# 基于多层 chord 的 web 服务工作流构建①

赵怀金 杨晓红 杨 峰 (山东财政学院 计算机信息工程学院 山东 济南 250014)

**摘 要:** 针对传统的集中式的服务发布、服务发现存在的问题以及服务工作流只能在流程定义阶段静态绑定服务的问题，提出了采用基于服务分类的三层 chord 来组织 web 服务的方法和逻辑工作流与资源层交互的模型。首先介绍了 web 服务分类模型，然后对结构化 p2p 资源层进行建模，并提出了逻辑工作流的定义方法，最后设计了流程匹配、服务发现、服务选择和服务绑定四个功能模块来实现逻辑工作流到可执行工作流的映射。

**关键词:** 服务工作流；资源组织；chord；服务分类；流程映射

## Constructing Web Service Workflow Based on Multi-Layer Chord

ZHAO Huai-Jin, YANG Xiao-Hong, YANG Feng

(School of Computer & Information Engineering, Shandong University of Finance, Jinan 250014, China)

**Abstract:** For the problem of service publishing and discovery in traditional centralized mechanism and that of service workflow binding static service at flow design stage, this paper proposes a method of using service classification based 3-layer chord to organize services and the model of logical workflow interacting with resource layer. Web service classification model is introduced first. Then, structured p2p resource layer is modeled, and the method of logical workflow definition is introduced. Finally, four functional modules including process matching, service discovery, service selecting and service binding to realize the mapping from logical workflow to executable workflow are designed.

**Keywords:** service workflow; resource organization; chord; service classification; workflow mapping

## 1 引言

Web服务技术通过采用基于xml的wsdl、soap、uddi 等标准规范有效解决了分布式计算中不同系统之间的互操作问题，基于 web 搭建的应用具有跨平台、松耦合、软件复用等优点[1]。Web服务工作流技术主要解决如何定义参与流程的各个服务之间的逻辑和时序关系，从而实现复杂 web 服务执行的自动化，并实现服务组件之间的动态交互、协调和状态保持等问题[2-5]。传统的基于 UDDI 的 web 服务发现机制采用基于关键字匹配的查询机制，不能表达相关的语义信息，并且集中式的 UDDI 发现机制使得系统会因服务器出现故障而全面瘫痪[6]。引入 P2P 技术来处理服务元数据的交换,能够克服传统 UDDI 技术中服务元数据集中注册、集中存放对搜索广度带来的限制[7]。P2P 网络具有严格的拓扑结构、节点和数据对象位置确定、高效路由容错与动态自适应等特点，非常适合在异构环境下进行资源组织。chord 环作为第三代 p2p 网络(结构化网络)，是一个基于带弦环拓扑结构的分布式系统，提供数据对象的存储、查询、复制、缓存等功能，并且作为分布式散列表，chord 几乎具有最优的路由效率[8]。文献[9]提出了一种基于服务关系本体的交互式服务生成方法，在服务生成中,通过挖掘服务的动态语义来解决服务组合中生成复杂业务所涉及的用户个性化、动态生成等问题；文献[10]提出

① 基金项目:国家自然科学基金(60603070)

收稿时间:2009-09-29;收到修改稿时间:2009-10-22

了一种面向用户需求的服务工作流构造模型，对功能相同或相似的服务采用生成树的方式进行服务聚类，并根据工作流的业务逻辑关系形成生成业务图，最后采用基于混合粒子群的Qos调度方法来生成服务工作流的最佳执行路径；文献[11]提出将业务流程表达为流程模板，利用服务选取代理进行语义 web 服务集成，提出的模型解决了针对 bpel4ws 的流程引擎不能在运行时动态发现和绑定服务的问题，并提出了搜索模板库来为服务增加语义描述，提高服务绑定的准确度；文献[12]提出了一个构建于完全分布式 p2p 网络的中间件，实现了复杂语义请求的资源搜索，采用基于本体概念的属性向量标记，并对发现的资源进行有效聚容，通过基于语义相似度的个性化推荐策略实现了复杂语义环境下的资源推荐。本文吸取了文献[9-12]研究思想，进行了以下几方面的工作：(1)对 web 服务资源进行按照层次概念进行分类；(2)构建 p2p 资源层，按照服务分类建立多层 chord 环来组织服务资源；(3)提出一种基于服务概念分类和服务质量的逻辑工作流 xml 描述；(4)设计了一种为逻辑工作流中的活动节点从 P2P 资源层中绑定具体服务的方法。

## 2　构建p2p服务资源层

### 2.1 创建服务分类

把网络上的服务按照概念层级关系进行服务分类，建立服务分类表，每一个底层概念代表一类服务，上层概念用于底层概念的进一步归类。我们在本模型中设计了一个两层服务分类方法，顶层概念按照领域划分，下层概念在领域内部再进行细分，下层概念能够对具体服务进行标识。

