hasNodeValue="true" NodeWeight="重要")
        (ValueInfo)
          (ValueType)numerical</ValueType>
          (PreferredValueType) COST </PreferredValueType>
          (EvaluateMethod) CALCULATE </EvaluateMethod>
          (EstimateFunction)
            (ContextItem Name="ProductID" DataType="string" Source="ConsumerRequest"/>)
            (ContextItem Name="Amount" DataType="integer" Source="ConsumerRequest"/>)
            (ContextItem Name="CustomerID" DataType="string" Source="ConsumerStatus"/>)
            (Function) GetPrice (ProductID, Amount, CustomerID)</Function>
          </EstimateFunction>
          (NodeValue/)
        </ValueInfo>
      </QoSItem>
      (QoSItem ItemName="Time" hasNodeValue="true" NodeWeight="一般")
        (ValueInfo)
          (ValueType)region</ValueType>
          (PreferredValueType) COST </PreferredValueType>
          (EvaluateMethod) SET</EvaluateMethod>
          (NodeValue)
            (1Bound)2</1Bound>
            (uBound)3</uBound>
          </NodeValue>
        </ValueInfo>
      </QoSItem>
      (QoSItem ItemName="GeneralPerformance" hasNodeValue="false" NodeWeight="重要")
        (SubItems)
          (QoSItem ItemName="AverageCustomerEvaluation" hasNodeValue="true" NodeWeight="重要")
            (ValueInfo)
              (ValueType)Grade</ValueType>
              (PreferredValueType) BENEFIT </PreferredValueType>
              (EvaluateMethod) STATISTICAL </EvaluateMethod>
              (Source)agency</Source>
            </ValueInfo>
          </QoSItem>
        </SubItems>
      </QoSItem>
    </SubItems>
  </QoSItem>
</QoS>

```

```

</QoSItem>
</SubItems>
</QoSItem>
</SubItems>
</QoSItem>
</QoS>

```

图 1 QoS 描述的例子

指标预测计算函数 EstimateFunction 有两种定义方式: 1) 以一个或多个 ContextItem 为参数的计算公式, 该计算公式直接定义在 QoS 信息中; 2) 以一个或多个 ContextItem 为参数的外部函数调用, 此时 QoS 定义中仅包含该函数调用的入口信息。

服务质量描述是服务语义描述的一部分, 可以采用 XML 语言实现。图 1 是一个网上售货服务质量描述的例子, 其中用户选择了成本、时间和平均用户评价三个指标(出于篇幅限制, 这里仅给出了简单定义)。

4 Web Service QoS 计算

4.1 服务 QoS 评价计算方法

本节以 Web Service QoS 评价和应用的典型场景: 基于 QoS 的 Web Service 选择为例讨论 Web Service QoS 评价计算过程。在这种场景下, 服务提供方中介(如 UDDI 服务)注册服务信息时, 需要按照中介服务规定的格式和内容在其服务描述中包含 QoS 信息。服务用户提出服务查找请求时同时包括 QoS 配置定义, 用于中介服务从多个功能满足的待选 Web Service 之间按照客户关心的指标和权重进行评价并推荐最合适的服务。整个过程包括服务 QoS 描述、服务 QoS 配置、QoS 指标赋值和综合 QoS 评价指标体系计算四个步骤, 详细说明如下:

