

# 上下文感知计算及系统框架综述

李蕊 李仁发

(湖南大学计算机与通信学院 长沙 410082)  
(lirenfa@vip.sina.com)

## A Survey of Context-Aware Computing and Its System Infrastructure

Li Rui and Li Renfa

(School of Computers & Communications, Hunan University, Changsha 410082)

**Abstract** Contexts like gestures, emotions, situations are very useful when a person interacts with other persons or nearby situation. On the contrary, computer can not use contexts efficiently. But computing-aware computing now is becoming more and more important since pervasive computing was put forward by Mark Weiser in 1991. The research of context-aware computing, especially its system infrastructure, is surveyed. The main contribution of this paper is an innovative system infrastructure conceptual model and the key problems of the model are discussed in detail.

**Key words** context-aware computing; context; system infrastructure; pervasive computing

**摘要** 当人们与其他人或周围的环境交互时,常常会无意识地利用到诸如手势、眼神、情境、环境状况等上下文信息。与之相反,计算机系统却很少能有效地利用这些信息。但自 Mark Weiser 提出普适计算以来,如何有效利用上下文信息开始得到充分重视并发展成为一种新的计算模式——上下文感知计算。通过对近 10 年来上下文感知计算尤其是系统框架的发展进行综述和分析,提出了一个新的上下文感知计算系统框架概念模型,同时对该概念模型所涉及的关键技术问题进行了深入探讨,这可为感知计算的进一步研究提供指导。

**关键词** 感知计算;上下文;系统框架;普适计算

中图法分类号 TP391;TP311

当人们与其他人或周围的环境交互时,常常会无意识地利用到诸如手势、眼神、情境、环境状况等上下文信息,并依此进行推断得出结论后作出适当的反应;与之相反,计算机系统却很少能有效地利用这些信息。首先,这是因为计算机无法在无意识的状况下利用上下文,所使用的上下文信息必须被明确指定;其次,计算机所能感知的上下文信息有限,且一般是低层上下文信息,对于人类行为、心理趋势等高层上下文信息必须通过复杂的建模、演绎和推理过程;最后,上下文感知系统框架的缺乏使得开发有效利用上下文信息的应用非常困难。这也导

致了上下文信息的有效利用和上下文感知计算长期被忽视。

20 世纪 90 年代初,Weiser 提出了普适计算<sup>[1]</sup>。作为其核心子领域之一的上下文感知计算开始得到充分关注,其研究主要包括如何获取上下文,如何对上下文进行建模和表示,如何有效利用上下文,如何构建支持上下文感知的系统框架等。许多大学和研究机构都提出了专门项目并对此进行了深入研究。

本文分析了上下文建模、感知、存储和有效利用等方面的研究现状,集中讨论了上下文感知计算的系统框架,指出了软件框架设计所存在的关键技术

问题. 本文的主要贡献在于提出了一个新的上下文感知计算系统框架概念模型, 并对该概念模型所涉及的关键技术进行了深入探讨, 可为进一步研究提供指导.

下面本文将就上下文感知计算进行分析, 然后集中讨论上下文感知计算系统框架的研究现状、概念模型及其存在的关键问题等, 最后对全文进行总结.

## 1 上下文感知计算

上下文感知计算是指系统能发现并有效利用上下文信息(如用户位置、时间、环境参数、邻近的设备和人员、用户活动等)进行计算的一种计算模式. 下面将就其定义、建模、感知、存储、如何有效利用等研究问题分别进行论述, 感知计算的系统框架则集中在第2节进行讨论.

### 1.1 上下文的定义

尽管上下文在某些环境中显而易见, 如某次谈话中谈话的时间、地点和氛围等很容易成为交谈双方确知的上下文, 但要给感知计算中的上下文下一个通用的定义却相当困难<sup>[2]</sup>. 早期的研究者多通过枚举方式来定义上下文, 如 Schilit 就将上下文分为3类来定义<sup>[3]</sup>:

- ① 计算上下文, 如网络的可用性、网络带宽、通信开销、周边的打印机、显示器等资源;
- ② 用户上下文, 包括用户的个性、位置、周围的人员、甚至社会关系等;
- ③ 物理上下文, 如光线的明暗、噪声的大小、交通状况、气候、温度等.

显然, 这样的定义既不够规范, 也容易产生混淆. 参照文献[2], 我们给出了更通用的定义: 上下文是环境本身以及环境中各实体所明示或隐含的可用于描述其状态(含历史状态)的任何信息. 其中, 实体既可以是人、地点等物理实体, 也可以是诸如软件、程序、网络连接等虚拟实体."

