

# 基于 Elastos 的普适计算环境研究

黄玉坤, 陈榕, 王芳

(同济大学基础软件工程中心 上海 200092)

**摘要:** 针对多数时间在线的消费电子网络环境, 如 3G 手机、数字电视等, 上海科泰世纪科技有限公司实现了 CAR (Component Assembly Runtime) 构件技术以及为其提供运行时支撑的 Elastos 网络构件运行平台。与传统的面向过程及面向对象编程模型相比, CAR 构件技术强调了目标代码级的软件构件制造及发布; 强调了利用 XML+脚本的形式进行软件构件装配; 强调了使用 URL 对软件构件编址的普遍性和重要性。借助于 URL 及 HTTP 等标准因特网基础设施, 网络服务器可以生成因时、因地、因人而异的 XML 脚本, 为人们动态提供最合适的软件构件。本文通过结合普适计算理论, 讨论基于 CAR 构件技术及 Elastos 平台来构建普适计算环境的可行性和方法。

**关键词:** Elastos; CAR; 构件技术; 普适计算

## Pervasive Computing Environment Based on Elastos

Huang Yukun, Chen Rong, Wang Fang<sup>1</sup>

(System Software Engineering Center of Tongji University, Shanghai 200092, China)

**Abstract:** Elastos, designed and implemented by Koretide (Shanghai) Co., is an embedded operating system to facilitate future consumer electronics, such as 3G Mobile Phones and HDTV's, which are connected to the network most of the time. CAR (Component Assembly Runtime) components run on Elastos. The three key features of CAR Technology are: (1) based on binary object code with class-information; (2) assigning a URL address to each CAR component; (3) class and method names of a CAR component become XML tags upon loading. Resort to ubiquitous Internet infrastructures, network portals may generate XML scripts, and dynamically assemble CAR components according to specific time, place and people. This paper discusses the potentials and benefits of adapting Elastos as a pervasive computing environment.

**Key words:** Elastos, Component assembly runtime, Component based programming, Pervasive computing

## 1. 研究背景

普适计算强调的是在广泛的空间(信息空间、物理空间及人类活动)上建立通用的计算环境。本文的重点是结合操作系统开发来研究普适计算的计算模型。我们认为: 对于普适计算中要求的计算透明性、资源分布性、硬件差异性、网络异构性和安全性等问题, 解决的关键在于要有一个良好的计算模型和系统软件支撑平台。

---

<sup>1</sup>收稿日期: 2007-; 修订日期: 2007-

项目基金: 国家移动通信产品研究开发专项项目: “面向服务的 3G 手机软件平台开发”, 财政部(财建[2005]182 号)、信息产业部(信部请函[2005]297 号)。

作者简介: 黄玉坤, 硕士研究生, 研究方向: 嵌入式操作系统、系统软件支撑技术; 陈榕, 同济大学基础软件中心主任, 科泰世纪首席科学家, 研究方向: 嵌入式系统, 构件技术; 王芳, 硕士研究生, 研究方向: 嵌入式操作系统、系统软件支撑技术。

Elastos 操作系统 2.1 版的发布, 表明完整的 CAR (Component Assembly Runtime) 编程模型已经具备了坚实的理论和实践基础<sup>[1]</sup>。Elastos 可以用在 3G 手机、GPS、数字电视、智能家电等各种电子设备中。在这些设备组成的计算环境的基础上, 抽象出的协作式分布计算模型将有助于普适计算信息空间的建设。

## 2. 相关研究

目前, 国内外对普适计算在人机交互、软硬件体系结构、计算模型、统一编程接口、系统自适应性、上下文敏感性、服务质量和计算安全等领域进行了广泛的研究。由于普适计算强调“无处不在”、“随时随地”提供计算服务, 因此, 移动通信设备和各种智能电子设备是普适计算的重要组成部分。在这方面, 美国斯坦福大学、CMU 和 Xerox 等研究机构关注于提高移动设备的交互能力, 通过网络在小屏幕设备上呈现 WEB 信息, 为用户提供计算服务的交互界面<sup>[2]</sup>。随着用户对计算服务的智能化要求越来越高, 提高计算服务的环境感知能力, 根据用户所处位置和触发条件, 主动预测计算服务和准备计算资源也是目前研究的热点。在上下文敏感计算方面, 研究比较典型的项目有 Stick-Enotes、Context Toolkit、Context Fabric<sup>[3]</sup>。另外, 上下文触发模型上, 1999 年 Stanford 大学的 Andrew C. Huang 提出了基于上下文触发的 ROME 模型<sup>[4]</sup>。针对 ROME 模型中的不足, 2004 年中科院软件技术研究所提出了新的触发模型 2CTM PC<sup>[5]</sup>。

