

面向普适计算的应用共享模型*

朱珍民¹⁺, 蒋发群^{1,2}, 苏晓丽^{1,2}, 李锦涛¹, 叶剑^{1,2}

¹(中国科学院 计算技术研究所,北京 100080)

²(中国科学院 研究生院,北京 100049)

Application Sharing Model for Pervasive Computing

ZHU Zhen-Min¹⁺, JIANG Fa-Qun^{1,2}, SU Xiao-Li^{1,2}, LI Jin-Tao¹, YE Jian^{1,2}

¹(Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100080, China)

²(Graduate University, The Chinese Academy of Sciences, Beijing 100049, China)

+ Corresponding author: Phn: +86-10-62600640, Fax: +86-10-62600650, E-mail: fqjiang@ict.ac.cn

Zhu ZM, Jiang FQ, Su XL, Li JT, Ye J. Application sharing model for pervasive computing. *Journal of Software*, 2007,18(Suppl.):54-62. <http://www.jos.org.cn/1000-9825/18/s54.htm>

Abstract: In pervasive computing environment, applications available for end-users are abundant, distributed and heterogeneous. In order to share these applications, this paper presents a user-centric applications sharing model which called U-ASM. The basic concepts and structure of U-ASM are defined, which abstract the applications from service providers as well as end-users. The paper uses ontology to organize distributed applications as a logic unit with semantic relationships and utilizes virtualization technology to encapsulate applications. Finally the paper presents a prototype of the U-ASM which validates feasibility and validity of the model. The research result has been applied in R&D infrastructure and facility development of ministry of science and technology and has great flexibility and extensibility.

Key words: pervasive computing; applications sharing; virtualization; quality of service

摘要: 为了有效共享普适计算环境中分布、异构的应用资源,提出了一种以用户为中心的应用共享模型 U-ASM,定义了 U-ASM 的基本概念和结构,从用户、服务提供者等不同层面对应用资源进行抽象;利用应用资源的语义关系,对应用资源进行语义建模,通过虚拟化技术构建应用资源的逻辑视图;通过应用资源共享原型,验证了模型的可行性和有效性.该成果已应用于国家科技基础条件平台建设计划,灵活性和扩展性良好.

关键词: 普适计算;应用共享;虚拟化;服务质量

随着计算机和互联网技术的广泛应用,不仅导致应用资源的种类和数量急剧增长,而且其异构性、分布性也日益增强.当前计算环境中的应用资源不仅包含 Web 服务,而且还有大量的遗留应用和系统,使得管理非常复杂,难以共享,利用率低.另外,用户的应用需求也日益多样化、个性化.因此,在异构计算环境中如何实现跨平台

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z112 (国家高技术研究发展计划(863)); the R&D Infrastructure and Facility Development of the Ministry of Science and Technology of China under Grant No.2005DKA63900 (国家科技基础条件平台专项基金)

Received 2007-04-15; Accepted 2007-11-25

的应用共享,使得用户能够随时随地、透明地访问大量存在的、分布、异构的应用资源,就自然成为当前计算环境亟待解决的一个重要问题^[1,2]。特别是随着“以人为本”的普适计算技术^[3]的发展,这种需求变得更加突出。

针对这个问题,研究人员和业界已经提出了许多相关的理念和解决方案。虚拟机技术^[4]是实现跨平台应用共享的常用解决方案,它通过虚拟机屏蔽硬件和操作系统的异构性,为终端用户使用异构平台的应用资源提供了方便。瘦客户计算(thin-client computing)技术能为终端用户提供远程桌面应用共享^[5]。面向服务的架构(SOA)理论为实现应用集成和按需服务提供了一种思路,使得服务能够以一种统一和通用的方式交互,实现服务间的松耦合^[6]。

本文提出了一种以用户为中心的应用共享模型(user-centric applications sharing model),简称 U-ASM,旨在为终端用户建立一个合理的、统一的应用共享的基础架构,以实现应用和应用系统的统一管理和按需服务,从而满足终端用户的个性化应用需求。

