

基于抽象状态机的普适服务组合分析与验证

易良辰, 黄林鹏

(上海交通大学 计算机科学与工程系, 上海 200240)

摘 要: 提出了一个普适环境下服务组合的框架, 使用抽象状态机对服务的行为进行不同精化层级的形式化描述, 然后利用 CoreASM 这一模型检测工具对服务组合进行模拟执行验证, 从而验证服务组合的正确性. 最后给出了一个运用此方法进行服务组合验证的典型应用场景.

关键词: 抽象状态机; 普适计算; 服务组合; 验证方法

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2013)03-0122-05

Analysis and Validation of Pervasive Service Composition Based on Abstract State Machine

YI Liang-chen, HUANG Lin-peng

(Department of Computer Science & Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

Abstract: A service composition framework of pervasive environment is presented in this paper. The behavior of services are defined by abstract state machine (ASM) in different level of refinement. To validate the correctness of the service composition, a model validation tool called CoreASM is used to simulate the execution. At last a study case is presented based on the above method.

Key words: abstract state machine; pervasive computing; service composition; validation methods

1 引言

抽象状态机 (Abstract State Machine, ASM) 是最早由 Yuri Gurevich 提出的一种多用数学模型, 后来逐渐演化成为一种形式化建模方法^[1], 并且已在许多领域中得到广泛的应用. ASM 可以对服务进行形式化建模, 通过对服务模型的逐步精化, 从而让不同精化层级的服务模型验证成为可能. 普适计算让计算无处不在, 随着无线接入技术的普及, 普适计算已经成为了一个研究热点. 所以在普适环境下通过 ASM 对普适服务组合进行建模和验证无疑是极具研究价值的课题.

本文首先给出了普适环境下的服务组合的一个典型的应用场景. 将不同的设备抽象为不同的服务, 并对这些服务用 ASM 进行建模. 同时提出一个服

务组合框架及服务组合其验证方法. 最后给出了一个案例分析, 利用文中提出的服务组合框架和建模方法分析一个智能医疗普适环境, 并验证其服务组合的正确性.

2 典型应用场景

本文通过对智能病房的应用需求作为示例阐述系统的各服务之间的联系以及相互之间的交互关系. 智能病房的应用需求如下: 每个病床旁边有一警报器, 当病人遇紧急情况时可按下警报器; 警报器一旦被处罚护士台会受到警报, 护士可将警报显示器关闭; 医生收到警报提示后会手持智能终端会发出通知, 手持智能终端会自动显示该病人的相关身体指标以及医疗记录; 医生来到病人处后智能终端自动记录医疗过程, 包括对话录音, 用药情况, 病症描述等; 病

收稿日期: 2012-06-05; 修回日期: 2012-07-21

基金项目: 国家自然科学基金项目(60970010); 国家重点基础研究发展计划项目(2009CB320705)

人身上的体征传感器会检测病人的身体状况,当检测到有体征数据处于危险范围时自动发出警报.

整个普适环境的设备由以下四部分组成:

- (1)氧气感知器、心跳感知器等体征感知器所构成的传感器网络
- (2)各类控制器,包括示例中提到的警报器,护士台的警报控制器等设备
- (3)人机交互设备,包括示例中提到的医生手中能够显示身体出现紧急情况的病人的身体实时数据、医疗历史记录和类似病例的手持智能终端等.
- (4)处理器设备,包括各类的基于网络的后台处理逻辑和规则.由感知器收到的数据应该如何处理,在怎样的条件下应该向控制器发送请求,向人机交互设备发送指令.

以上对于智能病房的场景分析和设备分类可以推广到其他的普适环境下,如智能家居,智能交通等普适环境.

3 服务模型

本节首先介绍了 ASM 的定义及其特点,并对设备和服务进行建模.论文^[2]中有对于物联网服务建模的相关工作,本节在此基础上,结合普适计算的特点,通过 ASM 对普适服务进行建模.

