

# 基于 Petri 网业务流程建模及到 BPEL4WS 的转换

洪晓霞, 方 洁, 江丕俊

(南京邮电大学, 江苏 南京 210003)

**摘要:** BPEL 语言是一种功能强大的描述业务流程的计算机可执行标准语言, 在 SOA (面向服务) 架构中扮演着将服务串联成商业流程的灵魂角色。但是由于它的不够直观和代码冗长等缺点, BPEL 不适合直接用于业务流程的抽象建模。因此研究其它抽象模型与 BPEL 间转换就相当有意义。在对 Petri 网和 BPEL 语言深入学习的基础上, 定义了 Petri 网的若干部件及部件与 BPEL 的对应转换, 然后提出了如何将业务流程 Petri 网模型转换成 BPEL 的算法, 并用实例进行分析, 验证该算法的有效性。

**关键词:** BPEL; Petri 网; 业务流程; 建模

**中图分类号:** TP311

**文献标识码:** A

**文章编号:** 1673-629X(2011)02-0170-03

## Translating the Model of Business Process from Petri net to BPEL

HONG Xiao-xia, FANG Jie, JIANG Pi-jun

(Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract:** BPEL4WS language is a powerful description language for the business process which can be executed by computer as a standard language. It is primary from service to the business process in SOA (service-oriented) system. But BPEL code is not intuitive and prolix, So it is not fit for the abstract modeling of business process. Therefore the translating from other abstract models to BPEL is very significant. Based the deep study on petri nets and BPEL language, this paper defines several components of the petri nets and the translating to BPEL code correspondingly, and then puts forward an algorithm about how to translate. Finally, a simple example was introduced for illustrating how to use the algorithm in business application which also can be proved the effectiveness of the method.

**Key words:** BPEL; petri net; business process; modeling

## 0 引言

BPEL4WS (Business Process Execution Language, 业务流程执行语言, 简称 BPEL) 是一种描述业务流程、计算机可执行的标准语言。BPEL 是一种描述能力强大的语言, 适用于十分复杂的企业过程<sup>[1]</sup>, 它的提出者指出它的两种可能用途: 实现可执行的业务流程和描述不可执行的抽象流程<sup>[2]</sup>。但是在当前的实践证明, BPEL 的抽象级别不适合用于业务流程设计和分析。因为 BPEL 语言不够直观, 且对于非专业人员来说, 使用起来较为困难。

目前针对业务流程进行抽象建模的方法主要有 Petri 网、UML、有限状态机等<sup>[3]</sup>。Petri 网于 1962 年被 Carl Adam Petri 作为一中过程建模和分析工具提出, 具有直观的图形表示、明确的语义、坚固的理论基础、

丰富的分析技术和适合于描述离散并行系统的特点, 因此, Petri 网被认为是一种高效的业务流程建模方法<sup>[4]</sup>。

由于 Petri 网和 BPEL 各自的优缺点, 针对 web 服务建模时, 可以同时使用这两种不同的建模工具, Petri 网应用于业务流程的抽象建模阶段, 对 Web 服务的模拟、分析和校验, 从而便于 Web 服务的随后实现, 而 BPEL 更适合作为一个标准的语言表现形式, 应用于实现可执行的业务流程<sup>[5]</sup>。因此为了在企业过程抽象建模与业务流程的执行之间架设桥梁, 就必须研究如何将图形化的企业过程语言 (如 Petri 网) 转换成 BPEL 代码。

## 1 BPEL4WS 简介

BPEL 本质上是一种程序设计语言, 它扩展了实现 Web 服务的结构, BPEL 过程定义与许多活动相关, 活动可以分为两大类: 基本活动和结构化活动<sup>[6]</sup>。基本活动是 BPEL 与其它 Web 服务之间交互的简单形式: BPEL 常用的基本活动或结构化活动如下: 接收 <receive>; 等候客户端发送消息来调用业务流程时, 用来