普适计算的异构性不仅表现在操作系统、协议体系的不同和不匹配上, 普适环境还面临设备的计算资源和计算能力、智能水平的不均衡上。在自适应系统方面, Brian D Noble 提出了用于移动数据访问的系统和应用相协作的自适应模型——Odyssey<sup>[6]</sup>。关于中间件技术, 对普适计算而言, JINI 中间件技术是一种比较成熟的技术<sup>[7]</sup>。JINI 提供一族网络协议和编程接口, 建立一种使所有设备可以交互的软件平台, 该软件平台允许下层具有不同的操作系统和接口。其他在网络通信方面, 国内外的研究者主要侧重在网络资源的分布性、移动性、异构性及安全性等方面。

本文主要研究基于 CAR 构件技术的普适计算环境, 通过使用 CAR 构件技术与 Elastos 构件运行平台, 构建出一种新型的普适计算环境。

基于 CAR 构件的普适计算环境主要由以下几部分组成:

- CAR 软件构件元数据组织、编程、构件动态组装技术;
- CAR 构件运行时支持技术, 即 Elastos 网络构件运行平台技术;
- ElWidget 与 XmlGlue 编程技术;
- 操作系统仿真技术;
- 面向方面编程技术、泛类编程技术;
- Domain 沙箱技术。

下面将就 CAR 编程与构件模型、Elastos 构件运行平台的体系结构、以及 CAR 构件技术对普适计算环境各种特性的支持等方面进行阐述。

## 3. CAR 编程与构件模型与 Elastos 网络构件运行平台

### 3.1 CAR 的含义

CAR, 即 Component Assembly Runtime, 是在运行时对软件构件进行组装并最终完成预计功能的一种软件技术<sup>[8]</sup>。

在 Elastos 运行环境中, Component Assembly 包含了两层含义: (1) 软件零件, 特指“目标代码单元”。在 CAR 编程规范中就是 DLL; (2) 软件部件, 是软件零件的集合。一般是个“半成品”, 通过 XML 或脚本包装成为“产品”, 也可以直接是个“产品”。软件部件不但包含一个或一组 DLL, 还包含了数字签名、下载压缩包、应用描述文件、元数据信息等。类似于 JAVA 里面的 JAR 文件、Windows 里面的 CAB 文件。

Elastos 中 CAR 的含义是“基于 CPU 指令集的软件零部件运行单元”, 简单理解就是“软件零部件运行单元”。

### 3.2 CAR 的技术内涵

CAR 是一种基于构件的软件运行支持技术。构件运行支持能力直接决定所支持的构件的编写方法、结构设计, 甚至算法选择。CAR 支持满足“故障独立性”(即某个部件失效不会引起其它部件的失效, 是硬件系统可靠性的基本特性)的运行环境。CAR 通过这种环境所提供的构件动态组装, 对外完成预计的计算任务。

CAR 是一种构件化的开发语言, 它只负责框架部分描述, 具体的实现逻辑由 C/C++ 等编程语言实现。CAR 所描述的框架部分以元数据的形式存在于构件的发行格式中, 元数据通过反射(reflection)机制参与构件组装计算。框架是将具体的应用逻辑通过类似于微软 COM 构件的方式(计数管理、接口查询、构件聚合)隐藏起来, 并把自己暴露在外的最终运行封装。它提供了构件的标准, 二进制构件可以被不同的应用程序使用, 使软件构件真正能够成为“零件”, 从而提高软件生产效率。

CAR 支持构件被分布式配置在不同计算容器(构件容器、进程、线程)中, 从而实现分布式、协同计算。CAR 定义了构件在各种情况下的通信方式、故障处理方式、安全机制等。

CAR 还定义了一种软件工程化方法, 软件发行与配置策略, 从而定义了面向服务的软件业务模型。

### 3.3 基本 CAR 编程

CAR 的编程涉及到 CAR 构件本身的语法定义及开发、CAR 构件的引用等, 这里简要介绍一下 CAR 构件编程, 详细介绍参考文献<sup>[8]</sup>。

- CAR 编程涉及到构件的实例化(New)、接口的获取(Query)、构件生命周期管理(AddRef、Release)等三个部分。
- CAR 通过生成 C++ 框架程序, 然后用 C++ 实现程序逻辑, 从而完成软件开发。

- 回调机制支持的事件模型，而非消息机制（如 Windows 的消息机制），是真正的网络编程模型，如果回调绑定在序列化的类上，CAR 运行机制将自动保证引用的序列化，而不用应用编程人员自己关注引用的序列化。