3.1 抽象状态机

抽象状态机(ASM)最早由 Yuri Gurevich 提出,可建模离散的动态系统,并用于连接计算模型和规格方法. EGON BORGER 有把它扩展成为一种系统的工程设计方法来指导软件的开发.^[3]

下面给出抽象状态机的定义:

定义 1 一个抽象状态机被定义为一个 4 元组 $M=(\Sigma, S, R, R_m)$. Σ 是方法名的有限集合, S 是 Σ 的初始状态的有限集合, R 是每个规则名的规则定义, R_m 是一个特殊的规则,称为 main rule.

变量/方法分为基本型和衍生型.基本型变量分为静态和动态变量.动态变量分为 in, out, shared, controlled. In 变量只能由 ASM 读取并且只能被环境所改变. Out 变量只能由 ASM 来写,并且环境对于 out 变量只读. Controlled 变量由 ASM 来读取和更新. Share 变量可被 ASM 来读写并且也可由其他的 ASM 或者环境来改变.要实现不同 ASM 之间的同步通信,可先由一方个 shared 变量进行赋值,然后另一个 ASM 访问这个 shared 变量.

3.2 设备建模

普适环境下的设备和实物可被视为实体,实体

有一些共同的特点.设备的网络构成了信息空间与物理空间的融合.文献^[4]中将这种融合分为两类,绑定以及自发的交互.

普适环境下的实体具有一些相同的特性,对这些特性进行抽象,并将这种抽象形式化地定义,具体定义如下所示.

定义 2 一个实体类表示为一个五元组, $E=(Eid, Op, Attr, Dom, M)$, 其中的 Eid 代表这个实体的唯一标识, Op 代表这个实体可以进行的操作集合; $Attr$ 代表 Op 操作可以改变的属性名称集合, Dom 是一个映射关系, $Attr \rightarrow DataValue$ 的一个映射,标识从属性集合到数据值的映射. $M=(\Sigma, S, R, R_m)$ 是一个符合定义 1 的 ASM,以此来描述该实体的行为.例如一个警报传感器实体的表示如下: $Alarm=(Alarm, \{push\}, \{sound\}, Dom_{al}, M_{al}, > . Dom_{al}(sound)=\{quiet, loud\}$, 一个警报传感器实体的抽象状态机 M_{al} 如图 1 所示.

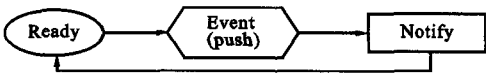


图 1 警报器抽象状态机示例

这个 ASM 中有一个抽象子状态机 Notify,可以根据具体的普适环境应用场景进行相应的精化.这里给出一个实例,抽象子状态机 Notify 的定义如下:

$Notify=staAlarm := on$
 $padPush := on$
 $patientID := detailedID$
 $sound := loud$

这个抽象状态机刻画的行为如下:警报传感器是病床附近病人可以控制的实体,当病人按下按钮时,会进入通知过程,接着再进入准备状态.通知过程子状态机会进行如下操作: $staAlarm := on$ 表示让护士台的警报灯进行显示, $padPush := on$ 表示值班医生的平板电脑上会出现警报推送, $patientID := detailedID$ 表示病人的 ID 号信息会通过变量 patientID 传递, $sound := loud$ 表示护士台和病人除都会收到声音的提示.

定义 3 一个具体的普适环境下的设备定义为一个三元组, $D=(Did, Eid, Oth)$, 其中的 Did 代表这个设备的编号,这个编号用于全局唯一地标识该设备. Eid 是这个设备所对应的实体类, Oth 表示该设备的一些具体特性,如其位置、生产年月等.

3.3 服务建模

普适环境下的服务通过与相应的设备之间进行

交互实现感知物理实体,控制各类设备的状态变换等功能.将这些的服务进行形式化的建模,服务组合框架如图 2 所示.

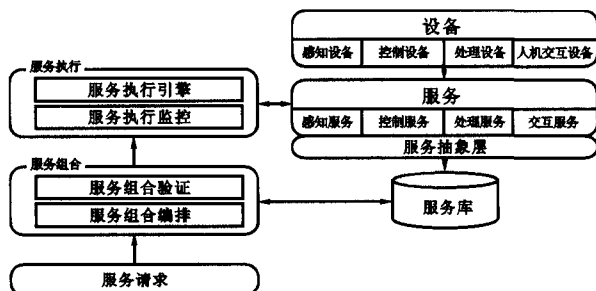


图 2 服务组合框架

定义 4 普适服务表示为一个三元组:

$S \equiv \langle Sid, Eset, M \rangle$, 其中的 Sid 是普适服务的 id, 是这个服务的标识. Eset 是这个服务的实体集合, 实体类的定义参见定义 2. M 标识这个普适服务的一个 ASM, 描述这个服务的具体行为. 根据上文提到的不同种类的设备, 对应的服务如下:

(1) 感知服务: 感知器设备一般只具有实体感知能力, 而不具备数据处理能力或者网络传输能力.

