

# 一种面向服务的体系结构参考模型

麻志毅 陈泓婕

(北京大学信息科学技术学院软件工程研究所 北京 100871)

**摘 要** 在对当前的面向服务体系结构研究的基础上,提出了一种用于设计面向服务体系结构的参考模型,深入地阐述了有关概念,详述了参考模型的结构以及其中的服务总线和服务合约的元模型,并提出了一个用于评价面向服务体系结构的成熟度模型.该参考模型为进一步设计面向服务的体系结构奠定了基础.

**关键词** 面向服务的体系结构;体系结构参考模型;成熟度模型

中图法分类号 TP311

## A Service-Oriented Architecture Reference Model

MA Zhi-Yi CHEN Hong-Jie

(Institute of Software Engineering, School of Electronics Engineering & Computer Science, Peking University, Beijing 100871)

**Abstract** Recently, SOA plays a more and more important role in software research and software development. On the basis of the research findings on SOA, this paper presents a SOA reference model that can be used to design SOA. The paper deeply discusses some concepts on SOA, and expatiates on the structure of the reference model and its service bus and meta model for service contracts. Moreover, the paper presents the maturity model for evaluating service-oriented architectures. The reference model lays the foundation for building service-oriented architectures.

**Keywords** SOA; architecture reference model; maturity model

## 1 引 言

SOA(Service-Oriented Architecture)是一种追求敏捷性的面向服务的体系结构,它把业务逻辑和具体实现技术这二者分离开来,因而遵循该体系结构所构造出来的应用系统能适应业务和实现技术的不断变化.此外,遵循 SOA 还有利于软件复用和系统集成.

至今,有很多文献较为详细地论述了 SOA,有些成果已经被应用于工程实践,但仍有不少重要问

题尚未得到妥善解决. Krüger 等人在文献[1,2]中重点论述了如何用构件来实现服务,但没有讲述如何用服务来组合服务,对 QoS(Quality of Service)也没有进行论述. MAIS 项目<sup>①</sup>着重考虑了对针对通道的具有适应性的信息系统如何进行基于服务的建模,但没有对服务的质量和服务的组合机制给出进一步的论述.文献[3~5]较为全面地阐述了 SOA 的思想,提出了宏观模型,但没有进一步论述模型的细节.文献[6]从面向服务的工作流访问控制的角度论述了 SOA,没有考虑怎样组合服务以及怎样描述服务的合约. Endrei 等<sup>②</sup>较详细地论述了 SOA,

收稿日期:2006-02-13;修改稿收到日期:2006-05-16. 本课题得到国家“九七三”重点基础研究发展规划项目基金(2002CB312003)、国家科技攻关计划项目基金(2003BA904B02)和国家自然科学基金(60473064)资助. 麻志毅,男,1963年生,博士,副教授,主要研究方向为软件工程与软件工程环境、面向服务技术和面向对象技术等. E-mail: mzy@sei.pku.edu.cn. 陈泓婕,女,1971年生,博士,讲师,主要研究方向为需求工程和构件技术等.

① Adorni M., Arcelli F. *et al.* MAIS reference model. <http://black.elet.polimi.it/mais/index.php>.

② Endrei M. *et al.* Patterns: Service-oriented architecture and Web wervices. <http://www.redbooks.ibm.com/redbooks/pdfs/sg246303.pdf>.

并应用于实践,但所提出的 SOA 实质上只适用于 Web 技术.一些国际性的标准,如 BPEL4WS<sup>[7]</sup>和 WSCDL<sup>[8]</sup>等也是针对 Web 技术的.很多对服务的定义也是基于具体技术的,如 Web 服务或 Jini 中间件;在对术语的定义和对 SOA 的理解方面也还存在有一定的差异.

针对上述问题,本文提出了一个 SOA 参考模型.其中规定了 SOA 总线所应有的结构,并从组织结构方面描述了 SOA 中的组织实体以及它们之间的关系.此外,本文对 SOA 中的重要概念进行了定义,并阐述了服务的合约元模型和 SOA 成熟度模型.

## 2 服务与 SOA 的语义

### 2.1 服务

服务的特性一般可归纳如下:

(1) 软件实体. 服务作为软件实体,通常为粗粒度的,对特定的高层业务逻辑进行了封装,向用户提供有意义的功能.该软件实体应该是长期稳定的,可版本化.

(2) 具有合约. 合约规定了服务的提供者和使用者的职责,其中含有接口和服务质量以及相关的约束.这种合约是自包含的且抽象于对服务的实现.通过它可以分离服务提供者 and 使用者,使二者的实现独立地变化.

(3) 可互操作的. 服务是部署和运行时的概念,通常基于消息进行互操作.

(4) 可组装的. 这是指可由服务构造服务,作为成分的服务可能来自多方的提供者.组装后的服务向用户隐藏内部细节,只向外提供封装了一定业务逻辑的合约.服务的可组装性支持快速、低代价的对应用系统的构造.

(5) 可复用的. 一个服务可以在多个应用中使用,也可以用其组装多个其它服务.

