

普适环境中任务计算范例研究综述

蒋发群^{1,2} 李锦涛¹ 朱珍民¹ 罗海勇¹

(中国科学院计算技术研究所 北京 100190)¹ (中国科学院研究生院 北京 100039)²

摘要 为了减少用户注意力的耗费,实现普适计算以用户为中心的目标,从高层、抽象的用户任务出发的任务计算范例逐步引起了国内外研究者的关注,并被认为是实现“以用户为中心”的普适计算的可行方式。基于此,对国内外关于任务计算范例的研究现状进行总结和分析,并在此基础上对任务计算范例未来发展趋势进行了展望。

关键词 任务计算范例,普适环境,普适计算

中图法分类号 TP393 文献标识码 A

Survey of Task Computing Paradigm in Ubiquitous Environment

JIANG Fa-qun^{1,2} LI Jin-tao¹ ZHU Zhen-min¹ LUO Hai-yong¹

(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)¹

(Graduate School, Chinese Academy of Sciences, Beijing 100039, China)²

Abstract To minimize distractions on a user's attention and realize the user centric ubiquitous computing, task computing paradigm has gained increasing acceptance and been regarded as a promising way for ubiquitous environment. The existing related researches on task computing paradigm were analyzed and compared, and the summary of the future trend of the development of task computing paradigm was proposed.

Keywords Task computing paradigm, Ubiquitous environment, Ubiquitous computing

随着计算设备和网络的发展与普及,我们正朝着普适计算^[1,2]世界前进。普适计算的目标是将通信设备、计算机等构成的信息空间与人们生活和工作的物理空间融为一体,建立“以用户为中心”的普适环境^[3]。在普适环境中,一方面,用户所面对的资源日益丰富,计算将不再限于传统的桌面环境。另一方面,用户的移动性也日益增强,并且在移动过程中用户可能跨越不同的计算环境,需要动态地使用不同环境中的资源来完成相应的用户任务。然而,由于用户任务的抽象级别与当前计算设施支持的抽象级别(如应用、服务、设备等)存在着“鸿沟”^[4],使得用户在完成任务的过程中必须对与任务不相关的配置活动耗费太多注意力。因此,需要在普适环境中自上而下地构筑自动适应用户及其任务的计算范例,从而为用户对高层、抽象的用户任务与环境进行交互提供支持。

针对这个问题,国内外研究人员提出了一种新的计算范例——任务计算。任务计算范例试图在用户熟悉的抽象层上为其提供计算支持,包括用户涉及的任务、目标、活动和计划,从而弥补代表用户意图的任务和服务之间在抽象级上的鸿沟。随着普适计算研究的发展和进一步深入,任务计算范例在普适计算领域逐渐受到越来越多的关注和重视,如CMU(Carnegie Mellon University)的Aura项目^[5]、MIT(Massachusetts Institute of Technology)的计划驱动的计算(Plan-

Driven Computing, PDC)^[6]、基于活动的计算(Activity-Based Computing, ABC)^[7]、富士通研究院的任务计算(Task Computing, TC)^[8]等,并被认为是实现普适计算的一种可行方式。因此,研究任务计算及其关键技术具有重要的意义。

本文从任务计算范例研究的不同侧面对其进行了综述,如任务的建模与表示、任务的获取等,最后展望了任务计算范例未来的发展趋势。

1 任务计算范例的挑战

任务计算范例作为面向普适计算的一种新的范例,其关键思想是屏蔽和接管普适环境中涉及服务和设备的低层的管理活动,以致用户能够就其想要完成的高层、抽象的任务而不是如何完成这些任务的实际机制在普适环境中进行交互。在普适环境中,任务计算范例需要处理以下挑战^[9,10]。

1.1 普适环境的特征

普适环境的开放性、动态性、异构性、分布性等特点给任务计算范例的实现提出了新的挑战。一方面由于用户的移动性增强,用户任务的执行或完成也具有更强的移动性,并且可能要跨越多个环境;另一方面,完成用户任务的服务资源也具有更强的动态性、不确定性。

1.2 用户任务的抽象性

到稿日期:2008-12-15 返修日期:2009-03-05 本文受 863 国家计划项目(2006AA01Z112, 2006AA10Z253, 2007AA12Z321), 国家自然科学基金(60873244, 60772070)资助。

