

基于 π 演算的智能流程建模

冯立元¹, 李德生², 李维翠¹

(1. 山东财经大学 计算机科学与技术学院, 山东 济南 250014;
2. 山东财经大学 网络信息中心, 山东 济南 250014)

摘要:为适应当前云计算环境下客户的个性化要求,一种新的智能流程的概念应运而生。首先提出了智能流程平台架构,描述了智能流程生成模块、社区云服务管理模块、智能流程执行模块三大模块。 π 演算能够描述通信拓扑结构的动态改变,具有强大的表达能力,随后运用 π 演算对智能流程进行建模描述。为了适应用户需求的不断变化,介绍了用户需求变更描述模型,并给出了形式化表示方法。依据互模拟理论验证建模正确性表明 π 演算建模较好地描述了智能流程下用户多变的个性化需求。

关键词: π 演算;智能流程;个性化流程;流程建模

中图分类号:TP311.52

文献标识码:A

文章编号:1673-629X(2012)09-0111-05

Intelligent Process Modeling Based on π -calculus

FENG Li-yuan¹, LI De-sheng², LI Wei-cui¹

(1. School of Computer Science & Technology, Shandong University of Finance and Economics,
Jinan 250014, China;

2. Network Information Center, Shandong University of Finance and Economics, Jinan 250014, China)

Abstract:In order to meet the customer's individual requirements under the current cloud computing environment, a new concept called intelligent processes is proposed. In this paper, intelligent process platform architecture was proposed, the intelligent process generation module, the community cloud service management module and intelligent process execution module of the platform were described. π -calculus can describe the dynamic changes in communication topology with strong expression skills. Then used π -calculus to describe the intelligence process. Described the user requirements change description model to meet the changeable requirements, and introduced the formal expression. Verification based on bisimulation showed that π -calculus described customer's changeable individual requirements perfectly.

Key words: π -calculus; intelligent process; personalized process; process modeling

0 引言

随着互联网的快速发展,网络已经深深影响了人们的生活和工作,从Internet上获取信息和服务来完成各种各样的任务已经成为一种趋势。受此影响,传统的业务流程模式面临着巨大的挑战。

首先流程使用者发生了变化。普通用户数目众多,而需求常常充满个性化,统一定制的流程模式无法满足用户需求^[1];其次,流程的活动完成者也面临着变化。当云计算大行其道时,完成活动的主体已经扩展到互联网中所有满足要求的云服务。云服务的数量呈现出爆炸式增长的趋势,具有相同或相似功能的服务

会越来越多。

旅游行业,特别是自助旅游业务,是一个正需要新型业务流程模式的领域。现在很多旅游网站像携程旅行网、乐途网等,都提供了许多旅游自助服务,例如景点查询、机票预订、宾馆预订等。但是作为一个普通游客,只是对旅游时间和旅游目的地比较明确,对于如何安排自己的这趟旅行缺乏经验。游客希望只需提出自己的需求就能从旅游网站上获取一个比较合理的旅游流程。

传统的流程模式已经不能适应当前云计算环境下客户的个性化要求。用户希望拥有一个更加好用易用的面向流程的应用模式,在此模式中用户只需要提出自己的需求,不必具备复杂的领域知识,也不必关心流程规划的细节,就可以得到一个合理得当的流程^[1]。此模式通过与用户的实时交互可以随时执行或终止流程,并根据用户需求不断更新流程,让用户享受更加人

收稿日期:2011-12-26;修回日期:2012-04-03

基金项目:山东省自然科学基金(2009ZRB019PF)

作者简介:冯立元(1987-),男,山东潍坊人,硕士研究生,研究方向为业务流程管理、Web服务组合;李德生,硕士,副教授,CCF会员,研究方向为服务组合、智能流程的构建方法、业务流程管理、CSCW。