(6) 可发布与可发现的. 服务的提供者通过服务登记处发布服务合约,以供使用者动态地发现和使用的.这也说明服务的使用者和提供者之间的关系是松耦合的.

综上所述,服务是具有合约的、有意义的软件实体,通常是粗粒度和自包含的,而且是可复用、可互操作、可组装、可发布和可动态发现的,通常基于消息与外部应用或其它服务进行交互.

### 2.2 SOA

现代社会中的业务组织为了快速且有效地适应业务变化,以保持竞争优势,需要一种具有敏捷性的

企业体系结构.IT 技术的快速发展也需要一种具有敏捷性的基础结构.SOA 即是一种追求敏捷性的面向服务的体系结构,通过它可以把业务变化与 IT 技术变化隔离开来.

我们认为 SOA 是一组策略、实践和框架.利用它所提供的基础设施,可跨网络地登记、动态发现、绑定与安全而可靠地调用基于合约的服务,能对服务进行组装,并通过利用与实现环境无关的消息协议使服务间能互操作;此外,它在运行服务时具有监控服务和处理例外的机制,提供对服务的质量和资源进行管理的基础设施.

## 3 SOA 参考模型

有些文献把 SOA 与具体标准、技术或其它实现细节关联起来,有的则不进行关联.为了予以区别,本文作者认为不与具体标准、技术或其它实现细节关联的 SOA 是一种 SOA 参考模型,而进行关联的 SOA 是 SOA 参考模型的一个应用.故本文作者认为 SOA 参考模型是一个用以指导建立具体的 SOA 的抽象框架,它描述了应用服务的环境设施以及其中的部件和部件间关系,且它不直接受任何具体标准、技术或其它实现细节的约束.这与 OASIS 当前的观点是一致的<sup>[9]</sup>.

### 3.1 SOA 参考模型的组织机构

针对典型的服务提供者—服务请求者—服务登记处的模式,现从组织机构角度提出如下的 SOA 参考模型(如图 1 所示).

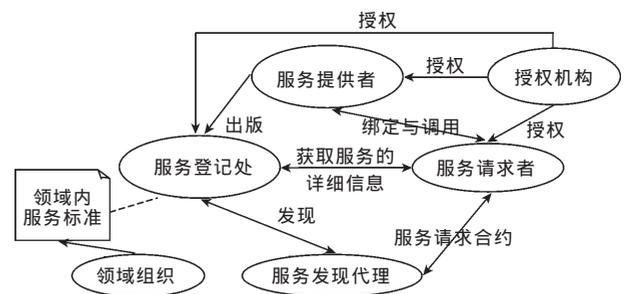


图 1 SOA 参考模型的组织机构

服务都需要在服务登记处按照服务登记合约进行登记.服务登记合约中除了服务合约外,还包括诸如授权、使用条件以及拥有者等信息.

服务提供者在授权机构取得授权后,向相应的服务登记处发布合乎质量要求的服务,以供服务请求者使用.

服务请求者在运行时可通过服务发现代理寻找服务登记处,以查找所需要的服务;在授权机构取得

授权后,在服务登记处获取进一步的信息,进而使用服务提供者所提供的服务。

服务登记处提供查找机制,也即对服务请求者所提出的服务请求,按服务合约等进行查找.若查找成功且在验证服务请求者具有相应的权限后,向其提供服务的地址等信息,服务提供者接到请求后按合约提供服务。

领域组织提供 SOA 标准,如对服务的合约如何描述等。

图 1 中的每个角色在逻辑上是独立的,在物理上一个组织机构也可以承担多个上述的角色。

还有多种体系结构风格可用于建立 SOA,如 ORB、P2P 和一些混合风格等,但采用这样的风格建立起来的 SOA 存在一定的问题<sup>[10]</sup>.即使要采用这样的风格建立 SOA,也要对上述参考模型进行适当的变种。

### 3.2 SOA 参考模型的层次性

业务组织应该进行业务过程优化,然后与 IT 组织一起对业务过程进行服务化.一个业务组织内的服务,一部分供自己内部使用,其余的供外部使用(对外发布).对于服务的实现,采用构件基础设施.图 2 描述了 SOA 参考模型的层次结构。

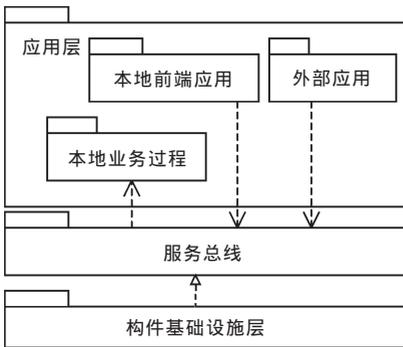


图 2 SOA 参考模型的层次性①

应用层中的本地前端应用以及外部应用都使用服务总线.一个业务过程由一些活动组成,一个基本活动对应一个服务,一个(子)业务过程对应一个组合服务.由于业务过程中的活动是由服务实现的,业务过程就变成了服务流。

服务总线是一种通过运行本地业务过程为服务请求者提供服务的设施,它分离了应用层和实现服务的构件基础设施层.本地应用或外部应用所请求的一个服务,实现了本地的一个业务过程,在其中也可能要请求其它的服务.被请求的服务可以是基本服务、由静态组合而产生的服务或通过动态组合而产生的服务。