#### （一）获得构件的实例

构件实例化（New）帮我们做了下面这些工作：

- 1、构件组装时构件的尺寸不知道，通过不定尺寸的支持，解决软件灵活的适配问题。
- 2、不知道软件运行“地点”，需要通过 ECode 携带程序执行路径上的出错信息。  
构件的定位是通过 URL 来标识的，可以配置于网络所触及的任意地方。
- 3、环境 context，环境会对构件有影响，藉此，实现 AOP 的支持。
- 4、按需计算（On Demand），New 之前，New 的实现代码还没有被加载。

在 CAR 中提供了 New 和 NewInContext 来创建构件对象，New 方法用来在同一个域内创建一个构件对象。NewInContext 方法也是用来创建一个构件对象，但用户可以指定该构件对象的对象环境即 context。

#### （二）获得构件的引用接口

Query 面向接口编程，与实现无关，不一定每个替换的构件都实现完全的功能接口定义。

每个 CAR 构件的接口都继承于 IObject。IObject 有以下四个方法（Query、AddRef、Release、Aggregate）。当编译生成一个构件的框架代码时，CAR 编译工具自动生成该构件重载的 IObject 的这四个方法。

#### （三）构件的生命周期管理

实时系统、嵌入式系统，对垃圾收集器（GC，Garbage Collection）式内存管理方法做不到可控，所以这里选用的是计数器管理方式实现构件生命期管理。

```
◎CARAPI_(ULONG) AddRef();
```

该方法把引用计数加 1，表示调用方已经复用了接口指针。更新之后的引用计数被返回。

```
◎CARAPI_(ULONG) Release();
```

该方法把引用计数减少 1，表示有一个接口指针已经被销毁了并且在当引用计数到达 0 的时候销毁对象。

#### （四）回调

callback 关键字用以声明一个 interface 接口是一个回调接口。回调接口中每个成员函数代表一个事件（event）。当特定事情发生时，如定时消息或用户鼠标操作发生时，构件对象产生一个事件，客户程序可以处理这些事件。构件对象中回调接口并不由构件对象实现，而是由客户端的接收器实现。接收器也是构件对象，它除了实现回调接口外，还负责与可连接对象进行通信。当接收器与可连接对象建立连接后，客户程序可将自己实现的事件处理函数注册，把函数指针告诉构件对象，构件对象在条件成熟时激发事件，回调事件处理函数。

CAR 编译工具会自动生成回调接口的 AddXxxCallback 和 RemoveXxxCallback（Xxx 代表回调事件名）函数提供给客户端使用来注册和注销事件处理函数。

### 3.4 Elastos 网络构件运行平台

Elastos 是一个传统意义上的通用嵌入式操作系统，支持多进程、多线程，抢占式、线程多优先级任务调度等特性，其内核叫 Zener。Zener 具有体积小、速度快、效率高的特点，适合于智能手机等消费类电子产品的嵌入式应用。其次，Elastos 作为一种网络构件运行平台，提供一个构件化的开发及运行环境。

Elastos 网络构件运行平台的实现遵循了以下三个原则<sup>[9]</sup>：

- (一) 采用 XML+脚本语言设计用户界面，并使用 XML+脚本语言来拼装 WEB 服务构件。
- (二) 用 URL 来对网页、构件进行编址，按需从网络上加载网页、脚本、构件。
- (三) 软件零件化生产，支持携带自描述信息（Class-Info）的 WEB 服务构件化编程。利用 C/C++、C#、JAVA 等编译语言实现 WEB 服务构件，并自动适配 XML 及脚本操作。