1 相关工作

随着计算机技术和网络技术的发展,普适计算环境中的资源基础架构日益庞大,资源的异构性和分布性日益增强。对于这些广域分布资源,我们可以粗略地将其分为两类,一类是物理资源,如 CPU、内存等;另一类是应用资源,如应用软件、Web 服务等。为了实现普适计算环境中各种资源的有效共享,研究人员和业界已经提出了许多理论方法及其相应的解决方案。按需计算、效用计算给出了资源共享的远景目标;网格计算也采用以“服务”为中心的概念,为终端用户提供资源共享和资源虚拟化环境。但是,这些技术目前主要是针对物理资源进行研究的。由于应用资源相对物理资源而言更为复杂,这也是本文所关注的重点。

虚拟机技术通过虚拟机屏蔽硬件和操作系统的异构性,为终端用户使用异构平台的应用资源提供了便利。但是虚拟机对硬件和网络的要求较高,同时又是一种静态解决方案,不能提供异构应用的动态融合。

瘦客户计算技术通过基于终端应用协议的远程显示技术,为终端用户提供远程桌面应用共享。其中有代表性的软件产品和协议包括 X-window 系统、Citrix 的 ICA 协议、Microsoft 的 RDP 协议、AT&T 剑桥实验室提出的 VNC 等^[5-7]。但是,瘦客户计算模式也有一定的局限:首先,该模式的共享粒度较大,提供的是整个桌面级别的应用系统共享,不支持应用程序级别的共享,从而安全性、可靠性受到影响;其次,对共享的应用资源之间缺乏相应的语义支持,使得应用资源彼此孤立,无法进行统一管理,因此无法实现多个服务节点间负载均衡和按需服务;另外,无法按需改变和定制个性化的用户环境。

SOA 理论为实现应用集成和按需服务提供了一种思路,Web 服务^[8]作为 SOA 的一个重要组成部分,它采用 WSDL、UDDI 和 SOAP 等标准和协议解决了分布式计算和代码重用等问题,倍受工业界和学术界关注。尽管 Web 服务具有高度的互操作性、跨平台性和松耦合性等特点,但它有其固有的局限和不足:首先,Web 服务更适合于静态信息的传输与共享,对复杂的 GUI 界面支持有限,交互能力欠佳;其次,如果要在 Web 上支持目前用户已经熟悉的桌面应用集合(如 Office、AutoCAD 等),就需要重新开发相应的应用组件,开发难度和成本增加,这对软件开发方和软件购买部署方都是一个严重的浪费。因此,Web 服务不能完全代替现有的 GUI 应用^[9]。

针对上述问题,本文提出了一种以用户为中心的应用共享模型 U-ASM,其核心是分别从用户、服务提供者等不同层面对应用资源进行抽象,并通过虚拟化机制将应用资源服务化,为终端用户提供个性化的服务视图,实现跨平台的异构应用资源共享和按需使用。

2 U-ASM 设计

2.1 U-ASM 架构

为了实现应用资源共享和综合利用,本文从用户、服务中介、提供者 3 个不同层面对普适计算环境进行抽象,提出了一种以用户为中心的应用共享模型——U-ASM。

记所讨论的普适计算环境为 W , 假设 W 包含 n 个服务提供节点 N_1, N_2, \dots, N_n , 其中每个节点存在若干个可用的应用资源.

定义 1. 应用资源 A_R 可以表示一个四元组: $A_R = \langle ID, A_T, A_D, A_I \rangle$. 其中 ID 为应用资源标识, 唯一确定当前计算环境中的某一应用资源; A_T 为应用类型, 同类型的应用具有相同操作接口、描述语义和行为; A_D 为应用描述, 由一组属性对象组成, 其语义由应用类型决定; A_I 是一个应用接口集合, 例如操作接口、呈现接口等.

定义 2. 一个应用资源的服务质量 (quality of service, 简称 QoS) 由 k 个应用属性来刻画, 记为 A_Q , 即 $A_Q = \{q_1, q_2, \dots, q_k\}$. 通过这个向量描述 QoS 属性, 从而支持基于 QoS 约束的应用资源选择.

定义 3. 令 A_R 是应用资源的集合, 那么按照功能属性把这些应用资源分成若干个非空子集, 使得 A_R 中的每一个元素都至少属于一个非空子集, 而同一个类中的应用资源的差异在于其 QoS 属性, 我们把这样一个应用资源集合称为应用资源类, 记为 A_C . 注意, 这些应用资源类的全体构成的集合只是 A_R 的一个覆盖, 并不一定是划分.