性化的服务,给用户一次全新的服务体验。这便是一种新的智能流程模式。

1 智能流程平台架构

依据上文描述,文中初步构建了智能流程应用平台架构模型,如图 1 所示。智能流程应用平台架构模型包括智能流程生成模块、社区云服务管理模块、智能流程执行模块。

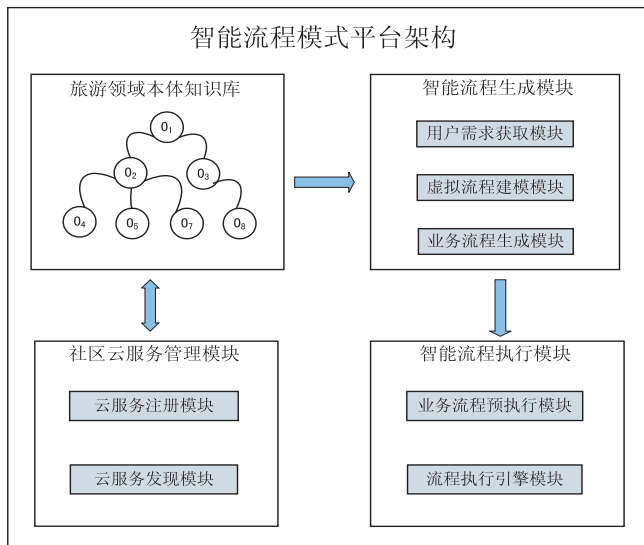


图 1 智能流程应用平台架构

1.1 智能流程生成模块

智能流程生成模块的作用是首先获取用户需求,并规范化处理、分析用户的需求形成需求描述文件,然后依据需求描述文件生成智能流程。智能流程生成模块包括用户需求获取、虚拟流程建模和业务流程生成三个子模块。

用户需求获取模块:用户需求获取模块的作用是通过与用户的交互获取用户个性化需求。智能流程模式下,友好的用户界面不仅仅是一种交互工具,它在保证完整获取用户个性化要求的同时还要对用户的需求做出分析和规范,为下一步顺利进行虚拟流程构建以及 web 服务发现匹配提供保障。

虚拟流程建模模块:虚拟流程建模模块的作用是根据用户需求获取模块得到的用户需求描述文件,自动生成一个虚拟流程。虚拟流程中定义了流程的执行顺序,并在每个任务节点上添加完成该任务的服务的描述信息。文中采用进程代数 π 演算对智能流程建模。

业务流程生成模块:业务流程生成模块的功能是根据虚拟流程的定义,获取完成流程活动的服务,进而转换成可以在流程执行引擎中具体执行的流程。其实现过程是业务流程生成模块从虚拟流程的定义中获取对完成流程活动的服务的要求和描述信息,然后调用

社区云服务管理模块的服务发现子模块,从社区云服务中查找与描述信息相匹配的 web 服务作为候选服务,最后根据服务质量参数 QoS 对候选服务排序,并选取排名靠前的若干服务交与用户作最后选择。

1.2 社区云服务管理模块

社区云服务管理模块的作用是提供基于社区云模式的 web 服务管理,并提供对 web 服务的注册、选取、评价等功能。社区云是对互联网上 web 服务的一种虚拟组织管理方式。具体到旅游领域,就是对旅游服务按照实现功能进行聚类分组、抽象封装成云服务并组成功能不同的社区云,具有相似或相同功能的服务由同一个社区云管理,每个社区云具有一定的自治性^[2]。

社区云服务管理模块由两部分组成:服务注册模块和服务发现模块。

服务注册模块:服务注册过程就是服务提供商将 web 服务映射到社区云的过程。在服务注册时首先要定位所属的社区云,根据定位社区云的策略结合领域本体找到服务所属的社区云类型,上传服务描述文件,映射为虚拟云

节点并向社区云成员发出公告。

服务发现模块:服务发现模块的作用是根据虚拟流程的定义查找和匹配符合要求的 web 服务。服务发现模块需要根据虚拟流程中任务节点上的服务描述信息,在社区云中查找符合要求的 service。社区云是具有相同或相似功能的 web 服务逻辑描述的集合,流程执行引擎只要准确定位所需服务所在的社区云,就能按照云服务节点的描述找到实际的 web 服务。