### 1.2 上下文的建模与表示

由上下文的定义可知上下文可以是描述实体状态的任何信息, 其种类丰富、数量众多. 在所有这些上下文中, 位置上下文是使用频率最高的, 其建模方式主要有坐标位置模型、符号位置模型和混合模型. 坐标位置模型是指通过将物理空间划分为区格来建立坐标系并使用元组来惟一精确地表示特定位置的方式. 如 GPS(全球定位系统)通过(经度, 纬度,

海拔)三元组来表示位置信息. 符号位置模型则是通过抽象的位置符号(名称)以及位置之间的相对关系来表示位置的一种方式. 如通过湖南大学、办公楼、东500米可表示某个指定位置;混合模型则是结合上述两种模型来表示位置的方式, 它的产生是因为坐标位置模型虽然可以精确地表示位置, 但很难直接推导出空间关系, 某些环境下甚至可能失效(如GPS不能在室内使用等);而符号位置模型又缺乏精确的位置信息所致. 除位置上下文外如温度等物理上下文、时间上下文、虚拟上下文等都可以建模, 但建模方法依据应用的不同而不同. 与此对应, 上下文的表示(数据结构描述)也依据应用而有所不同, 主要方法包括键值对、标记语言、面向对象的模型、逻辑模型等<sup>[4]</sup>.

显然, 上下文的建模和表示与应用相关的特点不利于上下文信息的交互和共享. UML扩展是进行上下文统一建模的途径之一, 如文献[5]提出了一种基于UML的面向上下文感知Web服务的建模语言ContextUML;文献[6]则通过扩展UML 2.0可支持对表示、上下文和活动等的高层建模;文献[7]依据UML扩展提出了通用的上下文结构模型, 并支持对上下文和活动之间的关系进行建模, UML建模语言和模型驱动开发(model-driven)的广泛采用使这种方法非常具有吸引力. 另一个极具希望的方法是本体论, 显然, 从语义层面对上下文建模将更有利于互操作和使用<sup>[8-9]</sup>, 文献[10]提出了一个包含增强词汇模型的上下文本体用于界面中上下文和动作信息的导航, 其中, 本体结构被定义为一组属性, 而本体词汇表则可依据领域和应用的需要进行设计;文献[11]中使用OWL本体标记语言建模上下文, 并支持语义表示、上下文推断和语义知识共享等;文献[12]中提出了一种基于RDF(resource description framework)的CSCP语言来建模上下文.

此外, 对整个上下文感知系统的建模也处于初步研究之中. 其方法主要包括有限状态机<sup>[13]</sup>、行为系统<sup>[14]</sup>和有有色Petri网<sup>[15]</sup>等.

总之, 上下文的建模和表示是上下文感知计算的第1步, 在上下文感知计算中占有重要的位置, 其研究具有重要意义.

### 1.3 上下文的获取与感知

能否正确地感知环境中的上下文是上下文感知计算的前提. 与上下文建模一样, 上下文感知方式依据类型和应用的不同而不同. 如低层上下文中的位置信息在户外可以通过GPS来获取, 在户内由于

GPS 的信号较弱无法穿透建筑物,可通过红外、射频、超声波等方式来获取。其他的低层上下文也可通过比较简单直接或仅结合少量运算的方式得到。如时间可以通过设备的内建时钟得到,方向和加速度可由相应的传感器直接感知,网络带宽则可以通过动态检测,邻近对象可通过位置间的相互关系来获取等。与低层上下文不同,高层上下文的获取要复杂得多。如通过计算机视觉进行视频跟踪和图像处理来感知用户的当前活动,运用人工智能技术由若干低层上下文联合演绎高层上下文并对当前的情境和行为作出判断<sup>[16-17]</sup>,对上下文进行预测和分析<sup>[18]</sup>等。

理想情况下所获取的上下文应该是精确、有效、恰当、无冗余和无冲突的。但实际上,由于同一上下文可以从不同的信息源通过不同的方式获取,获取上下文时间间隔以及上下文本身的精确度、可靠性、有效性等限制导致获取的原始上下文经常发生冲突,存在不完整和有效性等问题<sup>[18-20]</sup>。通过将上下文分为感知、静态、配置和派生等 4 种类型,文献[21]提出了支持不完整和不精确上下文建模的 ORM(object role modeling)语言,该语言不仅支持对上述 4 类上下文的建模,还支持对候选上下文、信息质量、未知上下文、历史上下文以及依赖关系等的建模。文献[22]基于面向对象的上下文模型提出了一个两阶段的上下文过滤流程,并可依据当前用户上下文将恰当的信息提供给用户。文献[23]通过综合上下文本体和模糊推理允许对上下文进行非确定性推理。文献[24]则提出了一个支持模糊上下文获取和处理的系统框架。

#### 1.4 上下文的存储、查询和管理

上下文固有的特点为上下文有效而可靠地存储、查询和管理带来了巨大的困难,如上下文的瞬时性、上下文以及上下文与对象之间的关联性特征和上下文查询通常以范围而不是单一值为条件等。尽管上下文存储在上下文感知中占有重要位置,是上下文感知计算实用化的关键技术之一,但目前相关研究却比较少。Gaia 提出了一种上下文感知文件系统 CFS,其中上下文以目录方式表示,路径对应于上下文的类型和值,上下文查询采用一阶谓词方式(如 Context(temperature,room3231,is,98F)等)并支持复杂的组合查询<sup>[25-26]</sup>。文献[27]基于关系数据库建立了上下文存储系统 CBS,可支持上下文的分布式存储、同步和高效查询,但依赖关系数据库限制了上下文的建模和表示。