定义 4. 我们将任意节点 N_i 中可用的应用资源称为有效应用, 记为 e_a .

定义 5. 有效空间 E_S 定义为分布在服务提供节点上所有有效应用 e_a 的集合, 记为 $E_S = \{e_a_1, e_a_2, \dots, e_a_m\}$, 它描述了所有可以利用的应用资源.

由于不同服务提供节点间的有效应用的语义表示和组织的异构性, 为了方便管理者进行统一管理, 因此, 需要对有效应用进行抽象. 本文利用虚拟化机制对有效应用进行抽象, 屏蔽它们在语义描述上的差异.

定义 6. 我们把对有效应用进行虚拟化的结果称为虚拟应用, 记为 v_a , 它是对有效应用的抽象.

定义 7. 虚拟应用类 V_C 是对虚拟应用聚类的结果, 虚拟应用相当于虚拟应用类的实例. 这样, $\forall v_a$, 则 $\exists V_C$, 使得 $v_a \in V_C$.

考虑到应用资源的分布特性以及用户的区域性, 为了便于管理者对应用资源进行管理和控制, 通常将当前计算环境分成多个小单元, 称为服务域. 假设当前环境 W 分成 t 个服务域.

定义 8. 服务域 S_A 可以用一个二元组表示, $S_A = \langle V_A, V_R \rangle$, 其中 V_A 是本服务域中所有虚拟应用的集合, 记为 $V_A = \{v_a_1, v_a_2, \dots, v_a_m\}$; V_R 是虚拟应用语义关系的集合.

定义 9. 抽象空间 A_S 是所有服务域的集合, 记为 $A_S = \bigcup_{i=1}^t S_A_i$.

随着以用户为中心的服务理念的进一步发展, 需要为用户提供个性化的计算环境, 因此需要从用户角度抽象应用资源. 对用户而言, 用户关注的是可以操作的应用接口, 如 GUI 接口等.

假设当前环境 W 包含 L 个用户 U_1, U_2, \dots, U_L , 每个用户的个人计算环境中包含若干个需要使用的应用接口, 其集合构成用户个性化的计算环境——个人空间, 个人空间的形成是对现有应用资源进行重构和定制的结果, 这种重构是动态的.

定义 10. 个人空间是任意用户 U_i 当前需要使用的应用接口集合, 记为 P_S_i .

定义 11. 用户空间是所有个人空间的集合, 记为 $U_S = \bigcup_{i=1}^L P_S_i$.

定义 12. U-ASM 模型可以用一个五元组表示, $U\text{-ASM} = \langle E_S, A_S, U_S, F, G \rangle$, 其中, E_S, A_S, U_S 分别是有效空间、抽象空间、用户空间; F 是用户空间到虚拟空间的映射, G 是虚拟空间到有效空间的映射.

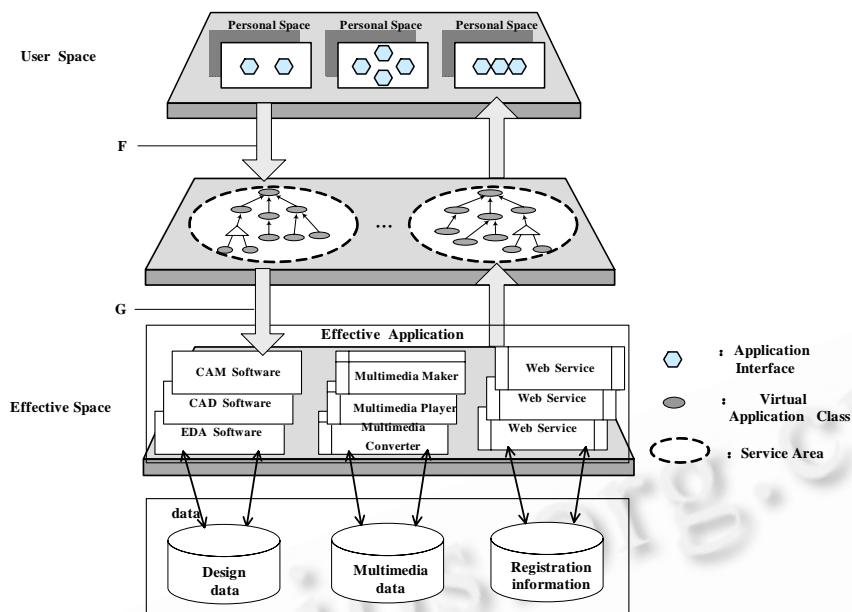


Fig.1 A framework of the U-ASM

图1 U-ASM 结构图

U-ASM 是一种分布式计算模型,其设计目标是:

- (1) 个性化服务:通过应用资源的动态重构技术为用户提供个性化的计算环境.
- (2) 透明访问:向用户提供透明的访问服务是网络计算、普适计算的一个重要特征,也是本模型追求的目标之一.用户只需要提交相应的应用请求,便可以访问所需的应用资源,而不必关心这些资源所在的位置等细节.
- (3) 应用资源的按需使用:根据用户需求和当前上下文为用户动态地选择所需的应用资源.
- (4) 通用性:支持多种异构平台(Windows, Linux 等);支持多种类型应用资源(传统的桌面应用、遗留系统和基于 Web 的应用等).

从图 1 可知,U-ASM 中的用户空间 U_S 是面向最终用户的,它将采用更高级、更容易为人理解的方式来表示和描述应用资源,同时将根据用户的个性化需求重构当前所需的计算环境——个人空间,这种重构是动态的.抽象空间 A_S 是整个模型的核心,一方面它将利用虚拟化机制屏蔽不同提供者所提供的应用资源在语义描述上的差异,为上层提供标准化的语义支持;另一方面,它将负责根据用户需求实现应用资源的选择以及智能协商决策过程,支持应用接口到应用实例的动态绑定,为实现应用资源的按需使用、个性化服务提供基础.有效空间 E_S 将负责完成应用资源的注册和监控.因此,U-ASM 体现了以用户为中心的服务理念.

2.2 应用资源组织和管理

应用资源组织和管理的主要目的是屏蔽应用资源所在物理平台的异构性和地理位置分布性,为透明地访问应用资源提供统一的、标准的接口,为实现应用资源共享提供基础.本文将充分利用应用资源的语义关系,对应用资源进行语义建模,并且通过虚拟化技术构建应用资源的逻辑视图,为应用资源自动发现提供基础.

2.2.1 应用资源的语义建模

随着普适计算环境中的应用资源日益丰富,传统的基于语法的描述方式难以实现应用资源的自动发现,同时容易造成大量的用户得不到所需的应用资源和大量的应用资源空闲的矛盾,难以满足用户的个性化需求.为了实现应用资源的自动发现和智能交互,本文将对普适计算环境中的应用资源进行语义建模.为了有效地描述应用资源的语义,本文引入了本体描述应用资源的语义关系.

本体作为一种有效表现概念层次结构和语义的模型,为应用资源和应用资源间关系的描述提供了强有力

手段^[10].它通过概念之间的关系来描述概念的语义,其中基本的概念关系^[11]见表 1.

Table 1 Basic conception relation of ontology

表 1 本体的基本概念关系

Name	Expression	Meaning
Kind relation	Kind-of	The relation between subclass and class in concepts
Whole-Part relation	Part-of	The relation between part and the whole in concepts
Instance relation	Instance-of	Relation between class target and class
Attribute relation	Attribute-of	One concept a attribute of another concept

本文采用 OWL-S 描述应用资源之间的关系,主要从以下 4 个方面表示应用资源及其关系:

- (1) 应用资源所属的概念类及其类层次结构.
- (2) 类的个体,表示具体的应用资源实例.
- (3) 属性及其值.
- (4) 类和类、类和个体、属性间的关系.

应用资源本体部分实例如图2所示.