1) 服务提供方在服务描述中定义 QoS 描述并注册到中介服务。中介服务为注册在其上的每个服务类型定义其综合评价指标体系, 包括通用指标及服务所属领域本体中定义的专用指标。用于指标取值和定义权重的模糊表示取值集合, 以及上下文定义信息也由中介服务定义和维护。注册到中介服务的 Web Service 必须采用符合该中介服务定义的综合评价指标体系定义及相应模糊表示方法和上下文定义。值得注意的是, 在由中介服务进行综合 QoS 评价计算的情况下, 每个 HasNodeValue=True 的指标, 其定义需要一个额外的属性 Source, 用于定义此指标取值的来源(agency/provider), 其中设定型指标的 Source="provider"。对于统计型指标, 若 Source 为 "agency", 说明该指标是中介服务自行统计的 QoS 指标(例如用户反馈评价等级), 而 Source 为 "provider" 则说明该指标由服务提供方自行统计, 通过特别定义的接口接受中介方查询。计算型指标的 source 属性定义了其计算发生的位置; 若为 "agency", 则服务提供方需要提供特殊接口, 用于中介服务获取其用于计算指标值的上下文信息; 若为 "provider", 则服务方需要提供接口用于中介方向其传递客户上下文信息并获取计算后的指标值。服务提供方可以通过一个特殊的 Web Service, 专门用于处理来自中介服务的 QoS 查询和计算请求。

2) 服务使用方进行 QoS 配置。用户向中介服务发出服务查询请求时, 中介服务按照服务类型向用户提供自身维护的服务 QoS 综合评价指标体系, 供用户选择指标进行 QoS 配置。用户可以定义多个 QoS 配置及基于上下文的 QoS 配置选择规则, 存储在中介服务方, 由中介服务根据探测到的用户

上下文自动选择合适的 QoS 配置,从而省却每次的 QoS 配置过程。

3) 针对每个功能、调用接口等描述符合服务请求定义的待选 Web Service, 中介服务根据当前 QoS 配置中包含的 QoSItem 依次察看其取值类型并赋值。其中, 设定型指标值直接从服务描述信息中提取; 统计型指标值按照其来源分别从中介服务统计信息中获取或通过预定义接口从服务提供方查询; 计算型指标按照其计算位置, 分别由中介服务从自行提取计算所需上下文信息进行计算或将必需的上下文信息传递至服务提供方进行计算。

4) QoS 综合评价计算。经过配置和指标赋值之后, 中介服务就可以对各指标节点计算其评价价值, 并利用加权和逐级进行规约计算, 最终获得唯一的综合模糊评价指标体系评价价值。中介服务可以按照综合评价价值对待选服务进行排序后提供给用户, 或仅提供评价价值最优的服务的信息。

4.2 模糊综合评价方法

由于综合评价指标体系中各指标的取值和权重可能采用模糊表示, 综合评价计算需要相应的特殊算法。下面详细介绍基于三角模糊数的模糊综合评价指标体系计算方法^[4]。

综合评价的一般方法是首先根据评价目的建立一个综合的指标体系, 然后对末指标逐个赋指标值, 经过合成得到最后的综合评价价值。其中, 末指标既存在指标值, 又存在指标评价价值, 而且两者具有某种对应关系; 而对于非末指标, 就只有指标的评价价值。综合服务质量计算过程中, 根据采集到的末指标模糊评价价值由下至上合成计算整个服务质量评价体系的综合评价价值。

用户提出的服务质量综合评价指标体系中, 评价权重和评价价值具有语言型、等级型、区间型、数值型等多种模糊表示。为了便于后续的综合指标计算, 必须将多种模糊表示统一为某种特定的模糊表示。本文提出基于三角模糊数的模糊表示转化方法, 将多种模糊表示一致化为三角模糊数, 用于后续计算。

4.2.1 三角模糊数的模糊表示形式

在服务质量综合评价中, 各指标的取值和对服务选择决策的影响程度(权重)都有一定的模糊性。在对多种模糊表示进行一致化时, 不能用经典数学的精确方法来对待。用三角模糊数来表达模糊含义是一种简便有效的方法。模糊综合评价中, 为了方便加权求和, 评价权重和评价必须位于区间 $[0, 1]$ 之内。为了适应综合评价的需求, 本文定义适用于综合评价的三角模糊数如下:

定义 1 综合评价三角模糊数采用一三元组 (l, m, r) 来表示, 其中 m 为模糊数中值, l 和 r 分别为模糊数的左值和右值。且 l, m 和 r 关系满足 $0 \leq l \leq m \leq r \leq 1$ 。综合评价三角模糊数定义 $\alpha = m - l$ 为三角函数的左跨度, $\beta = r - m$ 为三角函数的右跨度, $L(u)$ 和 $R(u)$ 为模糊数的左、右隶属函数:

$$L(u) = \frac{u}{m-l} - \frac{l}{m-l}, R(u) = \frac{u}{m-r} - \frac{r}{m-r}.$$

4.2.2 多种模糊表示的一致化

模糊综合评价三角模糊数为多种模糊表示的一致化提供了理论基础。下面列出本文涉及到质量指标取值和权重的各种模糊表示一致化为三角模糊数的方法。

数值型: 设该指标可能的最大值和最小值分别为 u_{\max} 和 u_{\min} , $L = u_{\max} - u_{\min}$; 指标当前取值为 x 。对于成本型的指标, $u = (u_{\max} - x)/L$; 对于效益型的指标, $u = (x - u_{\min})/L$ 。该评

价指标对应的三角模糊数表示为 (u, u, u) , 即左值、右值和中值相等。

区间型: 设 $[\alpha, \beta]$ 为一个评价价值区间, 区间 $[l, r]$ 为该评价价值区间内的一个评价价值, 且满足 $0 \leq \alpha \leq l < r \leq \beta$ 。则该评价价值对应的三角模糊数为:

$$\text{效益型: } \left(\frac{l-\alpha}{\beta-\alpha}, \frac{l+r-2\alpha}{2 \times (\beta-\alpha)}, \frac{r-\alpha}{\beta-\alpha} \right)$$

$$\text{成本型: } \left(\frac{\beta-r}{\beta-\alpha}, \frac{2\beta-(l+r)}{2 \times (\beta-\alpha)}, \frac{\beta-l}{\beta-\alpha} \right).$$