#### 1.5 上下文的有效利用

除建模、感知、存储外,上下文如何被有效地利用是一个更重要的问题。Schilit 将上下文感知计算分为就近对象选择、自动配置、上下文信息和命令、上下文触发等 4 类。而 Dey 则将其分为适时地显示信息和服务、自动执行服务、标记信息等。即便如此,也未能完整描述上下文信息的全部已知用途。如在 CASIS 系统中通过使用设备状态、光线明暗、用户位置、方向、活动历史等上下文可用于降低语音识别系统的错误率并处理二义性<sup>[28]</sup>。文献[29]则利用位置、时间、活动等上下文用于增强小组交互和协作等。Williamson 等人甚至利用了上下文感知对 TCP/IP 协议进行优化<sup>[30-31]</sup>。

还有许多的研究者认为有必要整合虚拟(计算)上下文和物理上下文以更好地理解用户的行为<sup>[32-33]</sup>,但目前绝大部分程序都是分开感知和处理的。Römer 等人指出 RFID 是弥补物理与虚拟世界鸿沟的简单且有效的技术,同时还提出了虚拟配偶(virtual counterparts)的概念<sup>[34]</sup>;Kimura 系统则更进一步地实现了物理与虚拟世界的相互整合和协作<sup>[35]</sup>。

由上述可知,开发上下文感知应用将面对种类繁多的上下文和方式迥异的感知策略<sup>[36]</sup>。将低层信息处理与高层应用开发分开,通过引入软件框架(或中间件)来处理原始上下文信息的采集、推断、格式的转换、信息的传播等以降低应用开发的复杂度是目前比较通用的方法,第 2 节我们将就此问题进行较深入讨论。

### 2 上下文感知计算的系统框架

上下文感知计算的系统框架主要负责对设备、上下文、物理环境等构成的计算环境进行管理、协调和调度,建立实体对象间互操作(interoperation)的基础,同时屏蔽计算环境的复杂、多样和动态性,为应用开发提供统一的框架和应用程序接口(API)。

本节内容按如下方式组织:首先分析上下文感知系统框架的研究现状,然后提出一个通用的系统框架概念模型,最后就系统框架设计的关键技术问题进行讨论。

#### 2.1 系统框架的研究现状

系统框架的研究相当丰富,当前已有大量的上下文感知计算系统框架。如 Gaia<sup>[25]</sup>是构建于通信中间件之上的系统框架,除了抽象级别不同外,它非

常类似传统操作系统. Gaia 通过将智能空间及所包含的资源抽象为可编程实体,并提供程序执行、IO 操作、文件系统管理、通信、错误检测、资源分配等六大传统 OS 所具有的服务,可为方便地构建以用户为中心、多设备、上下文感知的移动应用程序提供支持.

Aura 是一个基于多代理( multi-agent )的框架,当环境、任务或上下文改变时,任务管理器将在无需用户干预的前提下进行资源映射并有效使用资源. ACAI<sup>[37]</sup>是另一个基于多代理的框架,主要包括上下文管理器代理、协调者代理和本体代理 3 个核心代理以及上下文提供者代理等. 其他基于多代理的框架还有 BerlinTainment<sup>[38]</sup>,CARMEN<sup>[39]</sup>.

CAPpella<sup>[40]</sup>框架认为只有终端用户才最清楚其活动、周围环境及隐含知识,因此该框架试图结合机器学习和 PBD( programming by demonstration )来帮助最终用户构建上下文感知应用. 与此类似的还有 Korpipaa 等人提出的框架,此框架基于黑板模式并能提供终端用户开发( end user development )支持<sup>[41]</sup>.

ActiveCampus<sup>[42-43]</sup>是一个典型的基于客户/服务器架构的框架,该框架的显著特点是支持组件间的职责划分和服务间的高度整合. CARISMA<sup>[44]</sup>框架则采用了反射(类似于操作系统的回调函数)模型,通过反射,该框架允许监视框架的内部行为并在需要时进行修改.

CAPNET<sup>[45]</sup>是支持移动多媒体应用的感知计算框架,该框架包括连接管理、组件管理、服务发现、消息等核心组件以及用户接口、媒体、上下文和上下文存储等,其特点是支持组件移动,组件可依据应用对资源的需求既可以在客户端也可以在服务端启动执行以解决移动终端资源受限的问题.

SOCAM<sup>[46]</sup>是一个面向上下文感知服务且包含基于本体的上下文模型的系统框架,该框架可高效地支持上下文的获取、发现、解释和访问等,整个框架主要包含上下文提供者、上下文解释器、上下文数据库、上下文感知服务和定位服务等部分, SOCAM 的另一个特点是基于 OSGi( open service gateway initial ),OSGi<sup>[47]</sup>是用于家庭驻留网关的开放工业标准,也是潜在的普适计算标准之一. CASM<sup>[48]</sup>是另一个面向服务的上下文感知框架,它由设计时和运行时两个部分组成,设计时主要包括传感器、环境、用户、任务和服务等的建模,而运行时则包括上下文管理器、事件通知系统和任务引擎等.

iCAP<sup>[49]</sup>是支持上下文感知应用交互原型快速构建的工具,可用于对上下文感知应用进行建模和评估. Topiary<sup>[50]</sup>则是另一个类似的工具,但主要面向位置上下文感知应用程序.