蒋发群(1977—),男,博士研究生,主要研究方向为普适计算、面向服务的计算等, E-mail: fqjiang@ict.ac.cn; 李锦涛(1962—),男,研究员,主要研究方向为普适计算、多媒体技术等; 朱珍民(1962—),男,教授,主要研究方向为普适计算、嵌入式系统等; 罗海勇(1967—),男,博士,高级工程师,主要研究方向为无线定位、普适计算等。

用户所涉及的任务(活动、目标、计划等)是普适环境中第一等级对象,反映了用户的意图。在任务计算范例中,任务是比普通环境中的服务具有更高抽象级别的概念。为了有效地完成用户任务,需要将用户任务映射到普通环境中合适的服务,因此需要构建合适的任务模型来表示抽象的用户任务。

1.3 用户任务的连续性

在普通环境中,用户的移动性日益增强。用户经常需要从一个智能环境移动到另一个智能环境。为了保证用户在不同的环境中能够继续进行自己的工作,需要保证任务的连续性,也可以称之为无缝性移动(Seamless Mobility)。

1.4 安全性

保证普通环境的安全性是普通计算研究领域一个非常重要的研究课题。在任务计算范例研究中,保证任务执行的安全也是一个必不可少的因素。如当前用户处于一个陌生的普通环境时,只有通过认证的服务才能够辅助该用户完成其任务。

2 相关研究分析

为了减少用户注意力的耗费,实现普通计算以用户为中心的目标,需要为用户就其关注的任务与环境进行交互提供支持。最近,为用户任务提供支持的普通计算范例在普通计算领域逐渐受到关注和重视,并得到认同。目前,国内外研究者从不同角度对任务计算范例进行了研究。

2.1 任务的建模与表示

随着普通计算的发展和任务计算范例研究的深入,研究者逐渐意识到,支持个性化的用户任务不仅仅取决于具体的服务发现技术、推荐技术等,还取决于与用户兴趣和偏好等密切相关的用户任务模型,而用户任务的建模尤其重要^[11]。

在任务计算范例中,任务建模是指以统一的、规范的方式描述普通环境中用户的任务需求,建立用户任务模型的过程。它是表达、理解和实现用户任务的关键所在。为了支持用户的移动性,CMU的研究者 Sousa 将用户任务建模为带有偏好的抽象服务的集合^[12]。这种任务建模方式扩展了早期研究将任务建模为应用的集合的观点,从而允许任务在不同的环境中利用不同的应用对其进行实例化。同时从对用户任务进行计算支持考虑,Sousa 使用 BNF 的语法规则和 XML 描述用户任务。但是,目前的任务只限于办公环境下移动用户使用多种简单应用程序(如编辑文档、浏览视频等)。在计划驱动的计算范例中,Gary Look 等人将用户目标抽象为计划,将计划作为用户任务的表示法,并以有向无环图来表示计划^[6]。随着任务计算范例研究的推进,语义 Web 技术被引入到任务建模和表示。陈恩义^[9]用实体关系模型建模普通环境中的任务,充分考虑了普通环境中任务与其它实体(服务、设备等)的关系,并用本体语言 OWL 描述任务本身及其与其它实体的关系。与此类似,Mokhtar 使用 OWL-S 本体标记语言建模用户任务,并支持语义表示、上下文推断和语义知识共享等^[13]。在此基础上,倪红波^[14]则从用户任务与上下文关系出发,提出了一种上下文依存任务建模方法,以上下文依存的层次思想来建模任务,其中每个任务能够进一步分解成一个子任务的集合或者原子任务实例。显然,在任务建模中引入语义关系来反映任务与其他实体的关系,将更加有利于支持复杂任务以及适应普通环境的动态性。

虽然不同的研究者根据各自的特定环境对任务进行相应的描述和表示,并且解决了任务建模的部分问题,但它们仍存在一定的局限性,难以满足任务计算范例的要求。例如,现有的任务建模研究没有描述任务的具体属性以及任务之间的关系,也没有讨论哪些上下文对任务更为重要。因此,建模普通环境中的用户任务仍然是任务计算范例的一个重要挑战。

2.2 任务获取

在普通环境中,能否正确地获取和感知用户任务,是任务计算范例的前提和基础。目前,任务的获取过程可以分为如下两类。

• 显示获取

这种获取方式主要通过用户定义和描述相应的任务,从而精确地获取用户任务。任务模板^[12]、服务请求语言^[15]以及任务描述语言^[16]是精确获取用户任务最常用的方式。这种显示获取过程需要用户直接参与,是一种相对静态的方式,只适合特定环境(如 EAI 环境)中相对稳定的任务。