在对服务的实现上,本文遵循大多数人的观点,

即应该用构件实现服务,且随同实现服务的构件一起部署服务.构件基础设施层提供实现服务所需要的各种环境和构件,负责用细粒度的构件实现粗粒度的服务.对于已经存在的应用(遗产系统),可包装成构件。

正是服务总线分离了与业务逻辑相关的应用层和与实现平台相关的构件基础设施,遵循该体系结构所构造出来的应用能适应业务和技术变化。

图 3 给出了 SOA 参考模型需要解决的问题。

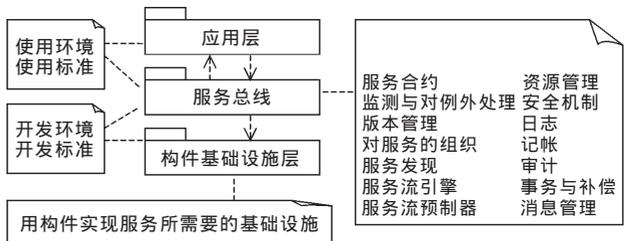


图 3 与 SOA 参考模型相关的问题

开发环境和开发标准分别是指开发服务和构件时所涉及到的各种开发环境和所要使用到的各种标准。

使用环境和标准分别是指进行面向服务的应用开发时,所需要的环境和所要使用到的各种标准。

用构件实现服务所需要的基础设施包括组装、部署和运行构件的设施以及对构件进行管理和对遗产系统进行包装等设施。

服务的合约是对服务在功能和质量等方面的规约;监控与对例外处理用于监视与分析各服务的执行情况,当服务出现例外时要及时处理;版本管理负责组织与存储各服务的版本以及对新版本的服务进行部署;对服务的组织是指按功能及 QoS 对服务进行划分,确定等价类,也即决定哪些服务是可以相互替代的;在服务总线上有本地服务发现代理,以决定到哪个(些)服务登记处发现所需要的服务;在服务流引擎上执行服务流,并要管理服务流的执行状态;服务流预制器用于根据业务过程构造服务流;在一组应用需要连接到关键的后端资源时,需要进行资源管理,在服务层确保各种操作访问资源的一致性;安全机制用于对服务的提供者 and 使用者进行授权/认证以及保证服务的消息的完整性和机密性等;事务与补偿用于保证数据的完整性;日志与审计主要用于恢复系统错误;由于要通过消息控制服务的执行,故对消息要进行管理,如对消息进行排队和同步

① 本图及以下各图(图 5 和图 6 除外)均采用 UML 表示法。

等. 在 3.4.3 节将进一步讨论上述内容的有关细节.

### 3.3 服务合约的元模型

至今还没有明显能作为一阶建模实体的一套建模概念用于面向服务的软件开发<sup>[11]</sup>. 本节要研究服务合约的元模型, 因为它给出了服务的应用语境, 这使得在 SOA 中服务间能够相互协作. 人们对服务的合约下了很多定义, 但都不够全面, 如很多仅捕获了客户能调用的操作的语法列表, 而没有给出交互模式或质量因素. 对服务的合约定义也多为非形式化的, 尽管有的采用了形式化定义, 也过于简略.

服务的合约是对请求者使用的服务和提供者提供的服务的元协调, 其中除了包括功能之外还包括质量描述和一些行为约束, 如图 4 所示.

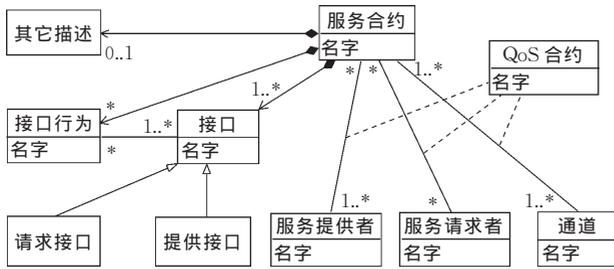


图 4 服务合约的元模型

一个服务合约具有一个或多个接口, 可具有一个或多个接口行为. 接口进一步地分为请求接口和提供接口. 一个服务合约必须具有至少一个提供接口, 可具有零个或多个请求接口. 各请求接口和提供接口分别由服务所需要和所提供的一组抽象操作组成.

接口行为描述当一个服务参与消息交换时该服务的对外可见的行为, 其中包括对操作的执行次序的约束以及对事务、补偿和对例外处理的策略的描述. 一个接口可受接口行为的约束, 一个接口行为约束一个或多个接口. 服务合约中可有多个接口行为, 用以支持服务行为的不同视图.

服务使用者、服务提供者和通道用于描述服务的使用环境.

服务与提供者间的 QoS 合约描述了提供者所保证的服务质量; 服务与请求者间的 QoS 合约描述了请求者所需要的服务质量; 服务与通道间的 QoS 合约描述了通道的响应时间、可用性、可访问性、带宽和可靠性等. 在 ISO/IEC<sup>[12~14]</sup> 中对 QoS 的内容做了具体的规定. MAIS 项目也确定了 225 个质量维. OMG 针对 QoS 对 UML 进行了扩展, 可用于对服务的 QoS 进行建模<sup>[15]</sup>.