CASA 是一个具有安全保证的上下文感知框架,该框架通过采用包含环境角色( environment role )的通用角色访问控制( GRBAC )模型可提供隐私和安全保证.

从以上可以看出,现有的系统框架主要是基于特定应用而提出来的,其适应范围有限. 即使 Gaia 已经提出了类操作系统的概念,但仍很难满足感知计算的全部要求. 构建上下文感知计算系统框架不仅仍是一个挑战,也是一个机遇.

2.2 系统框架概念模型

由第 2.1 节可知,尽管系统框架的关注面和应用范围不同,但通常都包含上下文感知、上下文转换、触发执行等部分. 为了更好地了解系统框架的组成结构及各部分的功能划分,我们提出了一个通用的系统框架概念模型(如图 1 所示).

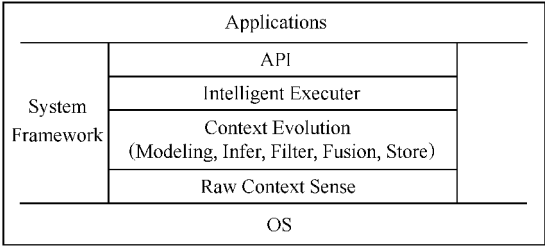


Fig. 1 Context-aware computing conceptual model.  
图 1 上下文感知计算概念框架

整个模型由 4 部分组成,分别是:

1) 上下文感知( raw context sense ). 上下文感知层负责从分布广泛的传感器中采集原始上下文并进行初步处理,使上下文感知与实际使用相分离. 一般来说,原始的上下文是模糊、不精确、不稳定甚至包含冲突的. 这主要是因为:首先,传感器的精度是有限的;第 2,同一上下文可由多个不同的传感器感知,为潜在冲突带来了可能性;第 3,上下文信息是高度分布的,可来源于任何时间和位置;最后,原始上下文所能提供的都是低层和初步的信息. 所有这些都给上下文的感知和处理带来了巨大的困难,因此有必要在感知层的基础上进行上下文的演化.

2) 上下文演化( context evolution ). 上下文的建模、过滤、推断、融合和存储构成了上下文演化的主要内容. 其目标是通过原始上下文进行过滤、推断和融合等得到各应用所需的高层上下文. 上下文

演化的另一个目标是构建统一的上下文模型以支持上下文互操作,并允许上下文以统一的方式自由传输。借助这些手段,简单的低层上下文通过联合演绎就可以得出应用所需的统一格式的高层上下文。

3) 智能执行体(intelligent executer)。智能执行体是本框架的核心,主要包括对感知触发、互操作、自适应策略、自配置和自组织技术等的支持。通过智能执行体的帮助,用户可以获得增强的用户体验,切实感受到感知计算所带来的好处。如自发的互操作可使用户免受打扰;自适应策略则允许用户自由地切换键盘、手写、语音等输入方式;自配置技术则可使用户远离手工配置的困扰。

4) 应用程序接口(API)。应用程序接口负责提供程序开发接口,以方便开发者充分利用框架的功能并快速构建感知应用。应用程序接口不仅包含框架所能提供的服务也应包含应用开发所应遵循的规范。

### 2.3 系统框架的关键问题

在上一节的概念模型的基础上,我们认为上下文感知计算系统框架的关键问题主要在于:

1) 上下文建模和表示。对各类上下文独立进行建模几乎不存在问题,但问题是上下文感知计算需处理各式各样的上下文,对不同的上下文采用不同的建模方式显然不能满足互操作的要求。因此当前上下文建模和表示的主要问题是建立统一的上下文模型和表示。从现状来看,上下文统一建模可以分为两个层面,其一是形式上的统一,不同的上下文采用相同的表示方式,如键值对、标记语言等。这可用于解决上下文的高效存储、查询等问题,但不涉及上下文的语义;其二是支持语义的统一,如本体、UML(可部分支持语义)等,通过对语义的支持可以使上下文的有效利用达到一个更高的水平,但同时也增加了实现上的难度。

2) 上下文过滤、推断和融合。上下文的过滤、推断和融合构成了上下文演化的主体,其关键问题是如何甄别有效的上下文,如何利用上下文演绎出有用的结论以及如何联合演绎等。这些问题的产生与所感知的原始上下文所具有的统一、低层、不稳定、不精确等特点有关。当前,上下文过滤、推断和融合多采用 Bayesian 网络、模糊推理和人工神经网络等方法解决。