• 隐式获取

这种获取方式主要是利用人工智能技术(如数据挖掘、机器学习等)根据关于用户、任务和服务的背景知识来“猜测、预测、推定”用户意图。例如 Isbell^[17]基于历史记录,利用 Markov 模型预测用户下一步将要执行的任务。倪红波^[18]利用一阶谓词逻辑进行任务推定,这种方式不需要用户的直接参与,只需要利用关于用户、任务和服务的后台知识,包括用户的历史信息以及领域专家建立的领域知识。

分析以上两种方式,第一种方式最简单、最直接,但用户参与过多。同时由于用户的意图通常是主观的和隐式的,因此不可能预先将它们进行完整描述。最关键的一点是在普通环境中希望终端用户愿意清晰地陈述意图以及所有必需的偏好,是不合理的。第二种方式的智能性、灵活性最高,但必须有强大知识库和人工智能技术的支持,同时其预测的准确度和效率还有待提高。

从人机交互的角度看,显示获取方式的目的是使用户“易交互”;隐式获取方式的目的是让用户“少交互”。因此,融合这两种方式优点的交互式任务获取方式可以改善普通计算交互过程,使任务获取具有更好的便捷性、智能性。由于上下文感知是提高人机交互智能性的重要途径^[19],因此交互式任务获取方式一方面利用上下文感知技术提高交互的智能性,从而减少显示的用户输入;另一方面用户在参与任务获取过程中能够根据上下文知识主动为用户推荐可选的后继任务以及任务的分解方案,可以更好地满足用户的个性化需求。通过交互式任务获取方法,用户的个性化需求或意图在知识库的支持下被交互式地创建、细化、分解,最终形成语义清晰、内容完备的用户任务模型,从而方便、准确地表达用户的需求。

2.3 任务映射

在任务计算范例中,高层、抽象的用户任务需要由普通环境中的服务资源来完成。因此,高层的用户任务需要与具体的服务资源进行关联,从而为用户匹配、选择最合适的服务来完成用户任务。在本文,我们将这个过程称为任务映射。任务映射是实现任务计算范例的一个关键。当前,研究者在任务计算范例实践中提出相应的任务计算映射机制。

Aura 项目中的任务映射机制主要利用理论效用模型在用户需求和环境中的能力与资源之间寻找最好的匹配。为了

使环境的效用最大化,任务映射过程不仅需要考虑到用户的偏好(如配置偏好、提供者偏好和服务质量偏好),而且需要考虑一些约束,如能力约束。为此,Sousa 将其抽象成一个最优化问题^[10]。在实际的求解过程中,任务映射问题被转换成多维度多重选择 0-1 背包问题^[20],从而利用最优化方法来求解。但是 Aura 没有考虑服务质量维度之间的相关性,而且需要依赖对应用的资源需求的精确预测。

Gaia 项目^[21]中的任务映射机制主要是通过一个语义发现过程来实现,以满足各种约束,如由开发者指定的参数约束、在本体中指定的约束和管理者为活动空间指定的策略等。在语义发现过程中,Gaia 将实体类的发现和实例的发现分开来处理,从而使发现过程更为灵活和有效。这意味着,为了选择合适的实体(设备、服务或应用等),首先发现满足类级约束的可能实体类,然后发现满足实例级约束的这些类的实例。在基于 Gaia 的任务执行框架^[22]中,整个发现过程包括下列步骤:(1)发现合适的实体类;(2)检查相似类的类级约束;(3)在当前活动空间中发现实体实例;(4)检查实例级的约束;(5)选择最好的实例。在选择最好的实例时,Gaia 也利用了多维效用函数对其进行评估,这一点与 Aura 类似。

在计划驱动的计算中,Gary Look 将计划作为用户任务的表示法^[5]。通常,对一个特定的任务存在多个相关的计划,同时满足同一个任务的计划可以具有不同服务质量,因此其任务映射机制通过计划选择机制来实现。为了选择合适的计划来执行,对候选计划进行成本效益分析,将选择具有最高效益成本比率的计划。其中,与计划相关的成本是指计划所使用的资源的成本;效益是由计划满足请求者偏好中指定的服务质量的程度来衡量。计划选择机制具体包括如下步骤:(1)动态地为每一个资源分配一个“成本”,反映其可用性以及任务可以消耗的资源值;(2)将用户的服务偏好转化为一个效用函数值,用来衡量特定的计划满足偏好的程度;(3)维持一个能够实现每个任务类型的计划库;(4)对于给定的任务请求和服务质量偏好,计算计划的成本效益,从而选择最符合用户的服务质量偏好以及对竞争资源使用最少的计划。