语言型: 语言集 $I = \langle i_0, i_1, \dots, i_m \rangle$ 代表一组按照评价取向由劣到优依次排列的有序语言评价价值集合。假设 i_j 为该语言集的任意一个语言评价结果, 则该评价结果的三角模糊数表示为 $(\frac{j-1}{m}, \frac{j}{m}, \frac{j+1}{m})$, 其中, j 为该元素在集合中的计数值, m 为该集合中最后一个元素的计数值。

等级型: 有序等级值集合为 $G = \langle g_0, g_1, \dots, g_m \rangle$, G 中元素按照评价取向由劣到优依次排列。设 g_i 为该等级集中的一个等级评价价值, 则该评价价值的三角模糊表示为 $(\frac{i}{m}, \frac{i}{m}, \frac{i}{m})$ 。由转化规则可知, 等级型模糊表示集合中等级值转化为三角模糊数后, 其左值、右值和中值相等。

4.2.3 综合模糊评价计算方法

上面所述的模糊表示一致化算法, 可以获得待选服务的 QoS 综合评价指标体系中所有叶指标结点的三角模糊数评价价值, 接下来的任务是利用这些评价价值获得服务质量综合评价价值。评价体系非叶节点的评价价值是其所有子节点评价价值的加权和, 其中子节点的评价价值是由不同类型模糊表示方法一致化得到的三角模糊数。节点的权重在 QoS 评价指标体系建立的时候也是采用模糊表达方式定义的, 一致化之后也是一个三角模糊数。但是在计算父节点评价价值的时候, 各子节点的权重需要做去模糊化和归一化, 计算方法为:

三角模糊数去模糊化算法 设 $F = (\alpha, m, \beta)$ 为任意三角模糊数, 其非模糊化评价价值为: $l(F) = \alpha + [(\beta - \alpha) + (m - \alpha)]/3$ 。

设某节点的所有子节点的权重 $(fw_1, fw_2, \dots, fw_n)$ 去模糊化后得到的 (w_1, w_2, \dots, w_n) 之间需要进行归一化: $W_i = w_i / \sum_{j=1}^n w_j$, 以保证该节点的所有子节点权重之和等于 1。

非叶节点的评价价值也是一个三角模糊数, 其左、中、右值分别是所有子节点评价价值左、中、右三值的加权和。权重是从某种最初的模糊表示经过一致化、去模糊化和归一化之后的介于 $[0, 1]$ 之间的常数。下面定义用于计算三角模糊数加权值的另外两个算法。

定义 2 λ 为一个常数, (α, m, β) 为三角模糊数, 定义常数与三角模糊数的乘法如下:

$$\lambda(\alpha, m, \beta) = (\lambda\alpha, \lambda m, \lambda\beta)$$

定义 3 设 (α_1, m_1, β_1) , (α_2, m_2, β_2) 为两个三角模糊数, 定义两个三角模糊数的加法如下:

$$(\alpha_1, m_1, \beta_1) + (\alpha_2, m_2, \beta_2) = (\alpha_1 + \alpha_2, m_1 + m_2, \beta_1 + \beta_2)$$

按照以上定义的三角模糊数相关计算方法, 加上已经采集到的各叶节点指标值, 则可以按照下式自下而上逐级计算各指标的指标评价价值, 直至根结点, 得到综合评价结果: $I_{i,j}^1 = \sum_{l=1}^{m_l} (I_{i,j}^1)_{l,j} w_{j,l} \cdot I_{j,l}$ 。综合评价结果也是一个三角模糊数, 可以通过去模糊算法转换成具体的数值进行比较。

4.2.4 例子

假定用户定义的 QoS 配置如图 1 所示,当前中介服务查找到的符合功能需求的待选服务有 3 个,且其相应的 QoS 指标取值分别如表 1。

其中 3 个指标的权重来自语言型模糊表示集合{不重要,一般,较重要,重要,很重要},分别是:cost 很重要,time 一般,用户平均评价:重要。指标 Time 为区间型,选择区间为[0, 5]。平均用户评价等级为等级型表示,取值集合为 1~5 星。按照以上规则得到各指标节点的三角模糊数评价价值和权重分别如表 2 所示。计算得到三角模糊数表示的综合 QoS 评价价值分别为:(0.5625, 0.5875, 0.6125), (0.575, 0.6, 0.625), (0.3625, 0.425, 0.4875),去模糊后分别为:0.5875, 0.6, 0.425。即按照用户对各指标的关心程度,Web Service 2 最符合其需求。

值得注意的是,在综合评价指标计算过程中,原子节点的评价价值是根据其指标值一致化而来,而非原子节点的评价价值有两种来源:若该节点无节点值,其评价价值即为子节点评价价值的加权和;若该节点有节点值,则按照默认规则取该节点值一致化结果为节点评价价值,此时其子节点评价价值和权重带来的影响被忽略。用户仍然可以选择该节点,仍然按照子节点评价价值加权和的方式计算评价价值,此时节点本身的节点值被忽略。

表 1 待选服务 QoS 信息

项目	Cost	Time	AverageCustomerEvaluation
Web Service1	500	1-2days	☆☆☆
Web Service2	480	3-4days	☆☆
Web Service3	520	1-2days	☆☆☆☆

表 2 各指标节点评价价值和权重

项目	Cost	Time	AverageCustomer Evaluation
Web Service1	0.5, 0.5, 0.5	0.6, 0.7, 0.8	0.6, 0.6, 0.6
Web Service2	1, 1, 1	0.2, 0.3, 0.4	0.4, 0.4, 0.4
Web Service3	0, 0, 0	0.6, 0.7, 0.8	0.8, 0.8, 0.8
NodeWeight	0.5, 0.75, 1	0.25, 0.5, 0.75	0.5, 0.75, 1