由于服务也可以由其它服务组成, 这就存在着

计算组合服务质量的问题. 有关这方面的计算方法, 请参看文献<sup>[16,17]</sup>.

除了在静态结构、动态行为和 QoS 方面对服务合约进行描述外, 还要用其它描述对其它所需要的信息进行描述, 如使用服务的法规等.

### 3.4 服务总线

本节首先分析服务总线支持的服务的种类, 然后论述服务总线的结构和它的基础设施.

#### 3.4.1 服务的种类

按照服务所起的作用划分, 服务总线支持的服务分为基本服务、过程性服务、协调服务和公共服务.

基本服务实现相对完整的业务逻辑, 也即具有独立的功能且管理自己的数据, 是无状态的.

过程性服务主要实现过程控制和少量的业务逻辑, 在其中要调用其它服务. 复杂的过程性服务是可以嵌套的. 过程性服务要维持业务过程的状态.

协调服务用于协调设计上不一致的服务, 为其它服务提供多方面的视图以及向已有的服务附加功能等. 协调者服务是无状态的.

本地前端应用和外部应用不但可以直接使用基本服务和协调服务, 也可以直接使用过程性服务, 以提高执行的效率和服务的可靠性. 以请求者来看过程性服务应该是原子的<sup>[18]</sup>, 尽管对提供者来说一个对外提供的服务可由其它服务组合而成.

公共服务是向组织外部提供的服务, 这样的服务可分为公共基本服务和公共过程性服务. 公共服务需要在外部服务登记处进行登记. 相对而言, 仅由一个组织内部使用的服务, 称为内部服务, 这样的服务仅在组织内部的服务登记处进行登记.

图 5 给出了各种服务以及它们所位于的层次.

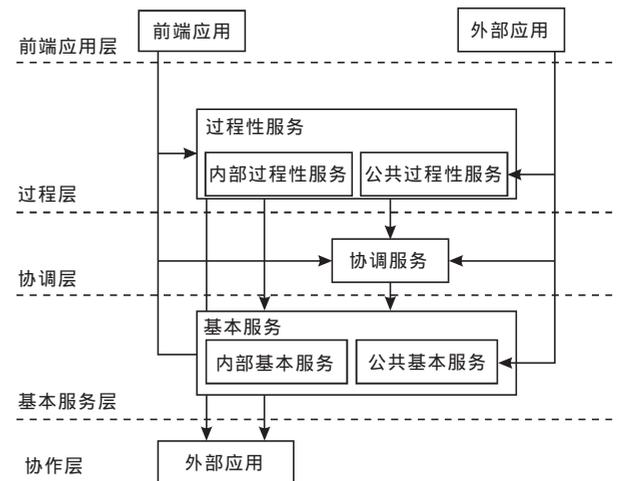


图 5 服务的种类以及它们所位于的层次

考虑到在分布系统中的通信效率以及存在着若干服务结合紧密的情况(例如,经过协调服务组合起来的服务),可以对服务进行预制,让这样的服务能够直接互操作,而不需要再经过在服务登记处进行查找.也即预制服务的提供者预先按业务过程逻辑确定其中的实现各个活动的服务,形成过程性服务.这样的服务被称为已预制的服务.

若不预先对过程性服务进行预制,在执行过程性服务时就要动态地查找业务过程中的结点所对应的服务.这样的服务被称为需编制的服务.

服务提供者可提供若干具有相同功能的预制服务,但各预制服务的 QoS 不同,以供请求者选择使用.这要求服务的提供者事先要尽可能全面地对服

务的功能与 QoS 的需要进行预测.对于需编制的服务,服务的提供者需要动态地计算所编制的服务的 QoS,以确定其是否满足请求者的要求.

### 3.4.2 服务总线的结构

服务总线是一种通过运行本地业务过程为服务请求者提供服务的设施,其结构如图 6 所示.

服务总线上有一个内部服务登记处和一个服务发现代理.内部服务登记处用于登记本地内部使用的服务和本地可供外部使用的服务(公共服务).服务发现代理先在内部服务登记处查找,若没有查找到,再到外部服务登记处查找;若查到的服务有多个选择,服务发现代理还要按服务合约进行排序.向外发布的服务还要登记到相应的外部服务登记处.