例如, 一个血压感知服务可以标识为 $\langle \text{PressureSensor}, \{\text{Pressure}\}, \text{PSM} \rangle$. PSM 代表这个血压感知服务行为的抽象状态机, 如图 3 所示, 代表一个血压传感器每隔 time 的时间读取一次传感器的值并且进行抽象子状态机 getValue 的操作, 最后重新进入 ready 状态.

getValue 抽象子状态机的定义为:

$\text{GETVALUE} \equiv \text{Channel} := \text{PressureCh!}$

$\text{pressure} := \text{value}$

这个抽象子状态机将信道设置为 PressureSensor, 并且在这个信道上将 Pressure 的值传递出去.

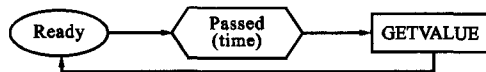


图 3 血压传感器抽象状态

(2) 控制服务: 控制服务对应的是大部分控制设备, 主要功能为对于实体进行操作行为. 此类服务主要的操作过程大多是首先接受一个指令, 在它的输入信道上接受一个变量, 然后通过输出信道上输出一个值.

(3) 处理服务: 处理服务所对应的设备是处理性设备, 主要功能是调用感知服务和控制服务. 例如, 因为感知器的功能大部分只有获取数据, 处理服务就负责将这些感知器所获得的数据进行抓取并且做相应的转发和处理工作. 这里抽象地表征为在不同

的信道上进行输入和输出.

(4) 人机交互服务: 人机交互服务对应的设备智能终端, 或类似设备. 例如平板电脑, 智能手机等. 人机交互服务不直接与感知服务和控制服务进行交互, 而是调用处理服务.

例如一个智能病房的控制终端 APP 的一个查看病人血压的服务定义 $\langle \text{SeePressure}, \{\text{ipad}\}, \text{UIM} \rangle$, 其中的 UIM 定义如下所示, 这个控制程序的操作者有两个可选项, 获取病人血压值和显示血压值.

$\text{UIM} \equiv \text{CM}$

PUIM

$\text{PUIM} \equiv \text{if Event}(\text{askPressure}(\text{Did})) \text{ then}$

step

$\text{EMIT}(\text{pulse}(\text{getPressure})),$

$\text{Value} := \text{pressureValue}$

$\text{notifyEvent}(\text{display}(\text{pressure}))$

else

$\text{DISPLAY}(\text{pressure})$

$\text{EMIT} \equiv \text{if Event}(\text{getPressure}) \text{ then}$

$\text{NOTIFYEvent}(\text{getPressure})$

4 服务组合框架

基于上文的服务建模, 本节提出的普适服务组合框架如图 2 所示. 不同的设备被抽象为相应的服务. 设备模块的定义参见上文的定义 3. 服务模块中每个服务都有对应的符合定义 4 的具体定义, 来刻画这个服务的相应行为, 从而在服务组合的验证阶段利用这个服务的形式化定义进行正确性验证. 本节将介绍该普适服务组合框架各模块的运行机制.

4.1 服务模块

如图所示, 服务模块可分为四种服务, 这四种服务已在上节进行过详尽的定义与阐述. 上一节中对这些服务做了形式化的定义, 从而在为服务组合模块中正确性验证提供可用的 ASM 模型. 但每个服务的具体的执行过程, 将对应具体的计算机编程实现或者用 BPEL4WS 来描述, 以便在服务执行阶段能够顺利地执行. 每个原子服务的执行流程对应一个符合定义 4 的定义, 得到其 ASM 刻画. 由于 ASM 的灵活性, 这样的刻画可以是针对这个服务不同层次的刻画. 例如对于一个控制自动开门的服务, 有一个抽象子状态机 OPEN, 对它的精化可以是简单的方法, 仅是将其状态设置为 OPEN. 也可是进一步精化, 例如在 OPEN 之前先将其门轴设定方向

(顺时针或者逆时针),然后打开马达使门轴开始旋转.