Masuoka^[8,24]等将任务映射机制主要分为两个阶段:服务发现阶段和服务组合阶段。其中,服务发现阶段使用 UPnP 技术,由服务发现引擎完成。与此类似,Mokhtar 也使用服务组合来实现任务映射^[13];陈恩义在服务发现阶段也使用 UPnP 技术,根据用户需求动态地选择服务^[9]。

上述研究中对任务映射主要考虑最大限度地满足用户的服务质量需求。在动态、开放的普适环境中,用户需要透明地使用服务资源。因此,任务映射对用户来说应当是透明的,用户不必关心其中的实现细节。

2.4 任务迁移

在普适环境中,用户的移动性日益增强。用户经常需要从一个智能环境移动到另一个智能环境。为了有效地支持用户移动性,即保证用户在不同的环境中能够继续进行自己的工作,以减少环境切换过程中用户的分心,国内外研究者开始关注任务迁移机制。任务迁移机制是指随着用户的移动,用户任务可以在多个具有不同性能的计算设备(如 PC/Laptop, PDA, Mobile phone 等)上或提供不同类型服务的计算环境中自由迁移,但不影响或者很少影响用户的使用和操作。

为了有效地支持用户移动性,国内研究人员试图通过迁

移应用来支持用户的移动性,这些应用被映射到本地设备。例如,在 Kimura 项目^[25]中,应用集合在一个智能房间中不同的显示设备之间进行迁移。南京大学的周宇^[26]提出一种基于代理的支持应用迁移的结构模型 MDAgent,并实现了相应的原型系统,以为应用的迁移提供移动管理、情境感知、资源匹配和推理机制等多方面的支持。在此基础上开发出若干应用并给出相关性分析。虽然通过应用的迁移能够部分支持用户的移动性,但分析不难发现,它们都将任务作为独立于特定设备的应用集合,难以满足普适环境中面向任务迁移时对无缝性、主动性的要求。与通过迁移应用来支持用户的移动性不同,Aura 通过迁移高层描述的用户任务来支持用户的移动性,这些用户任务将被映射到本地应用。在任务迁移过程中,Aura 将用户任务相关的所有信息迁移到一个新的环境,并通过新环境中的环境管理器来支持迁移的任务。为了支持任务迁移,用户任务及其相关信息都采用 XML 进行描述^[10]。为了满足普适环境中任务迁移对无缝性的要求,张德干等提出了基于 Mobile Agent 的任务无缝迁移策略,用于解决迁移过程中的迁移算法、迁移时延、迁移失效等问题。其特点在于基于合理的迁移粒度和更小的迁移时延,通过移动 Agent 的移动来证明迁移的无缝性^[27]。在此基础上,为了满足任务迁移对无缝主动性的要求,张德干等又提出了构件化无缝主动迁移机制中的资源调度策略,包括资源描述、映射和仲裁等策略^[28]。

在任务计算范例中,任务迁移的对象是高层、抽象的用户任务,同时需要应对普适环境的异构性、动态性等方面的挑战。显然,它与分布式计算线程/进程的迁移、移动计算中的平滑越区切换、无缝漫游不是同一层次上的概念。因此,任务无缝迁移问题是任务计算范例研究中一个极具挑战性的课题。

2.5 系统架构

为了适应普适环境的特点和用户的个性化需求,实现“以用户为中心”的目标,任务计算范例需要合适的架构支持。为了实现任务计算范例,普适计算领域的研究者根据相应的实践提出了一些系统架构。

Aura 架构^[5,29]是一个将用户任务概念作为第一类实体的系统架构,试图在时间和空间上支持跨越异构环境的任务管理。Aura 架构的主要目标是支持被建模为服务集合的用户任务以及支持用户在移动过程任务的连续性,从而可以超越现有的桌面管理技术,减少用户注意力的耗费。这个目标主要是由 Aura 架构中的任务层来实现。当环境、任务或上下文改变时,任务层利用效用理论模型在用户任务和环境中的能力、资源之间寻找最好的匹配,从而有效地使用资源。由于 Aura 架构的目的是代表用户按需、自动地配置计算环境,因此基于 Aura 架构的系统可以看作一个自适应的系统。