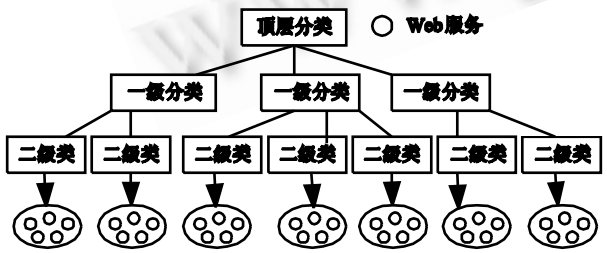


图 1　web 服务分类示意图

网络上存在大量功能相似但服务质量(例如响应时间、可用性、成本、能力、声望等)不同的 web 服务，功能相同的服务位于同一服务类别，从而把这一类服务用下层概念进行标识，从而能够把众多服务按照概念层级关系有机组织起来。服务类别划分如图 1 所示，图中表明服务概念经过了两次分类后，最底层的服务类别作为一系列具有相同服务功能的服务类别标识。

### 2.2 按照服务分类关系组织多层 chord 环

按照上述服务分类层次结构关系，把众多 web 服务用结构化的 Chord 环来分类组织。我们采用自底向上的方式构建，具体做法为：

(1) 把众多 web 服务按照一级分类和二级分类划分成多组服务集合，每组服务集合内的服务都具有相同的服务功能，但所具有的服务质量各异，每一个 web 服务都存在于一个 chord 节点上，知名 chord 节点保存了对该 web 服务类别的哈希值。

(2) 在每一个底层 chord 服务环中，选取一个二级知名节点，在同一个领域下的从众多 chord 环中选取的知名节点就会组成中间层 chord 环，每一个知名节点负责与同级的中间层 chord 环知名节点建立联系。

(3) 按照一级分类，在每一个中间层 chord 环中选取一个一级知名节点，把这些一级知名节点再次组织成一个顶层 chord 环。

采用多层 chord 组织 web 服务资源的结构模型如图 2 所示：

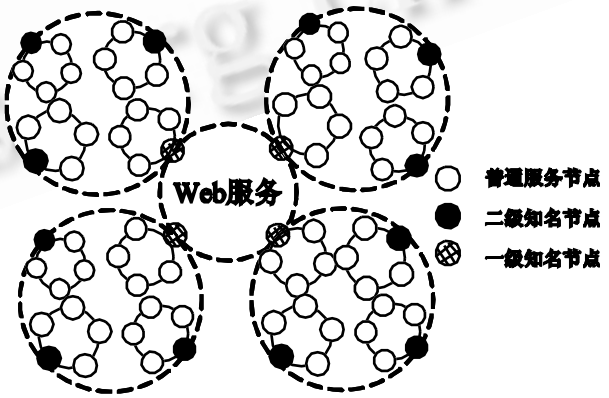


图 2 基于多层 chord 的 web 服务组织

如上图所示，最底层的具有相同服务能力的服务通过底层 chord 组织在一起，每一个底层 chord 环再分别选出一个二级知名节点，位于同一领域概念下的知名节点再相互连接构成第二层 chord 环，从第二层 chord 环中再次选取一个一级知名节点组成第三层

chord 环。这样就通过 chord 环组织实现了按领域划分大类，再从每一领域内划分小类来组织 web 服务资源。

### 2.3 服务节点在多层 chord 环中的维护策略

服务节点申请加入 p2p 服务资源层的策略如下：

Step1. 一级和二级知名节点维护了同一层 chord 的其他知名节点的路由信息，位于上层 chord 环中的知名节点只负责消息的路由转发。

Step1. 每一个服务都有两个(一级分类、二级分类)类别标识，首先对一级分类标识进行哈希操作，定位到顶层 chord 环中的节点由此进入一级分类的 chord 环中。

Step2. 根据二级分类定位底层 chord 环，找到具体 chord 的知名节点，这样就锁定了该服务所在的具体 chord 环。

服务节点退出策略由 chord 环自行维护，不用人为设定退出策略。

### 2.4 在多层 chord 环中进行服务查找的策略

服务查找策略与服务加入策略相似，同样是根据服务的(一级分类、二级分类)标识进行两次哈希定位，先定位领域 chord 环，再定位领域下的具体 chord 环，从而返回的结果是底层 chord 环中所有的服务，这些服务将通过第 4 章中的服务选择模块进行筛选。

通过这种分类的方式来查找所需要的服务，可以最大限度的排除无关服务的干扰，提高了查询效率。同时我们充分利用类别标识作为我们的知名节点的 KEY 值，可以省去了记录大量知名节点的工作，提高了系统的可靠性。