总结 随着 Web 服务相关技术的发展,会有越来越多的组织将业务过程作为服务,通过网络对外发布。而功能相似的服务会在服务质量上展开激烈竞争,因此服务质量在 Web Service 的描述、选择、监控和改进中占据的地位会越来越重要。本文在分析 Web Service QoS 综合性、动态性、模糊性和可配置性等特征的基础上提出了基于综合模糊评价指标体系的扩展的服务质量模型,以及此模型基础上基于服务上下文的动态服务 QoS 计算评价算法。本文方案可以弥补当前相关研究和标准的不足,为面向 QoS 的服务描述和使用提供更加有效的支持。

下一步研究将侧重面向 QoS 的动态服务选择过程,基于

QoS 协约的服务运行监控以及基于 QoS 的服务过程改进等方向。

参 考 文 献

- Cardoso, Jorge, Sheth, et al. Quality of service for workflows and web service processes. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2004, 1(3): 281~308
- Busse R. Cooperative Support Service - Quality of Service. GMD, Zurich, 21/September/2000. Available at <http://www.crossflow.org/public/wsfoils/QoS.ppt> on 2005-2-22
- Chandrasekaran S, Silver G, et al. Service Technologies and their Synergy with Simulation. In: *Proceedings of the 2002 Winter Simulation Conference (WSC'02)*, San Diego, California. 2002. 606~615
- 刘英华. 可配置模糊综合评价模型研究与系统实现:[硕士学位论文]. 上海:上海交通大学计算机系, 2004
- GU Xiaohui, Nahrsted K, YUAN Wanghong, Wichadakul D, XU Dongyan. An XML-based Quality of Service Enabling Language for the Web. *Journal of Visual Languages and Computing*, 2002, 13: 61~95
- Mani A, Nagarajan A. Understanding quality of service for Web services: Improving the performance of your Web services, <http://www-128.ibm.com/developerworks/webservices/library/ws-quality.html>
- WSDL: Web Services Description Language. Version 2.0. Available at <http://www.w3.org/TR/2004/WD-wsdl20-20040803>
- DAML-S: DAML-S 0.7 Draft Release. Available at <http://www.daml.org/services/daml-s/0.7/> on 2005-2-23
- OWL-S: Semantic Markup for Web Services. W3C Member Submission, Available at <http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/>
- Paolucci M, Kawamura T, Payne T R, et al. Semantic Matching of Web Services Capabilities. In: *Proc. of the First Intl. Semantic Web Conference, 2002*
- Wombacher A, Fankhauser P, Mahleko B, et al. Matchmaking for Business Processes Based on Choreographies (Extended version of the EEE 2004 paper). In: *International Journal of Web Services*, 2004, 1(4): 14~32
- Gonzalez-Castillo J, Trastour D, Bartolini C. Description logics for matchmaking of services. In: *Proc. of the KI-2001 Workshop on Applications of Description Logics, 2001*
- Trastour D, Bartolini C, Gonzalez-Castillo J. A semantic web approach to service description for matchmaking of services. In: *Proc. of the Intl. Semantic Web Working Symposium (SWWS), 2001*
- Liu Hao, Han Yanbo, Li Gang, et al. Achieving Context Sensitivity of Service-Oriented Applications with the Business-End Programming Language VINCA. *GCC 2004*, 89~96