收稿日期: 2010-06-26; 修回日期: 2010-09-20

基金项目: 中国移动通信公司合作项目 (南邮 07 外 52)

作者简介: 洪晓霞 (1984-), 女, 硕士研究生, 研究方向为基于 IPv6 的下一代通信网络。

触发 BPEL 流程;回复<reply>,是整个流程的终点;赋值<assign>和调用<invoke>,是两个相连的任务,先与合作伙伴的接口的输入变量赋值,然后调用合作伙伴的服务;顺序<sequence>表示后一个活动只有在前一个活动结束后才能执行;选择<switch>根据条件判断选择执行相应的子流程;并行<flow>则是所有的活动同时执行;<while>定义循环结构,直到对指定的布尔条件求的值不再为真;<pick>;基于互斥条件的时间或外部触发活动;还有一些文中没有用到的活动等<sup>[7]</sup>。

BPEL 并不适合普通建模者直接使用,它的抽象能力还不适合业务过程的分析和设计,业务过程的分析、设计一般依靠 UML 和 Petri 网等更为高级的抽象建模语言,因此,从需求建模语言(如 Petri 网)到业务流程执行语言 BPEL 的映射是必要的。

## 2 Petri 网相关知识

Petri 网最早由德国科学家 Carl Adam Petri 于 20 世纪 60 年代初提出的,它是一种图形化的数学建模工具,可用来描述和分析具有并发性、异步性、并行性、非确定性和分布式等特点的信息处理系统。Petri 网的图形化的表示方法使其直观、易理解, Petri 网的数学理论基础使其可以深入分析系统的行为和性质<sup>[8]</sup>。近四十年来, Petri 网理论得到快速发展,各种高级和扩展 Petri 网被提出并成功应用于系统性能评价(系统分析与验证、模糊逻辑推理、工作流管理系统等领域)<sup>[9]</sup>。

### 2.1 Petri 网的相关定义

定义 1:一个三元组  $PN=(S, T; F)$  是一个 Petri net<sup>[10]</sup>, 当且仅当:

$$S \cup T \neq \emptyset;$$

$$S \cap T = \emptyset;$$

$$F \subseteq (S \times T) \cup (T \times S); (' \times ' \text{表示笛卡尔积})$$

$$4) \text{ dom}(F) \cup \text{cod}(F) = S \cup T; \text{其中:}$$

$$\text{dom}(F) = \{x \in S \cup T \mid \exists y \in S \cup T: (x, y) \in F\},$$

$$\text{cod}(F) = \{y \in S \cup T \mid \exists x \in S \cup T: (x, y) \in F\}$$

$S, T$  分别称为 PN 的库所(place)集和变迁(transition),  $F$  为 PN 的流关系(flow relation)。

定义 2:令  $PN=(S, T, F)$  是一个 Petri 网,  $C$  是 PN 的一个部件(特殊子网),令  $PN|_C=(S', T'; F')$ , 其中  $S' \subseteq S, T' \subseteq T, F' \subseteq F$ 。

若  $C$  为 sequence(顺序)部件,则  $|T'| > 1 \wedge (\forall s \in S' \mid |s'| = |s| = 1) \wedge (\forall t \in T' \mid |t'| = |t| = 1)$ ;

若  $C$  为 flow(并行)部件,则  $|T'| > 4 \wedge (\forall s \in S' \mid |s'| = |s|)$

$$\wedge \exists (t_1 \in T', t_2 \in T', t_1 \neq t_2) (|t_1'| = |t_2'| = 1 \wedge t_1' \subset t_1' \wedge t_2' \subset t_2') \wedge (t_1' \cup t_2' = S')$$

若  $C$  为 switch(分支)部件,则  $\exists s_1, s_2 \in S' \forall t \in T' (|t'| = s_1 \wedge |t'| = s_2 \wedge S' = \{s_1, s_2\})$ ;

若  $C$  为 while(循环)部件,则  $\exists t \in T', s \in S' \mid s'| > 1, s' \subseteq s'$ 。