## 3 逻辑工作流的描述

### 3.1 逻辑工作流与可执行工作流定义

定义 1. 逻辑工作流是指组成处理流程的各个节点绑定的不是具体的 web 服务，而只是对服务的功能性和 Qos 描述。服务描述信息（类别和服务质量）可以在流程编制时静态输入，也可以通过人机交互界面根据流程执行顺序动态的为每一个活动节点输入服务描述信息。

定义 2. 可执行工作流是指在逻辑工作流的基础上，针对每一个活动所要求的服务分类从本文构建的 p2p 资源层中动态到查找候选的的服务实例集合，然后再对候选集合进行基于 Qos 的筛选，进而为每一个活动绑定具体服务实例，组成服务工作流。

### 3.2 逻辑工作流的 xml 描述

逻辑工作流的描述主要由三部分组成：①流程描述信息，是对流程的概要性描述，包括启动条件、生成日期、适用范围等；②控制结构，包括顺序、循环、分支等常用结构元素③活动元素，是流程中表达服务要求的节点，包括了对服务类别和服务质量的描述，本文中采用两个分类概念来描述服务功能性要求，采用性能、花费、可靠性、可提供性和声誉五个 Qos 参数来描述对服务的非功能性要求。文档描述片段如下所示：

```
<logical flow name=" logical workflow example" >
<Process Description>
<Starting conditions> … … </ Starting conditions><!—启动条件-->
<creation date>……</creation date><!—生成日期-->
<Applicable scope>……<Applicable scope><!—适用范围-->
……
</ Process Description>
<sequence><!—顺序结构-->
<parallel><!—并行结构-->
<activity category=A subcategory=A1 time<t1 price<p1 capacity<c1 availability>p1 reputation>r1 />
<activity category=B subcategory=B2 time<t2 price<p2 capacity<c2 availability>p2 reputation>r2 />
……
</parallel>
<activity category=D subcategory=D4 time<t4 price<p4 capacity<c4 availability>p4 reputation>r4 />
<activity category=E subcategory=E1 time<t5 price<p5 capacity<c5 availability>p5 reputation>r5 />
……
</sequence>
</logical flow>
```

上述流程定义可以通过流程编辑器辅助用户生成，一些常用的逻辑流程以流程模板的形式存储在数据库中，用户根据个性化的要求配置里面的活动参数，从而可以生成如上所示的流程定义文件。其中描述信息用于用户需求与流程的匹配，具体方法可以通过编制规则文件的方式进行，在第 4 章中将会进行阐述。之后是用控制结构组织的若干活动，每一个活动规定了对所需服务的功能类别和 Qos 要求。

## 4　逻辑工作流映射为可执行工作流

如图 4 所示，我们设计了流程匹配、服务发现、服务选择和服务绑定四个功能模块来完成可执行服务工作流的生成。该模型分为逻辑工作流的产生和映射两个阶段。

### 4.1 逻辑工作流的产生

逻辑流程的产生也即图 4 中流程匹配模块，其内部实现是通过规则引擎(如 Jboss rules)来进行流程匹配的，匹配过程如下：(1)用户把对流程的需求信息(包括流程启动条件参数、流程生成日期、适用范围等)输入到规则引擎的推理机作为匹配条件，同时系统中预先设定的存放在规则库中的文件作为规则引擎的推理规则。(2)把匹配条件和推理规则装载到推理机中并由匹配器进行规则匹配，将匹配成功的规则实例放到执行器中。(3)匹配的结果由执行器输出，在此模型中匹配的结果就是流程案例库中存在的案例名称，根据案例名称对流程案例库进行检索即可得到所需的流程定义文档。通过规则引擎实现流程匹配的架构如图 3 所示：

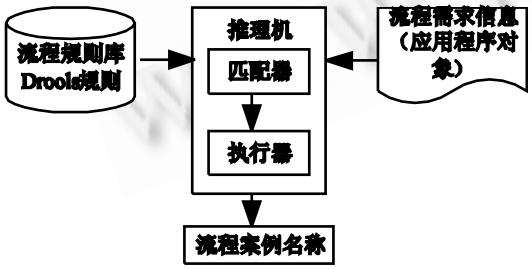


图 3　规则引擎实现流程匹配

### 4.2 逻辑工作流的映射

实现该功能涉及到对逻辑工作流中各个业务活动所关联的 web 服务的发现、选择、绑定的过程，对应于图 4 后面三个功能模块，具体步骤如下：

Step1. 把匹配出来的具体的流程定义输入服务发现模块，该模块逐个解析每一个活动节点，把每一个活动对应的两级服务类别输入 P2P 资源层，P2P 资源层通过上文提到的服务查找策略返回每一个活动所对应的一组候选服务列表。