基于 Gaia 的任务执行框架^[22]是一个构建于通信中间件之上的系统框架,能够支持新任务的自动配置、自动修复和快速成型。它允许开发者和管理者就高层的、参数化的任务来编程和配置普适环境。此框架基于高级任务规范模型 Olympus^[30]并能为终端用户开发任务提供支持。任务参数的值可以由终端用户提供或者通过基于环境当前状态、上下文敏感策略和学习用户偏好的框架自动推断。与 Aura 架构类似,基于 Gaia 的任务执行框架关注普适环境的动态性,且为将应

用动态地映射到一个特定活动空间中的可用资源提供支持。

TCE的架构^[24]是一个层次化的系统框架,自底向上包含了4个层次:实现层、服务层、中间件层和呈现层。TCE系统架构将语义Web服务技术应用于普适计算场景,将环境中的功能抽象为语义Web服务,同时把环境中拥有的服务显示在Web页面上,从而为自动的发现、组合服务提供便利。与此类似,陈恩义在Smart Platform的基础上提出了面向任务计算的总体框架,从而能够在开放的普适环境中有效地执行用户任务^[9]。

ADCI(Activity Driven Computing Infrastructure)^[31]是一个活动驱动的计算架构,包含了4个松散耦合的子系统:定位和上下文感知子系统、活动管理子系统、活动发现子系统和客户端应用。该框架的显著特点是支持与任务处于同一抽象级别的活动,使得普适环境中的用户能够在设备之间进行切换,且在移动过程中能够维持他们当前的工作环境。相对于Aura中的任务,ADCI中的活动显得更为具体,主要是对医疗保健环境中的病历浏览、登记和开处方等^[32]。

基于计划的系统框架^[6,23]认为终端用户愿意清晰地陈述所有必需的偏好是不合理的,该框架由4个主要的部件组成:计划选择器、资源管理器、计划执行器和计划监控器。该框架采用了人工智能中类似规划的方法,在资源的基础上构建了计划库。用户计划的执行过程即目标的解析过程,就是根据一定的策略,如当前的上下文信息、用户的习惯和偏好等,选择特定的计划来执行。

由于研究角度和侧重点的差异,上述支持任务计算范例的系统架构具有各自的特点,并且解决了任务计算范例的部分问题。大量的研究和实践表明,面向服务的计算和中间件技术适宜解决普适计算领域的挑战性问题,并在一些普适计算系统^[33-35]中得到了验证。因此任务计算范例的系统框架需要融合面向服务的计算和中间件技术,从而有利于解决任务计算范例所面临的挑战性问题。

根据普适环境中任务计算范例应解决的挑战性问题对现有的相关工作进行一个简单的对比,如表1所列。

表1 任务计算范例相关工作比较

	普适环境的 动态性	普适环境的 异构性	用户任务的 抽象性	用户任务的 连续性	安全性
Aura	✓	✓	✓	✓	
ABC			✓	✓	✓
PDC	✓	✓	✓		
TC	✓	✓			
Gaia	✓	✓	✓		✓

结束语 随着普适计算研究的发展,在用户熟悉的抽象层面上提供计算支持的任务计算范例逐渐受到学术界和工业界的关注和重视。目前国内外研究者对任务计算范例的研究主要集中在任务建模与表示、任务映射、任务迁移等方面。这些研究都试图寻求简单、有效、可扩展的任务计算技术,为用户任务的实现提供自动化、智能化支持,从而最终实现以用户为中心的普适计算模式。

虽然国内外研究者对任务计算范例进行了研究,取得了一些可以作为借鉴的经验,但任务计算范例依然面对众多挑战,如任务建模需要统一的标准、缺乏组织和管理服务的基础结构等。因此,任务计算范例仍然需要进一步的深入研究。一方面需要融合各种新思想、新理论、新技术,如语义Web技

术、网络技术、人工智能等,提炼出面向任务计算的方法学与支撑技术体系,才能实现任务计算的理想目标;另一方面,以普适计算领域作为载体,构建相应的任务计算系统,通过具体的应用实践来促进任务计算范例的研究与应用。研究任务计算范例具有提升作用的关键性理论和技术,并应用于普适计算系统开发以及普适环境的构建,将有利于推动普适计算的跨越式发展。随着任务计算研究的不断深入和技术的不断发展,任务计算范例所涉及的各种问题最终必将得到解决,并且将逐步应用到人们的工作、学习、生活环境中,为人们提供更有效的服务。