4.2 服务库

在服务库中保存了服务模块中的各类服务.下面给出服务库的定义:

定义 5 Repository 服务库被定义为所有服务组成的集合的一个子集,其中包括了原子服务 Sid 和服务组合 Scid. 后文会给出 Scid 的定义,它代表了在服务组合阶段,经过验证得到的服务组合编排会作为服务组合而不是原子服务存入服务库中.

4.3 服务组合模块

4.3.1 服务组合编排

服务组合编排模块以服务库中的原子服务作为基础,利用服务对应的多个 ASM 组成并发的服务,其形式化表示为:

定义 6 假设有 n 个普适原子服务 $s_1s_2, \dots, s_n$, 其中 $s_i (1 \leq i \leq n)$ 满足定义 4, $s_i \equiv \langle \text{Sid}, \text{Eset}, M_i \rangle$, 则这 n 个服务所构成的服务组合表示为:

$$SC \equiv \langle \text{SCid}, \bigcup_{i=1}^n, \text{Eset}, M_1 \parallel M_2 \cdots \parallel M_n \rangle$$

根据 ASM 的定义,各自动机并发执行相应的动作.并通过前文所述的变量,in, out, controlled, shared 变量进行 ASM 之间的通信.由于每个 ASM 分别刻画了原子服务的行为,所以其定义也对应了服务的行为定义.服务与环境或服务之间的通信通过 shared 共享变量进行通信达到并发执行的目的.服务内部通过 controlled 变量进行通信.服务之间的同步通信通过 in,out 进行.

4.3.2 服务请求

服务请求向服务组合模块发送相应的请求,服务请求的定义如下:

定义 7 Request $\equiv \langle \text{SCid}, \text{Dset} \rangle$, 其中 SCid 代表了服务的对应的在服务库中的 SCid, Dset 是设备实体 D 的集合,代表了服务请求所要用到的实际设备实体编号.

4.3.3 服务组合验证

基于抽象(ASM)的执行验证(Validation)在于通过抽象状态机模拟执行服务组合模型,从而在服务组合阶段验证该服务组合的各种属性.由于抽象状态机的灵活性,它可以基于不同层次的服务组合抽象进行相应层次的模型验证.现在主流的 ASM 执行工具有 C-based XASM^[5], .NET-executable ASML engine^[6], CoreASM Execution Engine^[7]. 本文将只用 CoreASM 模型执行工具. CoreASM 的设计具有灵活性和可扩展性.它基于一个 ASM 执

行内核,在此内核的基础上加入了插件以增加其他的功能.与其他的 ASM 工具相比,具有灵活,稳定,易用等特点.

服务组合的正确性^[8]可以分为状态可达性(reachability),系统安全性(safety),系统活性(liveness).

本文选用模型验证工具 CoreASM 作为本文服务组合的正确性验证工具.将由服务组合编排结果构成的 ASM 集合输入到 CoreASM 中后,用 CoreASM 来完成模拟验证工作.该工具会对服务的行为进行模拟,从而模拟出这些服务之间的交互过程,并观察数据的变化,包括属性值,数据指,状态等.监控状态机会对这整个过程中的运行轨迹进行记录.同时可以在模拟验证中加入性质检测状态机,从而检测某些不安全的状态是否会被达到,某些性质是否能够得到满足.

4.4 服务执行

本文的重点在于服务组合及服务验证阶段.针对服务执行模块,文章仅给出一个指导方法,但是对具体执行细节不做深入探讨.

4.4.1 服务执行引擎

根据服务组合阶段得到的服务编排结果,服务执行引擎将进行具体的执行过程.在执行过程中,引擎会动态地根据对应服务选择相应的设备进行执行.这里我们给出一个服务执行的抽象状态机但是对子状态机不做深入的精化.服务执行定义:

$$\begin{aligned} \text{ExecuteASM} &\equiv \text{Send} \\ &\quad \text{Execute} \\ &\quad \text{Receive} \end{aligned}$$

4.4.2 服务执行监控

在服务执行的过程中需要对服务的各属性进行监控从而判断各个要求的属性是否达到要求.在上一小节提到的服务组合验证阶段,利用 CoreASM 对服务进行验证的过程中.会加入验证监控子状态机,从而判断在模型模拟执行的过程中是否出现了错误的状态.