3) 上下文存储、查询和管理。上下文的有效存储和管理是上下文感知计算实用化的关键基础之一,但当前其研究尚未得到足够重视。上下文有效

存储的困难不仅来源于上下文的瞬时性、关联性等特点,还与上下文的建模和表示息息相关。

4) 互操作。互操作的含义相当广泛,既包括人与设备间的交互,也包括设备与设备之间、服务与服务之间的交互。人与设备的交互(人机交互, HCI)一直是该领域研究的重点。如设备的剧增(各类感知设备的大量部署)导致手工配置设备变得不切实际,这就有赖于新的自配置、自组织技术来解决人机交互问题<sup>[51]</sup>。此外日益丰富的人机交互方式则要求多通道用户界面技术来统一和协调。设备间互操作同样也是难点之一,这主要源自设备间的互操作的自发执行。服务间互操作则相对比较简单,可通过共享空间或消息传递等来解决<sup>[52]</sup>。

5) 智能执行。智能执行是上下文感知计算的主要体现。它是一个范围相当广泛的概念,包括如设备间自发的互操作,内容和表示的自适应等。因此智能执行不是一个单一的问题,与感知应用的目标关系密切,也有赖于丰富的技术手段来解决。

6) 应用程序接口。应用程序接口需提供对应用程序开发的完整支持,目前对应用程序开发多通过编程框架和函数接口等来实现。尽管应用程序接口并不影响框架的独立运行,但接口的好坏对应用开发的影响极大。好的接口可为应用的快速安全构建、测试等提供有效帮助,坏的接口则不利于应用开发甚至导致框架崩溃,因此,应用程序接口设计也是软件框架设计时应考虑的一个关键问题。

7) 鲁棒性。鲁棒性是指设备和用户免受临时故障或错误干扰的特性。与分布式计算相比,上下文感知计算环境中的资源受限、移动性大、环境易变、设施不可靠等特点使这个问题变得更为突出。分布式计算中基于资源冗余的容错策略也不再适合,因为感知计算下该问题的本质是资源失效下的容错<sup>[53]</sup>。

8) 安全与隐私。安全与隐私问题一直与信息技术的发展相伴相随。上下文感知计算给安全、信任和隐私提出了更大的挑战,这源于感知计算的开放性、自发的互操作、智能执行等特点。由于环境的动态性,传统的基于角色和访问控制等静态的安全解决方案已不再适用。为保证安全与隐私,Langheinrich 提出了若干指导性原则<sup>[54]</sup>。需要注意的是,安全与隐私不仅仅是技术问题,其解决有赖于政策和法律的帮助。

9) 标准化。随着上下文感知计算研究的兴起,许多的企业、科研机构等从各自不同的方面进行了

研究. 但协调机构和技术标准的缺乏致使上下文信息无法交互, 应用难于集成. 现在, 部分标准化组织已开始关注此领域, 如美国国家标准和技术协会下属的信息技术实验室( NIST/ITL )通过联合企业和研究机构制定了普适计算包括上下文感知计算领域的详细研究计划, 并负责协调、标准制定和测试等工作<sup>[55]</sup>. OSGi 则是一个潜在的可支持家庭环境下感知计算构建的工业标准之一. 然而, 标准化工作仍任重而道远.

### 3 结 语

通过有效地利用上下文信息来模拟人类在潜意识的交互, 上下感知计算将使应用具备智能感知、启动和执行的特点, 应用将更人性化, 也更方便人们的使用. 特别是在移动领域以及普适计算领域, 由于用户需求和环境的快速变化, 上下文信息的利用显得更具价值. 但上下文感知计算不是一蹴而就的, 其最终的实现依赖许多技术的发展, 如:

1) RFID. RFID 是弥补物理与虚拟世界鸿沟简单而有效的技术<sup>[34]</sup>. 目前, RFID 至少可以从两方面帮助构建感知计算: 第 1, 惟一标识参与感知计算的物理对象, 建立物理与虚拟世界间的关联; 第 2, 存储少量的历史上下文信息. 随着技术的发展, RFID 将可以与环境中的传感器以及数据源进行接口或直接感知环境参数<sup>[56-57]</sup>, 构建物理与虚拟世界更紧密的联系离不开 RFID 的帮助.

2) 本体论. 本体论用于指代物质存在在哲学上已经有很长的历史了, 在人工智能领域, 本体则是对领域概念的形式化描述, 它采用词汇表示领域知识并通过它描述该领域的特殊情形. 使用基于本体的方法建模上下文最大的好处在于可采用语言无关的方式描述上下文, 并可对领域知识进行形式化分析, 如在感知计算中可使用一阶谓词逻辑进行上下文推断.

3) 多通道用户界面技术. 从人机交互技术的角度看, 多通道用户界面技术是其关键之一, 也一直是研究关注的焦点. 多通道用户界面技术主要研究如何包容人机交互的多种方式, 以达到超越单一方式的协同效果, 其目标是使普通用户能非常自然地使用计算机.