参 考 文 献

- [1] Weiser M. The Computer for the 21st Century[J]. Scientific American, 1991, 265(3): 94-104
- [2] 郑增威, 吴朝晖. 普适计算综述[J]. 计算机科学, 2003, 30(4): 18-22
- [3] 徐光佑, 史元春, 谢伟凯. 普适计算[J]. 计算机学报, 2003, 26(9): 1042-1050
- [4] Bardram J E, Christensen H B. Open Issues in Activity-based and Task-level Computing[C]// Proceedings of the First International Workshop on Computer Support for Human Tasks and Activities. University of Aarhus, Aarhus, Vienna, Austria, 2004
- [5] Garlan D, Siewiorek D P, Smailagic A, et al. Project Aura: Toward Distraction-free Pervasive Computing[J]. IEEE Pervasive Computing, 2002, 1(4): 22-31
- [6] Look G. Plan-based Proactive Computing[D]. Massachusetts Institute of Technology, 2003
- [7] Bardram J E. Activity-based Computing: Support for Mobility and Collaboration in Ubiquitous Computing[J]. Personal and Ubiquitous Computing, 2005, 9(5): 312-322
- [8] Masuoka R, Parsia B, Labrou Y. Task Computing-the Semantic Web Meets Pervasive Computing[C]// Proceedings of the Second International Semantic Web Conference. Sanibel Island, Florida, USA, 2003: 866-881
- [9] 陈恩义. 面向任务计算的研究[D]. 北京: 清华大学, 2005
- [10] Sousa J P. Scaling Task Management in Space and Time: Reducing User Overhead in Ubiquitous-Computing Environments[D]. Pittsburgh PA: Carnegie Mellon University, 2005
- [11] Ingmarsson M. Modelling User Tasks and Intentions for Service Discovery in Ubiquitous Computing[D]. Sweden: Linköping University, 2007
- [12] Sousa J P, Poladian V, Garlan D, et al. Task-based Adaptation for Ubiquitous Computing[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C, 2006, 36(3): 328-340
- [13] Mokhtar S B, Georgantas N, Issarny V. Ad Hoc Composition of User Tasks in Pervasive Computing Environments[C]// Proceedings of the 4th Workshop on Software Composition. 2005: 31-46
- [14] Ni H B, Zhou X S, Zhang D Q, et al. Context-dependent Task Computing in Pervasive Environment[C]// International Symposium on Ubiquitous Computing Systems. Seoul, Korea, 2006: 119-128
- [15] Aiello M, Papazoglou M, Yang J, et al. A Request Language for Web-services Based on Planning and Constraint Satisfaction[C]// Lecture Notes in Computer Sciences. Springer, 2002: 76-85

(下转第28页)