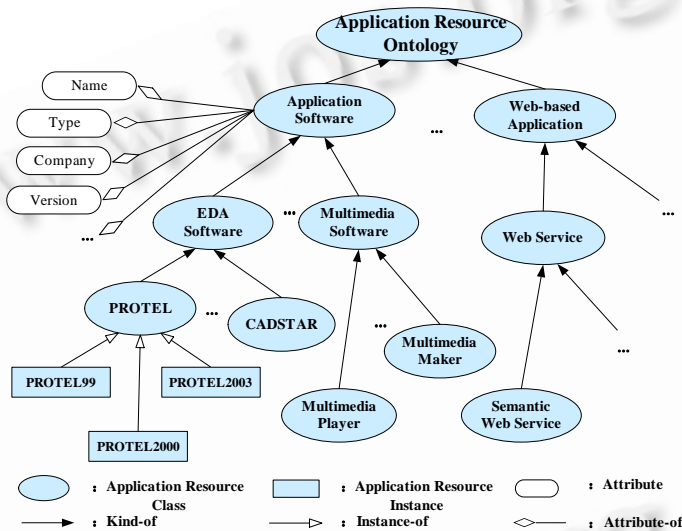


Fig.2 Partial instance of application resource ontology

图 2 应用资源本体实例

应用资源本体所描述的是应用资源相关的概念及其相互关系、属性及与概念的关系,因此应用资源本体所表达的是一种应用资源的语义信息.应用资源的语义描述将会使异构、分布的应用资源具有机器以及人可理解的语义信息,是对应用资源进行有效组织和管理的关键,为自动化、智能化的操作和使用这些资源提供了便利.

2.2.2 基于语义的虚拟化

虚拟化技术能够有效地屏蔽实现和操作层面的复杂性,是实现应用资源统一管理和按需服务的基础,正逐渐获得研究者的广泛关注.现有的虚拟化技术可以分为两类:一类针对一组具有协作关系的资源进行虚拟化;另一类针对一组功能相同的资源进行虚拟化^[12].为了方便用户理解和使用应用资源,本文所用的虚拟化技术类似于第 2 类,它将采用抽象、转化等手段,为相同类型的应用资源提供一致的服务访问方式.

本文中虚拟化过程将利用应用资源间的语义关系,具体实现过程如下:

- (1) 有效应用的封装和抽象:首先对不同服务提供者提供的语义信息进行标准化和规范化,然后根据应用资源类型对有效应用的应用接口进行抽象和封装,屏蔽有效应用底层差异性,最终形成具有标准、统一的接口的虚拟应用.例如,如果是 Web 服务类资源,则利用标准的协议(如 SOAP,WSDL 等)将其封装成一个服务标准的

Web 服务;如果是基于 GUI 的桌面应用资源,则需要封装其显示协议,抽象其呈现接口.这样就可以为有效应用构建统一的逻辑资源视图.

(2) 虚拟应用的聚类:对相同功能的虚拟应用进行聚类,形成虚拟应用类,其中一个虚拟应用类中有若干个抽象对象——虚拟应用.

(3) 虚拟应用类的组织:利用应用资源的语义关系,计算虚拟应用类的语义相关度,利用本体组织虚拟应用类,形成相应的语义网络.本文通过虚拟应用类的属性相关与否来计算它们之间的语义相关度^[13].对于任意两个虚拟应用类 V_C_i 和 V_C_j ,那么它们的语义相关度可按照公式(1)进行计算.

$$Sim(V_C_i, V_C_j) = \frac{\alpha \times |V_C_i.Attr \cap V_C_j.Attr|}{\alpha \times |V_C_i.Attr \cap V_C_j.Attr| + \beta \times |V_C_i.Attr \setminus V_C_j.Attr| + \gamma \times |V_C_j.Attr \setminus V_C_i.Attr|} \quad (1)$$

其中 $\alpha, \beta, \gamma \geq 0$; $V_C_i.Attr, V_C_j.Attr$ 分别表示虚拟应用类的属性集合, $\cap, \setminus, /$ 分别是集合的并、模、差运算,参数 α, β, γ 是为应用类 V_C_i 和 V_C_j 之间相同属性和不同属性而赋予的不同权重,表示相同属性和不同属性在区别不同虚拟应用类的过程中所起的作用不一样.

2.3 基于 QoS 的应用选择机制

随着普适计算环境中共享的应用资源数量和种类的增加,用户选择空间不断增大,针对同一种用户需求可供选择的应用资源会越来越多.传统的静态绑定、手工选择或只考虑功能的选择机制已不能满足用户的需求.因此,用户希望应用选择机制能够从 QoS 出发进行比较和选择,动态地选择最适合用户需求的应用资源.因此, QoS 在应用资源选择过程中占据重要的地位.目前基于 QoS 约束的选择机制的研究工作越来越多,本文采用基于 QoS 属性计算的选择机制^[14,15],对应用资源的 QoS 进行动态评估,从而满足用户的个性化需求.