定义 3:令  $X = \{\text{sequence/flow/switch/while}\}$ , 若称  $C$  为最大  $X$  部件,则当且仅当: $C$  为  $X$  的一个部件,并且不存在  $C'$  为 PN 的另一个  $X$  组件,满足  $C \subset C'$ 。

### 2.2 部件与代码转化

2.1 节中已定义了 Petri 网主要部件的数学表示方法,各部件的 BPEL 代码如何实现如表 1~4 所示。

表 1 sequence 部件的 BPEL 代码实现

顺序部件	转化后的变迁	变迁 tc 对应的 BPEL 代码
		<pre>&lt;sequence&gt; &lt;invoke name="t1"/&gt; &lt;invoke name="t2"/&gt; &lt;/sequence&gt;</pre>

表 2 switch 部件的 BPEL 代码实现

分支部件	转化后的变迁	变迁 tc 对应的 BPEL 代码
		<pre>&lt;switch&gt; &lt;case condition="con1"&gt; &lt;invoke name="t1"/&gt; &lt;/case&gt; &lt;case condition="con2"&gt; &lt;invoke name="t2"/&gt; &lt;/case&gt; &lt;/switch&gt;</pre>

表 3 while 部件的 BPEL 代码实现

循环部件	转化后的变迁	变迁 tc 对应的 BPEL 代码
		<pre>&lt;while condition="con1"&gt; &lt;invoke name="t"/&gt; &lt;invoke&gt; &lt;/while&gt;</pre>

## 3 Petri 网模型的 BPEL 代码实现算法

为了使生成的 BPEL 模型正确且可维护,因为我们在转化的时候应选择最合适的 BPEL 结构来替换 Petri 网中的相应部分。Petri 网模型的 BPEL 代码实现方法的基本思想是:采用分层分解 Petri 网的策略,将

Petri 网的变迁对应于 BPEL 的原子活动,将 Petri 网的各部件映射为 BPEL 的相关代码(如何映射,遵从上面的介绍),然后将一个简单的变迁(当为循环部件时为一个库所和变迁)来替代 Petri 网复杂的部件,加入到变迁库中,直至变迁库中只剩下一个变迁为止(一般最后都是由顺序结构转化为一个变迁)<sup>[11]</sup>。

表 4 flow 部件的 BPEL 代码实现

并行部件	转化后的变迁	变迁 tc 对应的 BPEL 代码
		<pre> &lt;sequence&gt; &lt;invoke name="t1"/&gt; &lt;flow&gt; &lt;invoke name="t2"/&gt; &lt;invoke name="t3"/&gt; &lt;/flow&gt; &lt;invoke name="t4"/&gt; &lt;/sequence&gt; </pre>

具体算法如下:

令  $PN = (S, T; F)$

(1)  $X := PN$ ; 当  $[X] \neq \emptyset$  时,按如下遍历全网:

(2) 当  $X$  中只包含一个变迁,那么转(7);

(3.1) 如果有最大的 sequence 部件  $C \in [X]$ ,那么选择  $C$ ,转(5);

(3.2) 如果有 switch 部件  $C \in [X]$ ,那么选择  $C$ ,转(5);

(3.3) 如果有 while 部件  $C \in [X]$ ,那么选择  $C$ ,转(5);

(3.4) 如果有最大的 flow 部件  $C \in [X]$ ,那么选择  $C$ ,转(5);

(4) 如果有其它的结构(若类似上述几种部件,则转化的代码也类似)  $C \in [X]$ ,那么选择  $C$ ,转(5);

(5) 把  $C$  映射为 BPEL 代码,将该代码添加到单个变迁  $t_c$  中;

(6.1) 如果是 while 结构  $X := (X - C) \cup t_c \cup s_c$ ,返回(2);

(6.2) 如果是其它结构,  $X := (X - C) \cup t_c$  返回(2);