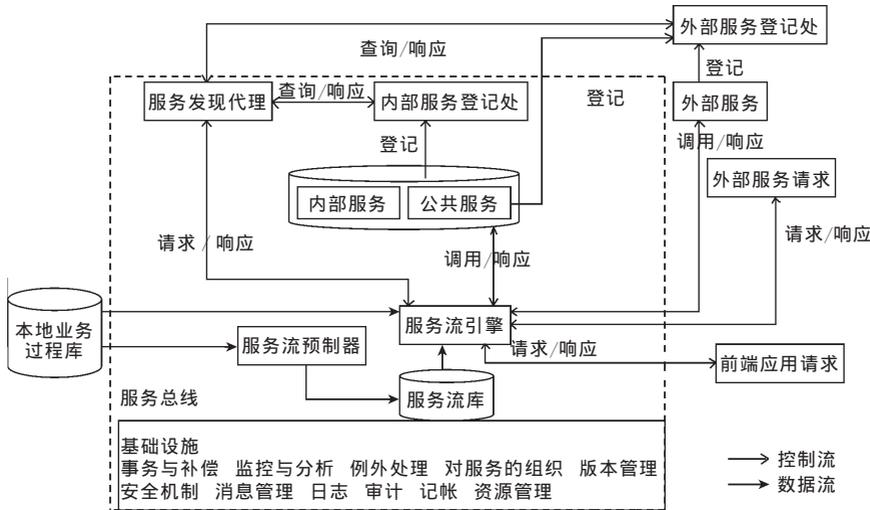


图 6 服务总线的结构

对于需预制的过程性服务,服务流预制器预先按业务过程逻辑,寻找相应的服务,形成服务流,放到服务流库中.

无论是外部应用还是本地前端应用请求服务,在进入服务总线后,都启动服务流引擎.服务流引擎判断当前所请求的服务是已预制的服务、基本服务还是需编制的服务.若是已预制的服务,则按照已经确定的服务流直接执行;若是基本服务,则直接执行它;若是需编制的服务,则需要查找和选择.若服务流引擎在执行一个服务时,该服务又调用了其它服务,也要把它提交给服务发现代理去查找和选择.若查找和选择的服务是内部的,则重复上述过程;否则调用外部服务.服务流引擎在执行一个服务时,要使用实现该服务的构件基础设施获得执行服务的结果.服务总线的主要活动流程如图 7 所示.

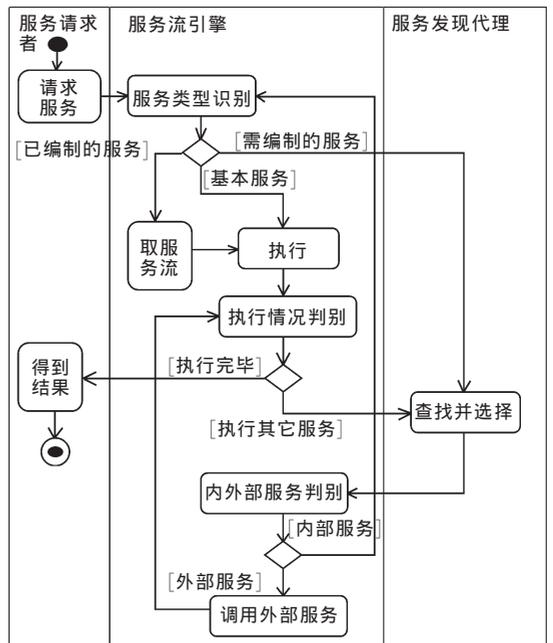


图 7 服务总线的主要活动流程

在上述过程中,若没有查找到所需要的服务,或执行服务的过程中出现故障,均会发生例外,在图中没有图示这种情况,对这种情况的处理请看 3.4.3 节。

服务流引擎不但要执行服务,还要管理服务流的状态,并要使用服务总线中的基础设施。

### 3.4.3 服务总线的基础设施

本节主要论述服务监测、例外处理、事务与补偿、日志和审计这几部分。至于资源管理、消息管理、对服务的组织、记帐和版本管理与安全机制,均可采用常见的处理方式,如采用相应的中间件或实现为专门的技术服务,本文对此不再进行论述。

#### 对服务的监测

服务总线提供对服务的执行情况进行监测的机制。它要履行两个主要功能:(1)分析服务的运行状况;(2)监测服务异常。

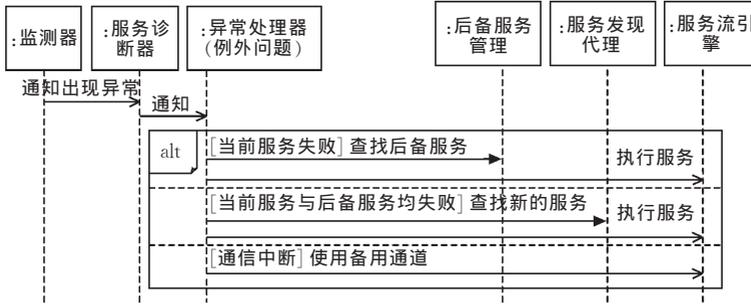


图 8 对例外的处理

在运行时,当监视器发现当前服务性能下降时,使用后备服务或通知服务代理发现另一个性能优良的服务代替它,处理方法与上述类似。

对于服务失败的情况,要按已定义的事务回滚到执行其前的状态,或进行补偿。

上述仅讨论了对服务在执行时出现的典型的异常情况进行处理的方法,对于一般例外的处理方法,请参看文献[19,20]。

#### 事务与补偿

若服务执行失败,系统要回滚到事务执行前存在的状态或进行补偿。对于长的事务,若有可能,要回滚到以前的适当位置或就从失败点恢复执行,并采取补偿措施。恢复执行时,理想的情况是不中断用户。在服务流引擎中对事务进行处理的具体思想<sup>①</sup>参见文献[21,22]。