5 案例分析

通过定义每个服务的抽象状态机实现服务之间的模拟交互,利用 CoreASM 对服务组合进行模拟执行验证.由于篇幅所限,案例中各服务的 ASM 定义不在此处列出.以下为抽象状态机网络模拟执行验证的部分结果.

如图 4 所示是智能病房的血压传感器,体温传

感器,报警声音装置,报警器控制服务和智能终端的服务模拟执行的部分结果.图中描述的情景是:一个病人的体温检测到其体温过高,血压升高,于是体征服务通知警报器发出警报,同时通知医生.医生听到警报并且手上的平板电脑也受到通知,相关的病人数据和历史资料也随即显示出来.然后医生进入病房对病情进行记录.这个过程涉及到前文提到的四类服务.在验证过程中发现了一个错误状态,即医生在记录的时候警报器仍然会发出警报.通过修改该服务的ASM得到正确的新服务.这个案例说明了本文提出的服务组合验证方法的正确性和可行性.

PressureSensor display:normal
TemperatureSensor; display; temperatureIsNormal
SoundDevice; quiet
AlertState; NoAlert
TerminalState; ready
PressureSensor display:high
TemperatureSensor; display; temperatureIsTooHigh
SoundDevice; loud
AlertState; alerting
TerminalState; ready
PressureSensor display:hith
TemperatureSensor; display; temperatureIsTooHigh
SoundDevice; loud
AlertState; alerting
TerminalState; notify

图4 CoreASM 运行结果

6 结束语

普适计算让计算无处不在,它使人们把注意力从计算回归到任务本身.在普适环境下如何保证合理、正确、安全地利用各种设备所提供的服务已成为了现在的一个研究热点.本文针对普适环境下不同的设备特点和服务进行了建模,在此基础上提出了一个普适环境服务组合框架.并且利用抽象状态机理论对不同类型服务进行建模,对服务组合的结果通过模型检测工具CoreASM进行正确性验证.

在服务组合编排阶段,将这些潜在的服务组合的潜在危险排除对于普适环境下的服务组合具有很大的积极意义.

在未来的工作中,我们将进一步研究普适环境下服务执行的具体过程,执行细节的算法,并将本文提出的服务组合框架投入到实际的应用中.

参考文献:

- [1] Yuri Gurevich. Sequential abstract state machines capture sequential algorithms[J]. ACM Transactions on Computational Logic, 2000, 1(1): 77-111.
- [2] LI Li-Xing, JIN Zhi, LI Ge. Modeling and verifying services of internet of things based on timed automata [J]. Chinese Journal of Computer, 2011, 34(8): 1365-1377.
- [3] Gurevich Y. Evolving algebras[J]. A Tutorial Introduction. Bulletin of EATCS, 1991(43): 264-284.
- [4] Xu Guang-You, Shi Yuan-Chun, Xie Wei-Kai. Pervasive/ubiquitous computing [J]. Chinese Journal of Computers, 2003, 26(9): 1042-1050.
- [5] Anlauff M and Kutter P. Xasm open source[EB/OL]. [2012-04-20]. Web pages at <http://www.xasm.org>, 2001.
- [6] Foundations of software engineering group, microsoft research. asmL[EB/OL]. [2012-04-20]. Web pages at <http://research.microsoft.com/foundations/AsmL>, 2001.
- [7] Farahbod R. The CoreASM Project [EB/OL] [2012-04-20]. <http://www.coreasm.org>.
- [8] Xu Guang-You, SHI Yuan-Chuan, XIE wei-Kai. Survey on web services composition methods[J]. Computer Engineering, 2008, 34(4): 79-81.

作者简介:

易良辰 男, (1988-), 硕士研究生. 研究方向分布式计算、普适计算.

黄林鹏 男, 博士, 教授. 研究方向程序设计语言、并行与分布式计算、人工智能.