2.3.1 QoS 属性计算

一个应用资源的 QoS 描述了应用资源满足用户需求的能力,我们可以从很多方面描述 QoS,如价格、响应时间、执行时间、可靠性、补偿率、赔偿率等.

根据定义 2,一个应用资源的 QoS 由 k 个属性来刻画,即 $\{q_1, q_2, \dots, q_k\}$,因此,用户对一个有效应用 e_a 的 QoS 期望 QoS_E 可以按照式(2)进行计算.

$$QoS_E(e_a) = \sum_{i=1}^k (\omega_i \times q_i) \quad (2)$$

其中, ω_i 为相应 QoS 属性的权值,表示用户对 QoS 属性的关注程度,并且 $0 \leq \omega_i \leq 1, \sum_{i=1}^k \omega_i = 1, q_i$ 为第 i 项 QoS 属性值, k 为 QoS 属性个数.

由于不同的 QoS 属性取值范围差别较大,并且不同的应用资源的 QoS 属性值也可能相差很大,因此直接将 QoS 属性值代入式(2)进行计算会影响函数的公平性.而且,在这些 QoS 属性中又可以分为两类,一类表示属性值取值越大越有利于服务请求者,如可靠性、补偿率等;另一类表示属性值取值越小越有利于服务请求者,如服务价格、响应时间、赔偿率等.因此,在进行应用选择前需要先对 QoS 属性按照式(3),式(4)进行规范化处理^[16].

$$q'_j = \begin{cases} \frac{q_j^{\max} - q_{ij}}{q_j^{\max} - q_j^{\min}}, & \text{if } q_j^{\max} - q_j^{\min} \neq 0 \\ 1, & \text{if } q_j^{\max} - q_j^{\min} = 0 \end{cases} \quad (3)$$

$$q'_j = \begin{cases} \frac{q_{ij} - q_j^{\min}}{q_j^{\max} - q_j^{\min}}, & \text{if } q_j^{\max} - q_j^{\min} \neq 0 \\ 1, & \text{if } q_j^{\max} - q_j^{\min} = 0 \end{cases} \quad (4)$$

2.3.2 应用资源选择算法

根据定义 2 给出的 QoS 描述向量和第 2.2.1 节中 QoS 属性计算方法,对满足功能需求及访问控制要求的候选服务进行过滤和选择,从而获得满足用户 QoS 约束的应用资源.

算法 1. 应用资源选择算法.

假设用户的应用资源请求为 U_R , 用户期望的应用资源集合为 U_E .

Step 1. 根据 U_R 取得用户上下文.

Step 2. 根据用户上下文获得推荐应用资源空间 R_S .

Step 3. 对于 $\forall e_a \in R_S$, 判断 e_a 中的当前状态, 如果可用, 则转 Step 4; 否则依次取下一个 e_a .

Step 4. 计算 $QoS_E(e_a)$.

Step 5. 如果 $QoS_E(e_a)$ 满足 U_R , 则将 e_a 加入 U_E ; 转 Step 3.

Step 6. 如果 $U_E = Null$, 则转 Step 7; 否则, 转 Step 8.

Step 7. 返回希望用户放宽 QoS 限制提示信息.

Step 8. 对 U_E 进行排序.

Step 9. 返回 U_E .

本文提出的应用资源选择算法并不是从所有的应用资源中进行选择, 而是从推荐应用资源空间 R_S 进行选择. 推荐应用资源空间 R_S 是用户登录时系统根据用户的 profile 和历史信息为用户生成应用资源索引, 它将随着用户当前上下文的变化而动态变化. 这样可以有效地减少应用资源选择的范围, 提高了选择的效率和精确度, 更好地满足用户的个性化需求.

3 原型实现

基于 U-ASM 模型, 我们已经实现了一个异构应用资源共享的原型系统 (heterogeneous applications sharing system, 简称 HASS). 为了继承用户已熟悉的基于 GUI 的交互方式, 该原型系统选择了虚拟桌面^[17]作为用户视图的呈现方式, 通过虚拟桌面对应用接口进行无缝集成, 用户能够以简洁和一致的方式来透明地使用网络上动态的、分布式的应用资源. HASS 的实现效果如图 3 所示.