针对服务支持业务过程逻辑的特点,当前这方面的技术弱化了 ACID 事务保证,强调用业务逻辑判断事务的成功与否,而不是用事务的是否全部完成来判断,且不主张采用回滚技术<sup>[23]</sup>。

对服务运行状况的分析,要统计服务的平均、最大和最小运行时间、出错情况以及对资源的使用等,还要监视服务的实时性能。

不存在过程性服务中所需要的服务、通信中断、服务出现故障或服务的性能严重下降等均属于系统异常。如果出现这样的情况,就要对例外进行处理。

#### 对例外的处理

如下是处理例外的一种方案:

(1)若所需要的服务不能执行或 QoS 达不到要求,则要使用备用服务。

(2)若当前与后备服务均失败,则要重新发现服务;若不能发现所需要的服务,系统失败。

(3)若属于通信中断,使用服务的备用通道;若不成功,系统失败。

图 8 描述了上述解决方案(2)和(3)中各方之间的交互情况。

#### 日志和审计

对例外的处理也可以通过日志和审计采取一定的措施来解决。对于分布式应用,局部日志应该是全局可见的,这就要求针对可能具有不同格式的局部日志要建立统一的接口。通过审计相关的日志决定采取解决例外的措施。

## 4 SOA 的成熟度模型

如何评价 SOA 的成熟度,目前有着不同的观点。很多 SOA 成熟度模型仿照 CMM 定义了建立 SOA 的指南和过程,如文献[24,25]<sup>②③</sup>按照企业实施 SOA 的过程定义了 SOA 成熟度模型。Gervas Douglas 认为 SOA 成熟度模型用于测量企业体系

① Procházka M. . OMG activity service. <http://jotm.objectweb.org/doc/ActivityService020610.pdf>.

② Mittal K. . Build your SOA Part 2: The service-oriented architecture model. <http://www.ibm.com/developerworks/webservices/library/ws-soa-method2/>, Nov 2005.

③ Groves D. . Successfully planning for SOA. <http://dev2dev.bea.com/pub/a/2005/11/planning-for-soa.html>, Sep 2005.

结构的成熟度<sup>①</sup>,但其中也把经济和企业实施 SOA 的过程作为考虑的因素;在他提出的 SOA 成熟度模型中去掉这两个因素,但其中又涉及到了具体的实现服务的技术且模型较为简单<sup>②</sup>.

对本文所提出的参考模型可根据需要进行裁剪,然后用以设计具体的 SOA. 本文根据具体的 SOA 所实现的参考模型的程度,对它进行分级,以

表明它属于哪个成熟度,也即它所能提供的能力. 为此,本文要从所支持的标准、基础设施和服务的种类三个方面构建 SOA 成熟度模型. 成熟度模型采用增量定义方式,即定义在第  $n$  级成熟度中的特征和能力,会自动包含在  $n+1$  级成熟度中.

在有关 SOA 成熟度模型的基础上,本文提出 SOA 成熟度模型,如表 1 所示.

表 1 SOA 成熟度模型

SOA 成熟度级别	特征	能力
1. 初始级	标准: 部门内部的服务合约 基础设施: 服务流引擎, 监测与对例外处理, 版本管理, 资源管理, 简单的事务与补偿处理, 日志与审计, 消息管理服务种类: 基本服务	在组织内部的前端应用使用基本服务; 服务间也可直接交互;
2. 内部协作级	标准: 部门间的服务合约 基础设施: 服务登记处, 对服务的组织, 复杂的事务与补偿处理 服务种类: 协调服务, 已预制的服务	在组织之间使用协调服务和已预制的服务; 通过动态发现进行服务间交互; 对服务的执行情况进行监控; 可通过协调服务使用基本服务;
3. 组织级	标准: 组织内部的服务合约 基础设施: 服务授权机构与权限识别, 服务流编制器; 服务种类: 需编制的服务	在组织内部有组织地使用服务; 可动态地生成服务流;
4. 组织协作级	标准: 组织间的服务合约 基础设施: 服务发现代理, 嵌套事务、服务运行情况的分析 服务种类: 公共服务	在组织之间有组织地使用服务;
5. 领域级	标准: 领域范围的服务合约 基础设施: 成熟的领域组织 服务种类: 成熟的公共服务	在领域内均支持标准的服务合约, 进行全面的 服务协作.

BEA 等<sup>[26]</sup>所提出的技术,能够构造达到第 4 级成熟度(组织协作级)的 SOA. SBB 2.0 是一个有着数百万用户的基于 SOA 实现的邮政系统<sup>[27]</sup>,它到达了成熟度的第 2 级;按照它的下一个版本的技术指标,它将要达到成熟度的第 3 级. CSG 是一个全球性的基于 SOA 实现的信用系统<sup>③</sup>,按照 2003 年发布的该系统指标,它的成熟度达到了第 4 级.