Step2. 服务选择模块采用一定的服务匹配算法逐一对每一个活动所对应的候选服务进行基于服务质量的筛选，最终为每一个活动选择了相应具体服务。

Step3. 把上一步产生的活动与具体服务的对应关系提交到服务绑定模块，该模块通过解析流程定义的 xml 文件，把相关逻辑活动按照一定的语法替换成具体服务元素的绑定，从而生成了一个绑定具体服务的服务工作流。

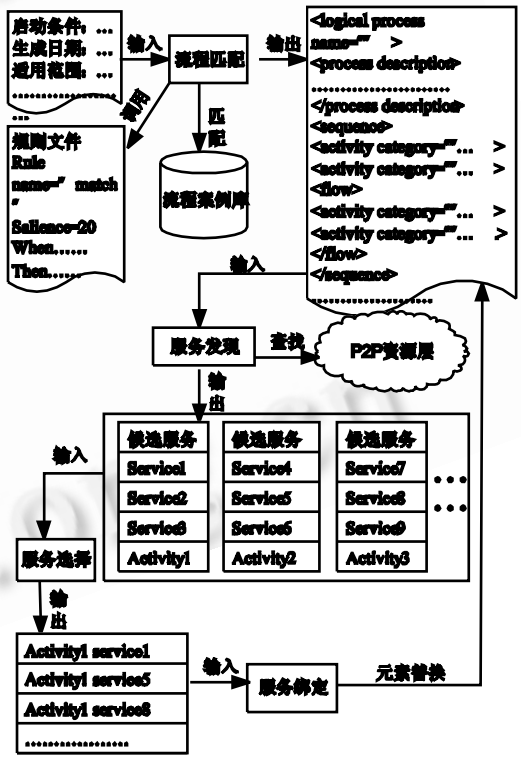


图 4　逻辑工作流到可执行工作流的映射过程

## 5　结语

本文针对 p2p 环境下 web 服务工作流动态生成问题，提出了逻辑工作流与可执行工作流的定义，给出了逻辑工作流到可执行工作流的映射方法。给出了用概念层级关系来组织服务资源的模型，并根据概念层级关系组建了三层 chord 网络来组织 web 服务，给出了逻辑工作流的 xml 表示以及如何与具体服务绑

定的方法，本文的逻辑工作流到可执行工作流映射过程中采用结构化 p2p 网络进行服务资源组织，充分利用了结构化 p2p 网络高效路由、准确定位、容易维护等优点，从而能够在工作流生成中快速进行服务发现，组成可执行工作流。

## 参考文献


1 Papazoglou MP. Service-Oriented computing: Concepts, characteristics and directions. Massimo M, ed. 4th Int'l Conf. on Web Information Systems Engineering (WISE 2003). Roma: IEEE Computer Society, 2003. 3－10.

2 Paolucci M, Kawamura T, Payne TR, Sycara K. Semantic matching of Web services capabilities. Horrocks, ed. the Int'l Semantic Web Conf. Sardinia: Springer-Verlag, 2002. 333－347.

3 Draluk V. Discovering Web services: An overview. Apers P, ed. the 27th Int'l Conf. on Very Large Data Bases. Roma: Morgan Kaufmann Publishers, 2001. 637－640.

4 Peer J. Bringing together semantic Web and Web services. Horrocks, ed. the Int'l Semantic Web Conf. Sardinia: Springer-Verlag, 2002. 279－291.

5 岳昆,王晓玲,周傲英. Web 服务核心支撑技术:研究综述.软件学报, 2004,15(3):428－442.

6 陈星豪,李陶深.一种基于P2P 的语义Web 服务发布和发现模型.计算机技术与发展, 2008,1(18):77－80.

7 陈德伟,许斌,蔡月茹,李涓子.服务部署与发布绑定的基于 P2P 网络的 Web 服务发现机制.计算机学报, 2008,4(28):615－625.

8 陈贵海,李振华.对等网络:结构、应用与设计.北京:清华大学出版社, 2007. 150－200.

9 徐萌,陈俊亮,彭泳,梅翔.基于服务关系本体的服务生成.软件学报, 2008,19(3):545－556.

10 胡春华,吴敏,刘国平,徐德.基于业务生成图的 web 服务工作流构造方法. 软件学报, 2007,18(8):1870－1882.

11 付燕宁,刘磊,张长海.流程模板驱动的web服务组合方法.吉林大学学报:工学版, 2008,38(增刊 2):169－172.

12 白云. P2P 环境中基于语义的资源自组织、发现及推荐研究[博士学位论文].重庆:西南大学, 2008.