- entiated service; modeling and performance analysis [A]// Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings, 8th International Symposium on [C]. Book, News, Inc, Portland, OR; IEEE, 2000; 441-448
- [2] 林闯. 一种资源共享系统的模型和近似性能分析[J]. 计算机学报, 1997, 20(10): 865-871
- [3] Balbo G. Performance issues in parallel programming [A]// Application and Theory of Petri Nets 1992, Lecture Notes in Computer Science [C]. Springer Verlag, 1992, 616: 1-23
- [4] Kensing F, Blomberg J. Participatory design: Issues and concerns [J]. Computer Supported Cooperative Work, 1998, 7(3): 167-185
- [5] Cabodi G, Camurati P, Quer S. Reachability analysis of large circuits using disjunctive partitioning and partial iterative squaring [J]. Systems Architecture, 2001, 47(2): 163-179
- [6] Zuberek W M, Rada I. Modeling and analysis of distributed state space generation for timed Petri nets [A]// 342th Annual Simulation Symposium(SS-2001) [C]. Stattle, WA: The Society for Computer Simulation International (SCS), 2001; 93-98
- [7] 林闯. 一种随机 Petri 网性能等价化简与分析方法[J]. 电子学报, 2002, 30(11): 1620-1623
- [8] Ciaodo G, Muppala J, Trivedi K S. SPNP; Stochastic Petri Net Package [C]// Proc. Petri Nets and Performance Models, 1989; 142-151
-
- (上接第 13 页)
- [16] Tran V X, Tsuji H. OWL-T: A Task Ontology Language for Automatic Service Composition [C]// Proceedings of the Sixth International Semantic Web Conference. Salt Lake City, Utah, USA, 2007; 1164-1167
- [17] Isbell C L, Omojokun O, Pierce J S. From Devices to Tasks: Automatic Task Prediction for Personalized Appliance Control [J]. Personal and Ubiquitous Computing, 2004, 8(3): 146-153
- [18] Ni Hongbo, Zhou Xingshe, Yu Zhiwen, et al. OWL-based Context-dependent Task Modeling and Deducing [C]// Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops. Niagara Falls, Canada, 2007; 846-851
- [19] 岳玮宁, 王悦, 汪国平, 等. 基于上下文感知的智能交互系统模型 [J]. 计算机辅助设计与图形学学报, 2005, 17(1): 74-79
- [20] Poladian V, Sousa J P, Garlan D, et al. Dynamic Configuration of Resource-aware Services [C]// Proceedings of the 26th International Conference on Software Engineering. Edinburgh, 2004; 604-613
- [21] Román M, Hess C K, Cerqueira R, et al. Gaia: A Middleware Infrastructure to Enable Active Spaces [J]. IEEE Pervasive Computing, 2002, 1(4): 74-83
- [22] Ranganathan A. A Task Execution Framework for Autonomic Ubiquitous Computing [D]. Urbana-Champaign: University of Illinois, 2005
- [23] Look G, Peters S, Shrobe H. Plan-driven Ubiquitous Computing [C]// Proceedings of the Workshop on AI in Mobile Systems (AIMS) at the International Conference on Ubiquitous Computing. Seattle, Washington, USA, 2003; 66-73
- [24] Song Z X, Labrou Y, Masuoka R. Dynamic Service Discovery and Management in Task Computing [C]// Proceedings of first Annual International Conference on Mobile and Ubiquitous Systems. Boston, Massachusetts, USA, 2004; 310-318
- [25] MacIntyre B, Mynatt E, Voida S, et al. Support for Multitasking and Background Awareness Using Interactive Peripheral Displays [C]// Proceedings of ACM User Interface Software and Technology. Orlando, Florida, 2001; 41-50
- [26] 周宇, 马晓星, 曹建农, 等. 普适环境下基于软件代理虚拟化的应用迁移 [J]. 软件学报, 2007, 18(8): 2038-2048
- [27] 张德干, 尹国成, 史元春, 等. 普适环境中的无缝迁移策略 [J]. 控制与决策, 2005, 20(1): 6-10
- [28] 张德干, 班晓娟, 曾广平, 等. 构件化无缝主动迁移机制中的资源调度策略 [J]. 计算机学报, 2006, 29(11): 2027-2036
- [29] Sousa J P, Garlan D. Aura: An Architectural Framework for User Mobility in Ubiquitous Computing Environments [C]// Proceedings of the IFIP 17th World Computer Congress - TC2 Stream/3rd IEEE/IFIP Conference on Software Architecture. Montreal, Canada, 2002; 29-43
- [30] Ranganathan A, Chetan S, Al-Muhtadi J, et al. Olympus: A High-level Programming Model for Pervasive Computing Environments [C]// IEEE International Conference on Pervasive Computing and Communications. Kauai Island, Hawaii, 2005; 7-16
- [31] Bardram J E, Christensen H B, Olsen A K. Activity-driven Computing Infrastructure-Pervasive Computing in Healthcare [C]// Proceedings of the Conference of Pervasive Computing, 2004
- [32] Bardram J E, Christensen H B. Pervasive Computing Support for Hospitals: An Overview of the Activity-based Computing Project [J]. IEEE Pervasive Computing, 2007, 6(1): 44-51
- [33] Bellur U, Narendra N C. Towards Service Orientation in Pervasive Computing Systems [C]// Proceedings of International Conference on Information Technology: Coding and Computing. Las Vegas, Nevada, USA, 2005; 289-295
- [34] Shehzad A, Ngo H Q, Lee S Y, et al. A Comprehensive Middleware Architecture for Context-aware Ubiquitous Computing Systems [C]// Proceedings of IEEE Fourth Annual ACIS International Conference on Computer and Information Science. Jeju Korea, 2005; 251-256
- [35] Mokhtar S B. Semantic Middleware for Service-oriented Pervasive Computing [D]. INRIA-Rocquencourt, France, 2007