4) 自适应、自配置、自组织技术. 感知计算中设备种类丰富、数量众多以及环境的限制是产生这类技术需求的主要原因. 自适应技术通过提供与环境

相适应的信息和行为可帮助感知计算达成部分智能目标, 而自配置、自组织技术则能借助对象间的分布与协作等帮助实现系统零配置和零管理的目标.

5) 认知模型. 感知计算需要像人类一样处理环境中隐含的上下文( 知识 ), 这就需要认真地研究人类认知的方法、过程并提出适合计算机采用的认知模型. 认知模型可以深层次地帮助对上下文进行感知、处理和合理利用. 但当前, 认知模型的研究成果非常有限, 整个研究仍处于起步阶段.

6) 人工智能技术. 通过将人工神经网络、Bayesian 网络、机器学习等人工智能技术用于信息过滤、推论、融合以及智能执行等领域, 可帮助感知计算实现智能化目标.

7) 系统建模和仿真技术. 上下文感知计算系统是一个大规模、复杂的、分布式、嵌入式混合系统, 其系统建构具有很强的复杂性和动态性, 给研究带来了很大的困难和很高的研究成本. 研究和建立满足感知计算系统建模和仿真、原型评价要求的工具软件, 允许对感知系统的各特定方面进行建模、仿真、评测和比较研究, 将极大地推动感知计算的发展.

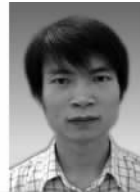
我们相信, 随着研究工作的不断深入和技术的不断发展, 上下文感知计算所涉及的各种问题最终必将得到解决, 并且将影响包括日常生活、工作和学习在内的人类生活的每一个角落, 计算机将以前所未有的密切程度来为人们提供更有效的帮助.

### 参 考 文 献

- [1] Mark Weiser. The computer for the 21st century[J]. Scientific American, 1991, 265(3): 66-75
- [2] Anind K Dey. Understanding and using context[J]. Personal and Ubiquitous Computing Journal, 2001, 5(1): 4-7
- [3] Bill Schilit, Norman Adams, Roy Want. Context-aware computing applications[C]. IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, 1994
- [4] Guanling Chen, David Kotz. A survey of context-aware mobile computing research[R]. Department of Computer Science, Dartmouth College, Tech Rep: TR2000-381, 2000
- [5] Quan Z Sheng, Boualem Benatallah. Contextuml: A UML-based modeling language for model driven development of context-aware Web services[C]. The Int'l Conf on Mobile Business, Sydney, Australia, 2005
- [6] Jan Van den Bergh, Karin Coninx. Towards modeling context-sensitive interactive applications: The context-sensitive user interface profile[C]. The 2005 ACM Symp on Software Visualization, St. Louis, Missouri, 2005
- [7] Michael Derntl, Karin A Hummel. Modeling context-aware e-learning scenarios[C]. The 3rd Int'l Conf on Pervasive Computing and Communications Workshops, Washington, DC, 2005