1.3 智能流程执行模块

智能流程执行模块负责具体执行智能流程生成模块生成的智能流程。流程执行模块包括业务流程预执行模块和流程执行引擎模块。

业务流程预执行模块:为了尊重用户的选择权,也为了执行过程更加友好和人性化,在业务流程正式执行之前,预先向用户展现模拟的执行结果。业务流程预执行模块就是预先执行业务流程,并以一种恰当的方式向用户展示流程预执行的结果。

流程执行引擎模块:流程引擎负责具体执行匹配了具体服务之后的业务流程,旅游流程执行引擎需要协调大量的服务提供商以及众多的旅游服务。流程引擎还要负责转换不同引擎之间的业务流程格式。当发生异常时,执行引擎负责动态地选择流程方案,保证流程的正常执行,尽最大可能使用户感受不到异常的发生。

2 智能流程建模

2.1 π 演算简介

π 演算 (π -calculus)^[3] 允许在通信中传递通道的名字,因此能够描述通信拓扑结构的动态改变,具有强大的表达能力,拓展了智能流程建模的灵活性。同时 π 演算又继承了 CCS 简洁优美的语义理论——互模拟,借助于互模拟理论可以方便验证智能流程建模的正确性。 π 演算的图形化表示如图 2 所示,其中 P,Q 表示进程, \bar{y},y 代表端口,箭头表示通道。这个图表示的就是进程 P 从 \bar{y} 端口沿着通道向进程 Q 发送一条消息,进程 Q 通过端口 y 接收这条消息。

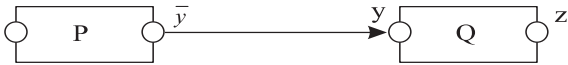


图 2 π 演算的图形化表示

2.2 运用 π 演算对智能流程建模

Tom 是一个酷爱周游世界各地的旅游迷,这次他要去美国洛杉矶度假,顺便观看 NBA 总决赛。Tom 之前并没有去过洛杉矶,所以他并不十分清楚如何合理安排这次旅行。Tom 访问了智能旅行社网站,并描述了他要去洛杉矶度假并观看 NBA 总决赛的需求。智能旅行社根据 Tom 的需求,并结合机票预订服务、宾馆预订服务、景点预订服务、NBA 球票预定服务等的特点,调用智能流程生成模块为 Tom 生成一个虚拟流程,即调用 NBA 球票预定服务订购 NBA 总决赛的球票,再依据比赛日程为 Tom 预定往返机票和住宿酒店,同时在比赛的间隙,根据景点预订服务为 Tom 安排景点旅游。

专家为 Tom 制定的简化的智能流程如图 3 所示,扩展的宾馆预订服务流程图如图 4 所示:

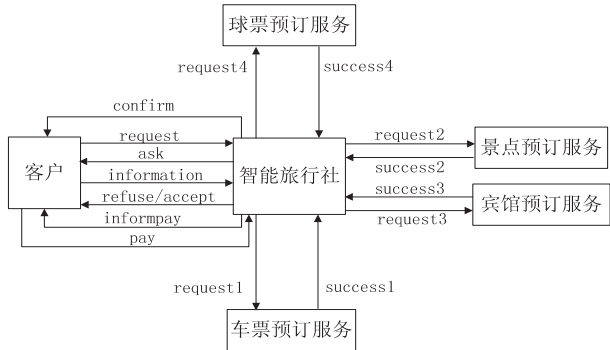


图 3 简化的智能流程图

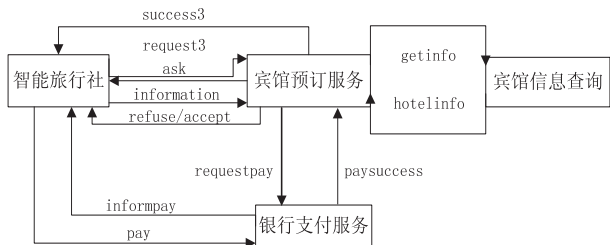


图 4 宾馆预订服务智能流程图

假设 x 为客户 client 与智能旅行社 agency 之间的通道, y 为智能旅行社 agency 与车票预定服务 book1 之间的通道, z 为智能旅行社 agency 与宾馆预定服务 book2 之间的通道, w 为智能旅行社 agency 与景点预定服务 book3 之间的通道, s 为智能旅行社 agency 与球票预定服务 book4 之间的通道。

对系统进行 π 演算建模如下:

客户:

$c = \{x, req, ask, info, acc, ref, infpay, pay, con\}$
 $client(c) = \bar{x} \langle req \rangle x(msg) [msg = ask] \bar{x} \langle info \rangle x(msg1) ([msg1 = ref] client(c) + [msg1 = acc] x(msg2) [msg2 = inf\ pay] \bar{x} \langle pay \rangle x(msg3) [msg3 = con] client(c))$

智能旅行社:

$t = \{x, y, z, w, req, ask, info, acc, ref, infpay, pay, con, req1, succ1, req2, succ2, req3, succ3, req4, succ4\}$

$agency(t) = x(msg) [msg = req] \bar{x} \langle ask \rangle x(msg1) [msg1 = info] (\bar{y} \langle req1 \rangle y(msg2) [msg2 = succ1] + \bar{y} \langle req2 \rangle y(msg3) [msg3 = succ2] + \bar{y} \langle req3 \rangle y(msg4) [msg4 = succ3] + \bar{y} \langle req4 \rangle y(msg5) [msg5 = succ4]) \bar{x} \langle inf\ pay \rangle x(msg6) [msg6 = pay] \bar{x} \langle con \rangle agency(t)$

车票预订服务:

$b1 = \{y, req1, succ1\}$
 $book(b1) = y(msg) [msg = req1] \bar{y} \langle succ1 \rangle book(b1)$

景点预订服务:

$b3 = \{w, req3, succ3\}$
 $book(b3) = w(msg) [msg = req3] \bar{w} \langle succ3 \rangle book(b3)$

球票预订服务:

$b4 = \{s, req4, succ4\}$
 $book(b4) = s(msg) [msg = req4] \bar{s} \langle succ4 \rangle book(b4)$

宾馆预订服务:

$b2 = \{z, req2, succ2\}$
 $book(b2) = z(msg) [msg = req2] \bar{z} \langle succ2 \rangle book(b2)$
对于扩展的宾馆预订服务流程建模,文中不再做过多阐述,可以参考相关参考文献[4,5]。

3 用户需求变更描述

智能流程应用模式中的建模方法不像传统流程模式“一次构建,长时间使用”,智能流程构建完成后,用户需求可能随时发生变化,为适应用户的个性化需求,需要智能流程模型不断调整。

在第一节描述的智能流程场景中,按照 Tom 的需求制定了个性化的业务流程。在动身前往洛杉矶观看

NBA 总决赛前, Tom 接到公司电话, 公司刚刚接了一个大订单, 公司要求 Tom 缩短旅游行程。Tom 比较无奈, 只好取消景点旅游服务。Tom 的需求发生了变化, 因此必须对智能流程做出调整。

3.1 需求变更描述模型

为解决用户需求变更问题, 文中参考并借鉴了用户需求变更描述模型^[6]。需求变更控制器由端口接收变更信息, 然后交由具体的进程演化控制器实施进程演化。进程演化包括增加新服务演化操作、删除服务演化操作、修改服务演化操作以及替换服务演化操作。

用户需求变更描述模型由需求变更控制器、进程演化控制器、演化过程规划器、演化规则检查器以及进程表达式组成, 用户需求变更描述模型如图 5 所示:

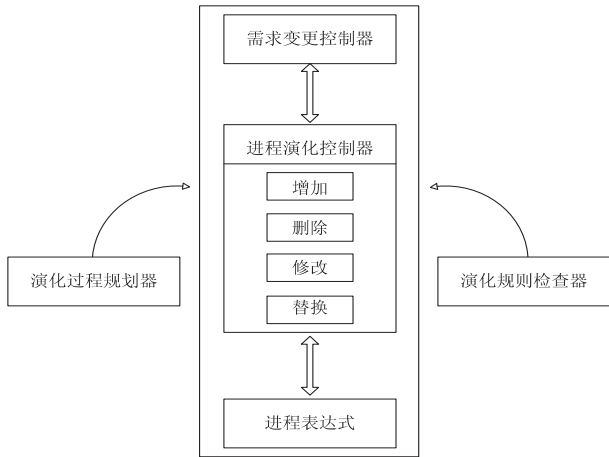


图 5 用户需求变更描述模型

演化过程规划器在接收到演化操作信号后, 发送消息到演化规则检查器, 请求演化规则检查器检查实现该演化操作是否符合进程规则库的规则。如果演化规则检查器返回的结果为 true, 也就是演化操作满足演化规则, 那么交与进程演化控制器执行该操作; 反之, 若返回结果为 false, 则需要根据用户需求制定新的演化步骤, 保证演化操作符合规则^[6]。

演化规则库中的规则是根据进程的特点定义的。这些规则用于在演化前获取变更信息, 并检查演化过程是否存在某种依赖关系而禁止进程演化。

在 π 演算中, 并没有一个方法可以直接对进程做出修改, 因此无法直接添加和删除子进程。因此, 需要改造原有进程以支持进程的动态更改。

(1) 可终止的进程: 假设 Process 是原进程, 那么可终止的进程表示为: $\text{Process} = \text{remove. } 0 + \text{Process}$ 。其中 remove 端口用于接收终止信号, 当收到终止信号后进程就会自动终止。

(2) 可变更的进程: 可变更的进程用于实现进程的动态演化, 既可以实现动态删除子进程, 又可以实现动态地添加子进程。可变更的进程表示方式如下:

$\text{comProcess} = \text{create}(\text{Process}).(\text{Process} \mid \text{comProcess})$, 其中 Process 是可终止的进程。

3.2 需求变更描述模型的形式化表示

为了简化描述, 文中将利用 π 演算对需求变更描述模型中的增加新服务演化操作和删除服务演化操作进行建模描述:

(1) 增加新服务演化操作。

$$\text{evolAgent}(e) = y(\text{msg})[\text{msg} = \text{add}] \bar{x} < \text{addcheck} > x(\text{msg})[\text{msg} = \text{more}] \bar{x} < \text{detail} > x(\text{msg})([\text{msg} = \text{no}] \bar{y} < \text{false} > \text{evolAgent}(e) + [\text{msg} = \text{yes}] \bar{y} < \text{true} > \text{evolAgent}(e))$$

$$\text{conAgent}(c) = x(\text{msg})[\text{msg} = \text{addcheck}] \bar{x} < \text{more} > x(\text{msg})[\text{msg} = \text{detail}] (\bar{x} < \text{no} > \text{conAgent}(c) + \bar{x} < \text{yes} > \text{conAgent}(c))$$

$$\text{addProcess}(a) = \bar{y} < \text{add} > y(\text{msg})([\text{msg} = \text{false}]. \tau \text{addProcess}(a) + [\text{msg} = \text{true}] \bar{z} < \text{create}(\text{Process})(\text{Process}) \mid \text{comProcess} > \text{addProcess}(a))$$

$$\text{comProcess}(p) = z(\text{msg})[\text{msg} = \text{create}(\text{Process})(\text{Process} \mid \text{comProcess})]$$

addProcess 进程在收到增加新服务演化操作后发送消息到演化过程规划器, 演化过程规划器收到消息后请求演化规则检查器依据进程规则库的规则判断添加该服务是否符合演化规则。如果演化规则检查器返回的结果是允许执行添加操作, 那么交与进程演化控制器执行该演化操作, 否则不能执行该演化操作。addProcess 进程收到演化规则检查器返回的结果为真的消息后, 对原进程表达式进行修改。