(7) 输出  $X$  的 BPEL 代码。

#### 4 转化实例

假设有一企业生产机械产品,其部分基本生产过程大致为:采购部完成采购后,生产过程由两个部门同时生产,分别生产主零配件,产品检测后若不合格则交予检修部检修或其他处理,合格者送交销售部销售。建模后的 Petri 网如图 1 所示<sup>[12]</sup>,其中  $t_1$  为下单,  $t_2$

为采购,  $t_3$  为生产机械核心零件,  $t_4$  为生产机械其它配件,  $t_5$  为组装机,  $t_6$  为产品检测,  $t_7$  为送检修,  $t_8$  为销售。

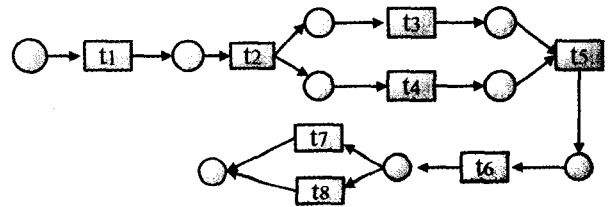


图 1 原案例 Petri 网

根据上述算法,发现上图所示的 Petri 网中无最大的顺序部件,但存在从变迁  $t_2$  到  $t_5$  的并行部件,将此部分替换为变迁  $tc_1$ ,并将代码加入其中,替换后的 Petri 网如图 2 所示。

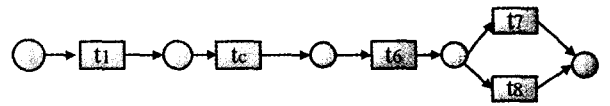


图 2 替换并行部件后的 Petri 网

在图 2 中发现存在最大的顺序部件,即从变迁  $t_1$  到  $t_6$ ,将此部分替换为单个变迁,并将转化的 BPEL 代码加入,以此类推,图 1 所示的 Petri 网转化为 BPEL 代码为:

```

<sequence>
<invoke name="t1"/>
<sequence>
<invoke name="t2"/>
<flow>
<invoke name="t3"/>
<invoke name="t4"/>
</flow>
<invoke name="t5"/>
</sequence>
<invoke name="t6"/>
<switch>
<casecondition="检测不合格">
<invoke name="t7"/>
</case>
<casecondition="检测合格">
<invoke name="t8"/>
</case>
</switch>
</sequence>

```

#### 5 结束语

给出了基于 Petri 网的业务流程模型的 BPEL 代码实现的算法:采用逐步分解的方法,先将 Petri 网的各个部件转化成带有 BPEL 标注的单个变迁,最后对这些变迁进行 BPEL 替换。

(下转第 202 页)

辑关系,又可以由作战行动效能计算得到不同级别作战任务的作战效能,从而发现部队作战过程中的薄弱环节,进行针对性的训练,提高部队总体作战能力。

## 5 结束语

IDEF3 建模方法简单易学,便于理解和掌握,具有严格、清晰、明确的语法,建立的模型易于交流和维护。文中利用扩展的 IDEF3 模型对部队作战过程进行形式化描述,达到各作战系统对任务的共同理解;同时,基于作战行动效能并结合 IDEF3 模型对部队的作战效能进行了评估,减小了主观定性因素对评估的影响。一方面,部队作战效能评估是一个比较复杂的问题,应继续研究如何扩充效能评估框架以适应部队多种作战样式的效能评估;另一方面,部队效能评估的效果很大程度上取决于作战行动效能,如何利用部队训练效果数据准确地得到作战行动的效能数据,将是下一步研究的重点。

### 参考文献:

- [1] 傅攀峰,罗鹏程,周经纶.对武器装备体系效能评估的几点看法[J].系统工程学报,2006,21(5):548-552.
- [2] 李秉,王凤山,李晓军.一种弹炮结合武器系统作战效能评估方法[J].计算机技术与发展,2009,19(6):217-220.