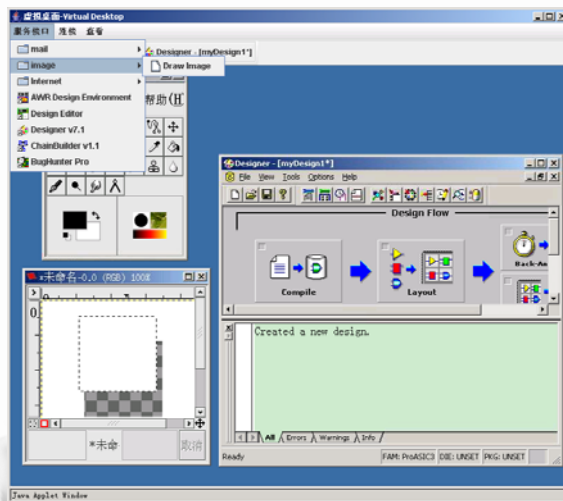


Fig.3 Application example of prototype system based on U-ASM
图 3 基于 U-ASM 的原型系统应用实例

基于 U-ASM 模型的 HASS 系统已经成为“国家科技基础条件平台应用服务支撑系统”的一个重要组成部分, 重点实现了面向服务的应用程序共享^[18]. 科技基础条件平台存在大量的异构交互式应用资源, 平台需要集成已有应用资源, 实现对分布式异构应用资源的统一管理和有效利用. 图 3 显示了 HASS 系统能够共享 Linux 环境中的应用 (如 Draw 绘图软件), 也能够共享 Windows 环境中的应用 (如 EDA 工具软件), 实现了跨系统平台的应用共享. 本文提出的 U-ASM 模型为此类问题提供了一个解决方案并得到了实际应用. U-ASM 模型能够有效

地屏蔽应用资源的异构性和分布性,实现应用资源的按需服务和有效共享,最终满足用户的个性化需求.

4 结束语

随着普适计算环境中应用资源数量和种类的增长,为了实现应用资源的有效共享和按需服务,本文论述了一种以用户为中心的应用共享模型 U-ASM,目的是为终端用户建立一个合理的、统一的应用资源共享的基础架构,以实现应用资源的统一管理和按需服务.该模型的特点在于通过从服务提供者角度、管理者角度和用户角度对应用资源进行抽象来屏蔽应用资源的分布性和异构性,同时通过虚拟化机制屏蔽应用资源在语义描述上的差异,为终端用户建立一个合理的、统一的应用共享的基础架构,实现跨平台的异构应用资源的有效共享和按需使用.

本文提出的 U-ASM 模型在“国家科技基础条件平台应用服务支撑系统”上得到了实际应用.该模型能够适用多种网络环境,具有很强的通用性和扩展性.由于用户需求具有的动态性、个性化的特点,为了克服现有的应用资源无法快速应对用户需求的缺陷,以用户为中心由最终用户构造应用的模式正成为研究热点,下一步的研究方向是在 U-ASM 模型基础上研究应用资源的组合机制,为用户提供一种简单、易用的方式自主地组合应用资源,构造业务级应用,从而满足用户的个性化任务需求.

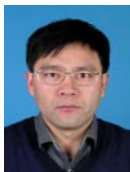
References:

- [1] Ouzzani M, Bouguettaya A. Efficient access to Web services. *Internet Computing*, 2004,8(2):34-44.
- [2] Kindberg T, Fox A. System software for ubiquitous computing. *IEEE Pervasive Computing*, 2002,1(1):70-81.
- [3] Satyanarayanan M. Pervasive computing: Vision and challenges. *IEEE Personal Communications*, 2001,6(8):10-17.
- [4] Whitaker A, Shaw M, Gribble S. Denali: Lightweight virtual machines for distributed and networked applications. In: *Proc. of the USENIX Technical Conf. Monterey*, 2002.
- [5] Mike S. Technologies for thin client architectures [Ph.D. Thesis]. University of Zurich, 2001.
- [6] Papazoglou MP. Service-Oriented computing: Concepts, characteristics and directions. In: *Proc. of the 4th Int'l Conf. on Web Information Systems Engineering*. 2003. 3-13.
- [7] Yang SJ, Nieh J, Selsky M, Tiwari N. The performance of remote display mechanisms for thin-client computing. In: *Proc. of the 2002 USENIX Annual Technical Conf. Monterey*, 2002. 131-146.
- [8] Yue K, Wang XL, Zhou AY. Underlying techniques for Web services: A survey. *Journal of Software*, 2004,15(3):428-442 (in Chinese with English abstract).
- [9] Liu CS, Li J, Yu HY. GAS: A GUI application sharing system in grid environment. *Computer Engineering*, 2007,33(14):98-100 (in Chinese with English abstract).
- [10] Li SP, Yin QW, Hu YJ, Guo M, Fu XJ. Overview of researches on ontology. *Journal of Computer Research and Development*, 2004,41(7):1041-1052 (in Chinese with English abstract).
- [11] Zhang YX. Fundamental relations research in NKI ontologies [MS. Thesis]. Yunnan: Yunnan Normal University, 2002 (in Chinese with English abstract).
- [12] Zhao ZF, Han YB, Yu J, Wang JW. A service virtualization mechanism for business user programming. *Journal of Computer Research and Development*, 2004,41(12):2224-2230 (in Chinese with English abstract).
- [13] Hu JQ, Zou P, Wang HM, Zhou B. Research on Web service description language QWSDL and service matching model. *Chinese Journal of Computers*, 2005,28(4):505-513 (in Chinese with English abstract).
- [14] Liu Y, Ngu AHH, Zeng LZ. QoS computation and policing in dynamic Web service selection. In: *Proc. of the 13th Int'l Conf. on World Wide Web*. New York, 2004. 66-73.
- [15] Zeng LZ, Benatallah B, Dumas M, Kalagnanam J, Sheng QZ. Quality driven Web services composition. In: *Proc. of the 12th Int'l Conf. World Wide Web*. New York: ACM Press, 2003. 411-421.
- [16] Zeng LZ, Benatallah B, Ngu AHH, Dumas M, Kalagnanam J, Chang H. QoS-Aware middleware for Web services composition. *IEEE Trans. on Software Engineering*, 2004,30(5):311-327.

- [17] Su XL, Li JT, Jiang FQ, Shi HZ, Guo JB, Zhu ZM. A system of application services access based on virtual desktop. In: Proc. of the 1st Conf. on Harmonious Human Computer Environment 2005. Beijing: Tsinghua University Press, 2006. 636-642 (in Chinese with English abstract).
- [18] Zhu ZM, Su XL, Li JT, Guo JB, Ye J. A user-centric service framework for pervasive computing. In: Proc. of the 1st Int'l Symp. on Pervasive Computing and Applications. Urumchi, 2006.42-45.

附中文参考文献:

- [8] 岳昆,王晓玲,周傲英.Web 服务核心支撑技术:研究综述.软件学报,2004,15(3):428-442.
- [9] 刘昌树,李军,余海燕.GAS:网格环境下的一种 GUI 应用程序共享系统.计算机工程,2007,33(14):98-100.
- [10] 李善平,尹奇韡,胡玉杰,郭鸣,付相君.本体论研究综述.计算机研究与发展,2004,41(7):1041-1052.
- [11] 张宇翔. NKI 本体理论中一些基本关系的研究[硕士学位论文].云南:云南师范大学,2002.
- [12] 赵卓峰,韩燕波,喻坚,王建武.一种支持业务用户编程的服务虚拟化技术——VINCA 聚合服务机制.计算机研究与发展,2004, 41(12):2224-2230.
- [13] 胡建强,邹鹏,王怀民,周斌.Web 服务描述语言 QWSDL 和服务匹配模型研究.计算机学报,2005,28(4):505-513.
- [17] 苏晓丽,李锦涛,蒋发群,史红周,郭俊波,朱珍民.基于虚拟桌面的应用服务访问系统研究.见:和谐人机环境 2005.北京:清华大学出版社,2006.636-642.



朱珍民(1962-),男,湖南张家界人,博士,教授,主要研究领域为普适计算,嵌入式系统.



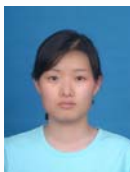
李锦涛(1962-),男,博士,研究员,博士生导师,主要研究领域为普适计算,多媒体技术.



蒋发群(1977-),男,博士生,主要研究领域为普适计算,面向服务的计算.



叶剑(1974-),男,博士生,主要研究领域为上下文感知.



苏晓丽(1979-),女,博士生,主要研究领域为普适计算.