- [ 8 ] Shih-Chun Chou , Wen-Tai Hsieh , Fabien L Gandon , *et al.* Semantic Web technologies for context-aware museum tour guide applications [ C ]. The 19th Int'l Conf on Advanced Information Networking and Applications , Taipei , China , 2005
- [ 9 ] Fabien L Gandon , Norman M Sadeh. Semantic Web technologies to reconcile privacy and context awareness [ J ]. Journal of Web Semantics , 2004 , 1( 3 ) : 241–260
- [ 10 ] Panu Korpipaa , Jonna Hakikila , Juha Kela , *et al.* Utilising context ontology in mobile device application personalisation [ C ]. MUM 2004 , College Park , Maryland , USA , 2004
- [ 11 ] Tao Gu , Hung Keng Pung , Da Qing Zhang. A middleware for building context-aware mobile services [ C ]. IEEE 59th Vehicular Technology Conference , Milan , Italy , 2004
- [ 12 ] Sven Buchholz , Thomas Hamann , Gerald Hübsch. Comprehensive structured context profiles( csep ) : Design and experiences [ C ]. The 2nd IEEE Annual Conf on Pervasive Computing and Communications Workshops , Orlando , Florida , 2004
- [ 13 ] Yu Wang. An FSM model for situation-aware mobile application software systems [ C ]. ACMSE '04 , Huntsville , Alabama , USA , 2004
- [ 14 ] Lu Yan , Kaisa Sere. A formalism for context-aware mobile computing [ C ]. The ISP DC/HeteroPar '04 , Ireland , 2004
- [ 15 ] Oh Byung Kwon. Modeling and generating context-aware agent-based applications with amended colored Petri nets [ J ]. Expert Systems with Applications , 2004 , 27( 4 ) : 609–621
- [ 16 ] Paul Castro , Richard Muntz. Managing context data for smart spaces [ J ]. IEEE Personal Communications , 2000 , 7( 5 ) : 44–46
- [ 17 ] Albrecht Schmidt , Kofi Asante Aidoo , Antti Takaluoma , *et al.* Advanced interaction in context [ C ]. The 1st Int'l Symp on Handheld and Ubiquitous Computing , Karlsruhe , Germany , 1999
- [ 18 ] Amir Padovitz , Arkady Zaslavsky , Seng Wai Loke , *et al.* Maintaining continuous dependability in sensor-based context-aware pervasive computing systems [ C ]. The 38th Hawaii Int'l Conf on System Sciences , Hawaii , USA , 2005
- [ 19 ] Kaori Fujinami , Tetsuo Yamabe , Tatsuo Nakajima. Take me with you ! : A case study of context-aware application integrating cyber and physical spaces [ C ]. 2004 ACM Symp on Applied Computing , Nicosia , Cyprus , 2004
- [ 20 ] Jason I Hong. The context fabric : An infrastructure for context-aware computing [ C ]. Conference on Human Factors in Computing Systems , Minneapolis , Minnesota , USA , 2002
- [ 21 ] Karen Henriksen , Jadwiga Indulska. Modelling and using imperfect context information [ C ]. The 2nd IEEE Annual Conf on Pervasive Computing and Communications Workshops , Orlando , Florida , 2004
- [ 22 ] Manuele Kirsch-Pinheiro , Marl' ene Villanova-Oliver , J' erome Gensel , *et al.* Context-aware filtering for collaborative Web system : Adapting the awareness information to the user's context [ C ]. 2005 ACM Symp on Applied Computing , Santa Fe , New Mexico , 2005
- [ 23 ] Mohamed Khedr , Ahmed Karmouch. Negotiating context information in context-aware systems [ J ]. IEEE Intelligent Systems , 2004 , 19( 6 ) : 21–29
- [ 24 ] Anind K Dey , Jennifer Mankoff. Designing mediation for context-aware applications [ J ]. ACM Trans on Computer-Human Interaction , 2005 , 12( 1 ) : 58–80
- [ 25 ] Manuel Roman , Christopher Hess , Renato Cerqueira , *et al.* Gaia : A middleware infrastructure to enable active spaces [ J ]. IEEE Pervasive Computing , 2002 , 1( 4 ) : 74–83
- [ 26 ] Renato Cerqueira , Christopher Hess , Manuel Roman , *et al.* Gaia : A development infrastructure for active spaces [ C ]. Workshop on Application Models and Programming Tools for ubiquitous computing , Atlanta , Georgia , 2001
- [ 27 ] Sharat Khungar , Jukka Riekk. A context based storage system for mobile computing applications [ J ]. Mobile Computing and Communications Review , 2005 , 9( 1 ) : 64–68
- [ 28 ] Lee Hoi Leong , Shinsuke Kobayashi , Noboru Koshizuka , *et al.* Casis : A context-aware speech interface system [ C ]. IUI '05 , San Diego , California , 2005
- [ 29 ] Alois Ferscha , Clemens Holzmann , Stefan Oppl. Context awareness for group interaction support [ C ]. MobiWac '04 , Philadelphia , PA , 2004
- [ 30 ] Carey Williamson , Qian Wu. A case for context-aware TCP/IP [ J ]. Performance Evaluation Review , 2002 , 29( 4 ) : 11–23
- [ 31 ] Carey Williamson , Qian Wu. Context-aware TCP/IP [ C ]. ACM SIGMETRICS '02 , Marina Del Rey , California , 2002
- [ 32 ] A K Dey , G D Abowd , D Salber. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications [ J ]. Human-Computer Interaction , 2001 , 16( 2–4 ) : 97–166
- [ 33 ] W K Edwards. Coordination infrastructure in collaborative systems [ Ph D dissertation ] [ D ]. Atlanta , USA : College of Computing , Georgia Institution of Technology , 1995
- [ 34 ] Kay Römer , Friedemann Mattern , *et al.* Infrastructure for virtual counterparts of real world objects [ R ]. Department of Computer Science , ETH Zurich , Tech Rep : IFVCORWO , 2001
- [ 35 ] Stephen Volda , Elizabet D Mynatt , Blair MacIntyre , *et al.* Integrating virtual and physical context to support knowledge workers [ J ]. IEEE Pervasive Computing , 2002 , 1( 3 ) : 73–79
- [ 36 ] Jason Pascoe. Adding generic contextual capabilities to wearable computers [ C ]. The 2nd Int'l Symp on Wearable Computers , Pittsburgh , 1998
- [ 37 ] M Khedr , A Karmouch. Acai : Agent-based context-aware infrastructure for spontaneous applications [ J ]. Network and Computer Applications , 2005 , 28( 1 ) : 19–44
- [ 38 ] Jens Wohltorf , Richard Cisse , Andreas Rieger , *et al.* Berlintertainment—An agent-based serviceware framework for context-aware services [ J ]. IEEE Communications Magazine , 2005 , 43( 6 ) : 102–109
- [ 39 ] Paolo Bellavista , Antonio Corradi , Rebecca Montanari , *et al.* Context-aware middleware for resource management in the wireless Internet [ J ]. IEEE Trans on Software Engineering , 2003 , 29( 12 ) : 1086–1099
- [ 40 ] Anind K Dey , Raffay Hamid , Chris Bechmann , *et al.* A cappella : Programming by demonstration of context-aware applications [ C ]. CHI 2004 , Vienna , 2004