(上接第 172 页)

在案例研究中采用了该转换算法,并根据实际应用生产中的业务流程例子进行分析,实现了该业务流程模型的 BPEL 代码实现,显示了算法的实用性和有效性。

目前研究 BPEL 代码转化为 Petri 网模型,然后对系统进行验证分析比较多,而从 Petri 网模型的代码实现则较少。但事实上一般普通建模者不太适合使用 BPEL 语言对业务流程直接进行抽象建模,而 Petri 网与 BPEL 又具有各自的优缺点,分别适合对业务流程进行抽象建模和可执行的业务流程建模,因此研究如何将 Petri 网转换为 BPEL 代码就很有意义和价值。

### 参考文献:

- [1] 陈艺军,王力生. Web 服务业务流程执行语言(BPEL4WS)引擎架构的研究[J].计算机工程与应用,2004(27):123-125.
- [2] Curbera F, Golan Y, Klein J, et al. Business Process Execution Language for Web Services [EB/OL]. 2006-02-16 [2007]. <http://dev2dev.bea.com/techtrack/BPEL4WS.jsp>.
- [3] 罗海滨,范玉顺.工作流技术综述[J].软件学报,2000,11(7):899-907.

- [3] 杨晓恕,张宏军,马勇波.合同战术训练评估系统体系结构[J].计算机工程,2008,34(15):54-56.
- [4] 付东,方程,王震雷.作战能力与作战效能评估方法研究[J].军事运筹与系统工程,2006,20(4):35-39.
- [5] 谢毅,唐任仲,缪亚萍.基于 IDEF3 的业务过程仿真模型的存储与获取[J].系统工程理论与实践,2005,25(12):69-75.
- [6] Young Jeong Ki. Conceptual frame for development of optimized simulation based scheduling systems[J]. Expert Systems with Applications,2000,18(1):299-306.
- [7] Sadiq O. Analyzing Process Models Using Graph Reduction Techniques[J]. Information Systems,2000,25(2):117-134.
- [8] Kim C H, Weston R H. The complementary use IDEF and UML modeling approaches[J]. Computers in Industry,2003,50:35-56.
- [9] 李建军,刘翔,黄光奇,罗雪山.一种作战任务分析仿真模型自动生成方法研究[J].系统仿真学报,2007,19(5):1114-1118.
- [10] 金伟新.“串并联”模型框架与作战效能评估[J].系统工程与电子技术,2001,23(7):41-43.
- [11] 宁可,李清,陈禹六.基于仿真方法验证 IDEF3 过程模型的有效性[J].清华大学学报(自然科学版),2005,45(4):540-544.
- [12] 杨晓恕.合同战术训练与管理评估系统关键技术研究[D].南京:解放军理工大学,2008.

- [4] 李龙澍,胡正梁.基于 Petri 网的 UML 形式化建模应用分析[J].计算机研究与发展,2010(4):76-79.
- [5] White S A. Using BPMN to Model a BPEL Process[J]. BPTrends,2005,3(3):1-18.
- [6] Arkin A, Askary S, Bloch B, et al. Web Services Business Process Execution Language Version 2.0 [M]. Working Draft. WS-BPEL TC OASIS, 2005.
- [7] Leymann F, Roller D. Modeling Business Processes with BPEL4WS[J]. Information Systems and E-Business Management, 2006, 4(3):265-284.
- [8] 毛伟伟,于素萍.基于 Petri 网的 Web 服务动态组合[J].计算机研究与发展,2009(11):61-63.
- [9] Toorn R. Component-Based Software Design with Petri nets: An Approach Based on Inheritance of Behavior[D]. Eindhoven: Eindhoven University of Technology, 2004.
- [10] 袁崇义. Petri 网原理与应用[M].北京:电子工业出版社,2005.
- [11] 樊玮,匡载华.基于 UML 的业务流程建模及到 BPEL4WS 的转换[J].航空计算技术,2010,40(1):79-84.
- [12] 谷建鑫,仇建伟.基于 Petri 网的工作流模型[J].计算机工程与设计,2005,26(2):513-515.