(2) 删除服务演化操作。

$$\text{evolAgent}(e) = y(\text{msg})[\text{msg} = \text{delete}] \bar{x} < \text{delcheck} > x(\text{msg})[\text{msg} = \text{more}] \bar{x} < \text{detail} > x(\text{msg})([\text{msg} = \text{no}] \bar{y} < \text{false} > \text{evolAgent}(e) + [\text{msg} = \text{yes}] \bar{y} < \text{true} > \text{evolAgent}(e))$$

$$\text{conAgent}(c) = x(\text{msg})[\text{msg} = \text{delcheck}] \bar{x} < \text{more} > x(\text{msg})[\text{msg} = \text{detail}] (\bar{x} < \text{no} > \text{conAgent}(c) + \bar{x} < \text{yes} > \text{conAgent}(c))$$

$$\text{delProcess}(a) = \bar{y} < \text{delete} > y(\text{msg})([\text{msg} = \text{false}]. \tau \text{delProcess}(a) + [\text{msg} = \text{true}] \bar{z} < \text{remove. } 0 + \text{Process} > \text{delProcess}(a))$$

$$\text{comProcess}(p) = z(\text{msg})[\text{msg} = \text{remove. } 0 + \text{Process}]$$

delProcess 进程在收到删除服务演化操作后发送消息到演化过程规划器, 演化过程规划器收到消息后请求演化规则检查器依据进程规则库的规则判断删除该服务是否符合演化规则。如果演化规则检查器返回的结果是允许执行删除操作, 那么交与进程演化控制器执行该演化操作, 否则不能执行该演化操作。delProcess 进程收到演化规则检查器返回的结果为真的消息后, 对原进程表达式进行修改。

依据本小节开头描述的用户需求变化, Tom 要取消景点旅游服务, 在已建立的智能流程模型基础上做如下演化:

```
del Process(a) =  $\bar{y} < delete > y (msg) [msg = true] \bar{z} < remove. 0 + book(b3) > del Process(a)$   
 $agent(t) = z(msg) [msg = remove. 0 + book(b3)]$ 
```

4 相关工作

关于智能流程这一新兴的研究领域, 国内的研究不过五六年的时间, 相关文献也并不多。相关研究主要集中在智能流程的用户个性化需求获取、智能流程建模、智能流程执行异常处理、智能流程中的 web 服务管理问题等方面。文中的主要工作是对智能流程建模, 既要准确描述用户的个性化需求, 又要随用户需求变更及时演化模型。

目前国内外出现的业务流程描述方法、技术和模型主要有: 基于 Petri 网的流程建模^[7,8], UML 建模方法^[9], 事件驱动过程链模型 (Event-driven Process Chain, EPC)^[10], 扩展的事件驱动过程链模型^[11], 基于进程代数的流程建模^[12]。 π 演算 (π -calculus) 是以进程间的移动通信为研究重点的并发理论。 π 演算的基本计算实体是名字和进程, 进程之间是通过传递名字来完成通信^[3]。 π 演算能够建立新通道, 并且能够描述结构不断变化的并发系统。

文中智能流程构建完成后, 在流程的执行过程中可能根据客户需求随时加以调整。因此, 智能流程的结构是不断变化的。而 π 演算可以通过建立新通道来描述结构不断变化的并发系统, 这是文中采用 π 演算作为流程建模描述语言的主要原因。

5 结束语

智能流程模式适应当前云计算环境下客户的个性化, 智能流程的个性化要求决定了流程的动态变化性。围绕着智能流程的构建, 文中利用 π 演算这一进程代数语言作为智能流程建模描述语言。

文中首先介绍了智能流程平台架构的三大模块, 然后结合应用场景构建了虚拟智能流程, 利用 π 演算

对智能流程建模。为了适应用户需求的不断变化, 文中介绍了用户需求变更描述模型, 并给出了形式化表示方法。下一步的主要工作是完善 π 演算建立的智能流程模型, 并实现 π 演算模型到可执行程序的编译工作。

参考文献:

[1] 李 晖, 崔立真, 王海洋. 基于 Web 服务的智能流程构建方法[J]. 通信学报, 2009, 30(5): 128-135.