- [ 41 ] Panu Korpipaa , Esko-Juhani Malm , Ilkka Salminen , *et al.* . Context management for end user development of context-aware applications[ C ]. MDM 2005 , Ayia Napa , Cyprus , 2005
- [ 42 ] William G Griswold , Robert Boyer , Steven W Brown , *et al.* . A component architecture for an extensible , highly integrated context-aware computing infrastructure [ C ]. The 25th Int 'l Conf on Software Engineering , Portland , Oregon , 2003
- [ 43 ] William G Griswold , Patricia Shanahan , Steven W Brown , *et al.* . Activecampus : Experiments in community-oriented ubiquitous computing[ J ]. IEEE Computer Society , 2004 , 37( 10 ) : 73-81
- [ 44 ] Licia Capra , Wolfgang Emmerich , Cecilia Mascolo . Carisma : Context-aware reflective middleware system for mobile applications[ J ]. IEEE Trans on Software Engineering , 2003 , 29( 10 ) : 929-944
- [ 45 ] Oleg Davidyuk , Jukka Riekki , Ville-Mikko Rautio , *et al.* . Context-aware middleware for mobile multimedia application [ C ]. MUM 2004 , College Park , Maryland , 2004
- [ 46 ] Tao Gu , Hung Keng Pung , Da Qing Zhang . A service-oriented middleware for building contextaware services[ J ]. Network and Computer Applications , 2005 , 28( 1 ) : 1-18
- [ 47 ] The OSGi Alliance . OSGi service platform core specification [ OL ]. <http://www.osgi.org> , 2005
- [ 48 ] Nam-Shik Park , Kang-Woo Lee , Hyun Kim . A middleware for supporting context-aware services in mobile and ubiquitous environment[ C ]. ICMB '05 , Sydney , Australia , 2005
- [ 49 ] Timothy Y Sohn . Icap : An informal tool for interactive prototyping of context-aware applications[ C ]. CHI 2003 , Fort Lauderdale , Floria , 2003
- [ 50 ] Yang Li , Jason I Hong , James A Landay . Topiary : A tool for prototyping location-enhanced applications [ C ]. UIST ' 04 , Santa Fe , NM , USA , 2004
- [ 51 ] Deborah Estrin , David Culler , Kris Pister , *et al.* . Connecting the physical world with pervasive networks[ J ]. IEEE Pervasive Computing , 2002 , 1( 1 ) : 59-69
- [ 52 ] Stephen S Yau , Fariaz Karim , Yu Wang , *et al.* . Reconfigurable context-sensitive middleware for pervasive computing[ J ]. IEEE Pervasive Computing , 2002 , 1( 3 ) : 33-40
- [ 53 ] Tim Kindberg , Armando Fox . System software for ubiquitous computing[ J ]. IEEE Pervasive Computing , 2002 , 1( 1 ) : 70-81
- [ 54 ] Marc Langheinrich . Privacy by design—Principles of privacy-aware ubiquitous systems [ R ]. Institute of Information Systems , IFW , Swiss Federal Institute of Technology , ETH Zurich , Tech Rep : PBD , 2000
- [ 55 ] Lynne Rosenthal , Vincent Stanford . Nist smart space : Pervasive computing initiative[ C ]. IEEE 9th Int 'l Workshops on Enabling Technologies : Infrastructure for Collaborative Enterprises , Gathersburg , MD , USA , 2000
- [ 56 ] Vince Stanford . Pervasive computing goes the last hundred feet with RFID systems[ J ]. IEEE Pervasive Computing , 2003 , 2( 2 ) : 9-14
- [ 57 ] Roy Want . Enabling ubiquitous sensing with RFID[ J ]. IEEE Invisible Computing , 2004 , 37( 4 ) : 84-86



**Li Rui** , born in 1980 . Ph. D. candidate in the School of Computer and Communication , Hunan University . His current research interests include context-aware computing and wireless network .

李蕊 , 1980 年生 , 博士研究生 , 主要研究方向为上下文感知计算、无线网络。



**Li Renfa** , born in 1957 . Professor and Ph. D. supervisor in the School of Computer and Communication , Hunan University . His main research interests include embedded system , virtual laboratory , wireless network , etc .

李仁发 , 1957 年生 , 教授 , 博士生导师 , 主要研究方向为嵌入式计算、虚拟技术、无线网络、网络数字媒体技术。

## Research Background

This study is supported by the National Natural Science Foundation of China under grant No. 60673061 .

Context-aware computing is a mobile computing paradigm in which applications can discover and take advantage of contextual information , such as the user location , time , nearby people and devices , and the user 's activity . Research on context-aware computing began about a decade ago . Since then , many researchers have studied this topic and built several context-aware applications to demonstrate the usefulness of the technology . But in general context-aware computing system is a distributed , large scale , sophisticated and hybrid system . It is very difficult to design and implement such system because of its complexity .