## 5 相关的研究及评价

OASIS 当前正从事 SOA 参考模型的标准化工作,其目标是定义最小的一组 SOA 核心概念,并确定其间的关系,用于提供构造具体 SOA 的共同语义<sup>[9]</sup>. 该参考模型只是处于概念定义与概念关联的层次,主要用于提供对 SOA 进行共同的理解,没有阐述如何运用这些概念深入地构造 SOA 的基本结构(如服务总线和组织机构). 针对目前还没有公认的企业服务总线的状况,CBDI 提出了一个 SOA 参考模型,其中给出了企业服务总线的主要构成模块,但偏重于如何进行实现,并主张它要全面地支持 Web 服务<sup>[28]</sup>,这使得在用其它技术建立 SOA 时该参考模型的能力受限.

文献<sup>[29,30]</sup>结合自己的产品从表示、业务过程、服务、构件和操作平台这五个层次描述了一个 SOA 参考模型,并在安全、管理、QoS 和监控这几个方面规定了服务的特性. BEA 和 IDC 提出的企业服务总线也与特定技术和产品相关<sup>[26,31]</sup>,前者也提出了一个示意性的 SOA 参考模型<sup>[32]</sup>. 文献<sup>[33]</sup>提出了一个概念性的 SOA 参考模型,它由概念模型、应用参考模型、SOA 过程和 SOA 模式组成. 上述的组织发布的 SOA 参考模型与特定的实现服务的标准、技术或产品结合过于紧密,这会影响其适用范围.

## 6 结束语

本文对 SOA 所涉及到的重要概念进行了定义和深入的阐述,明确地界定了 SOA 的组成部分以及其间的关系. 在此基础上形成的 SOA 参考模型,

① Douglas G., Jason on maturity & SOA. <http://www.mail-archive.com/service-orientated-architecture@yahoogroups.com/msg00542.html>, Oct 2005.

② Douglas G., SOA maturity model. <http://www.mail-archive.com/service-orientated-architecture@yahoogroups.com/msg00460.html>, Oct 2005.

③ Michael H., Bath U.. The business-oriented background of service backbond. <http://www.servicebackbond.org/>, January 2004.

能以规范的形式用于指导对具体 SOA 的建立以及评价,也有益于对 SOA 的共同理解。

该参考模型涵盖了本文所提到的各种 SOA 参考模型中的必要概念和构成。该参考模型没有限定具体的 SOA 所必须遵循的实现标准和技术,因而它具有通用性。经过检验,它均适合于建立本文所提到的各种 SOA。

在 SOA 参考模型中,应该给出其可选构造块的选择规则和策略。在对服务合约的元模型的描述上,本文采用了 UML2.0 的概念与表示法,但并不是基于 UML2.0 元模型扩展的。这些都是需要进一步研究的内容。