[2] 李维翠, 李德生, 褚东升. 基于社区云的按需个性化流程构建方法研究[J]. 计算机工程与设计, 2011, 32(6): 1906-1909.

[3] Milner R. 通信与移动系统: π 演算[M]. 林惠民译. 北京: 清华大学出版社, 2009.

[4] Zhang Chi. Web Service Interface Extending and Description Based on π Calculus[C]//2008 International Symposium on Computer Science and Computational Technology. [s. l.]: [s. n.], 2008: 203-206.

[5] 胡庆成, 邢春晓, 杨吉江. 基于 PI-演算的网上并联审批业务流程建模及验证[J]. 计算机应用研究, 2007, 24(12): 46-50.

[6] 袁文杰, 应 时, 吴可嘉, 等. 基于 π 演算的反射式需求规约描述方法[J]. 计算机工程与科学, 2010, 32(6): 146-148.

[7] van der Aalst W M P. The application of Petri nets to workflow management[J]. The Journal of Circuits, Systems and Computers, 1998, 8(1): 21-66.

[8] 杨玉梅, 刁永锋. 基于 UML 顺序图的 Petri 网建模[J]. 计算机技术与发展, 2007, 17(10): 130-134.

[9] 郁书好, 苏守宝, 刘仁金. UML 和 OWL 在本体建模中的比较研究[J]. 计算机技术与发展, 2007, 17(1): 155-157.

[10] van Dongen B F, van der Aalst W M P, Verbeek H M W. Verification of EPCs: Using Reduction Rules and Petri Nets[C]//Proceedings of Advanced Information Systems Engineering. [s. l.]: [s. n.], 2005: 372-386.

[11] 李建中, 陈良欲. 扩展的事件过程链方法及其在 BPR 的应用[J]. 系统工程, 2000, 18(1): 42-48.

[12] 姚 青, 洪余柯, 王海洋. 基于过程代数的可变业务流程建模方法[J]. 计算机集成制造系统, 2009, 15(9): 1721-1730.

(上接第 110 页)

应用[J]. 计算机工程与应用, 2005, 41(3): 108-110.

[12] Enck W, Ongtang M, McDaniel P. Understanding Android Security[J]. IEEE Security & Privacy, 2009, 7(1): 50-57.

[13] Schmidt A D, Schmidt H G, Batyuk L, et al. Smartphone Malware Evolution Revisited: Android Next Target? [C]//Proc. of the 4th IEEE International Conference on Malicious and Unwanted Software. [s. l.]: IEEE Computer Society, 2009.

[14] 胡 伟. Android 系统架构及其驱动研究[J]. 广州广播电视大学学报, 2010, 10(4): 96-101.

[15] 杨丰盛. Android 技术内幕[M]. 北京: 机械工业出版社, 2011: 101-132.

[16] 李 刚. 疯狂 Android 讲义[M]. 北京: 电子工业出版社, 2011: 382-385.

基于 π 演算的智能流程建模

作者：冯立元，李德生，李维翠

作者单位：冯立元, 李维翠(山东财经大学 计算机科学与技术学院, 山东 济南 250014)，李德生(山东财经大学 网络信息中心, 山东 济南 250014)

刊名：计算机技术与发展

英文刊名：Computer Technology and Development

年，卷(期)：2012(9)

本文链接：http://d.g.wanfangdata.com.cn/Periodical_wjfz201209030.aspx