### 参 考 文 献

- Krüger I. H., Mathew R. . Systematic development and exploration of service-oriented software architectures. In: Proceedings of the 4th Working IEEE/IFIP Conference on Software Architecture, Oslo, 2004, 177~187
- Krüger I. H. . Service specification with MSCs and roles. In: Proceedings of the IASTED International Conference on Software Engineering, Innsbruck, 2004, 387~403
- Modafferi S., Mussi E., Maurino A., Pernici B. . A framework for provisioning of complex e-Services. In: Proceedings of the 2004 IEEE International Conference on Services Computing, Shanghai, 2004, 81~90
- Genoveva Vargas-Solar. Toward aspect oriented services coordination for building modern information systems. In: Proceedings of the 5th Mexican International Conference in Computer Science, Colima, Mexico, 2004, 353~360
- Schuschel H., Weske M. . Automated Planning in a service-oriented architecture. In: Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, Modera, Italy, 2004, 75~80
- Xu Wei, Wei Jun, Liu Yu, Li Jing. SOWAC: A service-oriented workflow access control model. In: Proceedings of the 28th Annual International Computer Software and Applications Conference, Hong Kong, 2004, 128~134
- IBM. BPEL4WS. [www-106. ibm. com/developerworks/web-service/library/ws-bpel/](http://www-106.ibm.com/developerworks/web-service/library/ws-bpel/)
- W3C. Web Services Choreography Description Language Version 1.0. 2004. <http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/>
- OASIS. Reference Model for Service Oriented Architectures (Draft 1.0). February 2, 2006. [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=soa-rm](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm)
- CBDI. Web Service Road Map. <http://www.cbdifirum.com/index.php3>
- Sprott D., Wilkes L. . Understanding service oriented architecture. Microsoft Architects Journal, 2004, (1): 10~17
- ISO/IEC. Information Technology—Quality of Service; Framework. ISO/IEC International Standard 13236, First Edition, Dec 1998
- ISO/IEC. Information Technology—Quality of Service: Guide to Methods and Mechanisms. ISO/IEC Technical Report 13243, First Edition, Nov 1999
- ISO/IEC. Open Distributed Processing: Reference Model—Quality of Service. ISO/IEC JTC1/SC21 Working Draft, July 1997
- OMG. UMLTM Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms. Sept 2004. <http://www.omg.org>
- Gu Xiao-Hui, Nahrstedt Klara, Yu Bin. An integrated peer-to-peer service composition framework. In: Proceedings of the 13th IEEE International Symposium on High Performance Distributed Computing, Honolulu, Hawaii, 2004, 110~119
- Gu X., Nahrstedt K., Chang R. N., Ward C. . QoS assured service composition in managed service overlay networks. In: Proceedings of the IEEE 23rd International Conference on Distributed Computing Systems, Providence, RI, 2003, 194~200
- Casati Fabio, Shan Ming-Chien. Definition, execution, analysis, and optimization of composite E-Services. IEEE Data Engineering Bulletin, 2001, 24(1): 29~34
- Cotroneo D., Di Flora C., Russo S. . Improving dependability of service oriented architectures for pervasive computing. In: Proceedings of the 8th IEEE International Workshop on Object-Oriented Real-Time Dependable Systems, Guadalajara, Mexico, 2003, 74~81
- Tanenbaum Andrew S., van Steen Maar En. Distributed System; Principles and Paradigms (Chapter 7). Pearson Education Inc., 2002
- OASIS. OASIS Business Transaction Protocol(1.0). <http://www.oasis-open.org/business-transaction/>, 2002
- W3C. Web Services Transaction Specification. [http://www-106. ibm. com/developerworks/library/ws-transpec/](http://www-106.ibm.com/developerworks/library/ws-transpec/)
- W3C. WS-Coordination 1.0. August 2002. [http://dev2dev. bea. com/technologies/webservices/standards.jsp](http://dev2dev.bea.com/technologies/webservices/standards.jsp)
- Attachmate Corporation. Fundamentals of Service Orientation. Feb 2005. [www. Synapta. com](http://www.synapta.com)
- Sonic Software Corporation. A new Service-oriented Architecture Maturity Model. 2005. [http://wp. bitpipe. com/resource/org\\_998677371\\_410/SOA-Maturity\\_edp. pdf? site\\_cd=ssc](http://wp.bitpipe.com/resource/org_998677371_410/SOA-Maturity_edp.pdf?site_cd=ssc)
- BEA. The Enterprise Service Bus: Building Enterprise SOA. Dec. 09, 2004. [http://dev2dev. bea. com/pub/a/2004/12/soa\\_ibarra.html](http://dev2dev.bea.com/pub/a/2004/12/soa_ibarra.html).
- Krafzig D., Banke K., Slama D. . Enterprise SOA Service-Oriented Architecture Best Practices. Prentice Hall, 2005
- CBDI. Modernizing Application Integration with Service Oriented Architecture. 2005. [http://guides. computerworld. com/cwguides/search/viewabstract/77950/index.jsp](http://guides.computerworld.com/cwguides/search/viewabstract/77950/index.jsp)
- IBM. Business Agility Mega Trend-SOA. May 16, 2005. [http://www. research/ibm/ com/journal/sj/441/crawford. pdf](http://www.research.ibm.com/journal/sj/441/crawford.pdf)
- Dodani M. H. . Mirror, mirror on the wall, whose SOA is the best of them all? Journal of Object Technology, 2005, 4(5): 67~74
- IDC. Oracle Builds Comprehensive SOA Platform. January 2005. [http://www. oracle. com/technologies/soa/idc\\_soa\\_](http://www.oracle.com/technologies/soa/idc_soa_)

platform.pdf

- 32 BEA. Domain Model for SOA. April 2005. [www.bea.com/soa](http://www.bea.com/soa)



**MA Zhi-Yi**, born in 1963, Ph. D., associate professor. His research interests include software engineering environment, service-oriented computing, and object-oriented technology.

- 33 Microsoft. Service Oriented Architecture. October 26, 2004. <http://msdn.microsoft.com/architecture/soa/>

**CHEN Hong-Jie**, born in 1971, Ph. D., lecturer. His research interests include requirement engineering and component technology.

### Background

At present, there are many achievements on SOA, some of which have successfully been applied in some domains. However, these are far from what people have been expecting. There are still some important unsolved problems. Krüger *et al.* discuss how to use components to realize services, but do not tackle how to assemble services and describe quality of service. The project MAIS focuses on modeling multi-channel adaptive information systems based on services. The project group presents the meta-model provisioning perspective, which consists of descriptions of service functions and quality of service and so on. However, the concept interface is not used in the meta-model, and how service functions are described is not discussed. Some SOA models at higher abstract level have been presented, which lack essential details to apply these models. Many definitions of concepts or components of SOA relate to specific imple-

mentation technologies, such as Web technology or Jini middleware. Some specifications, such as BPEL4WS and WSCDL, are also for Web technology. However, Web technology can only realize some aspects of SOA. Moreover, there are some differences between the definitions of terms or in the understanding of SOA.

For solving problems above-mentioned, this paper presents the SOA reference model that can be used to design SOA. The paper deeply discusses some concepts on SOA, and expatiates on the structure of the reference model and its service bus and meta-model for service contracts. Moreover, the paper presents the maturity model for evaluating service-oriented architectures. The SOA reference model can be used as a guide for building specific SOA, and provide the foundation for stakeholders to understand SOA.