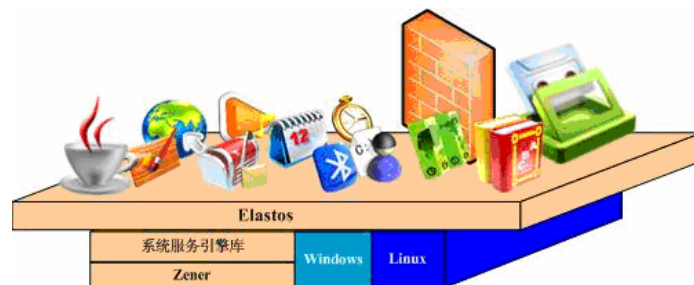


图 1 Elastos 网络构件运行平台

从图 1 可以看出，在网络构件运行平台上，普适计算环境中的各个运行的模块相对独立，每个模块都可以抽象成为一个独立的构件。由于传统的动态链接库 DLL 动态调用是基于地址（Address）及通过栈来实现传统参数的传递，DLL 尺寸及传递参数栈的改变必然导致函数地址的改变。所以难以做到应用的二进制动态升级与动态拼装<sup>[10]</sup>。在 Elastos 中，应用对构件 DLL 的功能调用是基于接口（Interface）的，各个子模块之间没有直接的调用耦合。WEB 计算服务通过 URL 来动态加载相关 DLL，创建构件对象实例。应用请求对 Web 服务的调用与接口顺序有关，与具体的函数地址无关。子模块的修改或升级只是在构件内部的局部的修改，更不会导致整个系统失败或产生应用的重新链接要求。所以，在 Elastos 网络构件运行平台上，可以方便地根据用户的服务请求进行二进制构件复用拼装。一次编译，重复可用。Elastos 支持的 CAR 构件技术为应用的二进制动态升级提供了基础支撑。

## 4. 基于 CAR 的普适计算

CAR 构件技术是一个内涵丰富的技术体系，它包括元数据计算、Elastos URL 自动软件构件寻址、反射机制、线程回调、XmlGlue 编程、构件运行时管理、面向方面的编程、泛类编程、Domain 沙箱技术等内容。它为普适计算的人机交互界面的设计、实现界面与控制逻辑的分离和动态换肤技术、软件构件的动态下载和按需加载、系统运行组件的自动配置和自适应、根据上下文动态聚合 Aspect 方面特性、第三方构件的安全验证和隔离运行等需求提供技术支持。

## 4.1 展示与控制、逻辑相分离的动态人机交互界面

基于函数式编程的思想，充分利用 CAR 构件的元数据计算能力，我们设计了面向最终应用“编程”的 XmlGlue 技术。

XmlGlue 是利用 XML 语言的计算描述能力，来描述程序的用户界面，同时对程序所需要使用的资源进行配置。当然，除了程序界面的描述，还需要采用脚本语言来完成界面上的一些逻辑的处理。由于 XML 和脚本语言都是解释执行的，以这样的编程方式，应用所需要的构件服务就像被胶水胶住一样在运行时动态地被 XML 和脚本语言粘结起来，共同来完成应用的需求。XmlGlue 基于 MVC 架构，它用 XML 语言描述视图（View），由 CAR 构件对应的模型（Model）进行逻辑处理，而 XmlGlue 则是作为控制器（Controller）部分，把 View、Model 两部分连接起来，从而实现界面与实现逻辑的分离。这种基于构件中间件技术的 MVC 编程模型能够很好屏蔽不同厂家的底层软件技术的差异，用户可以很方便地在同一个 UI 视图下采用不同厂家或不同版本号的软件引擎，也可以在不同的 UI 视图下重复使用同一个厂家提供的引擎。

XmlGlue 在 Elastos 构件平台中粘合的对象是一个个的构件，它的粘合可以使功能分立的构件组合成一个可以向用户提供完整功能的应用实体。从另一方面说，由于它的存在，消除了构件间的耦合，使构件只需符合 XmlGlue 的接口标准，就能和其他构件协同工作。

Elastos 提供了一个完整的运行时环境：XmlGlue Runtime。XmlGlue Runtime 是一个支持 XmlGlue 的中间件平台，可以运行在 Elastos、Linux、WinCE 等多个操作系统之上，对 XmlGlue 的应用而言，Runtime 完全屏蔽了操作系统的差异。XmlGlue 继承了 CAR 技术的特性，所以，只要符合 CAR 标准的组件，都可以在 XmlGlue Runtime 中使用。这样使得 XmlGlue 应用具有很好的扩展和移植能力。动态换肤技术是用 XmlGlue 技术实现的。它通过本地化引擎与 XML 动态匹配解释，实现在不需要修改本地二进制引擎代码的情况下，只是简单的替换 UI 的 XML 描述语言来重新封装原有的构件化应用引擎，再结合 JavaScript 实现逻辑控制流操作，从而达到迅速的更换人机交互 UI 的效果。

ElWidget 是基于解释型语言（XML、JavaScript、Lua 等）与编译型语言（C/C++、JAVA 等）的一种混合语言编程技术。基于 XmlGlue 技术和 CAR 构件技术，Elastos Widgets 引擎可以动态扩展，完成通用计算平台的功能。采用这种技术实现的 Widget 程序是最小的计算服务单元，经代理协调机制对外提供人机交互、事件感知服务。该技术非常适合消费类电子产品——处理小量数据，促进脚本语言与个性化引擎间互动。用这种编程技术开发出来的迷你小程序（Elastos 小精灵）已被应用在和欣智能手机上，用户可把从某个 WAP 网站上搜集的数据置入一个 Widget 中。比如把股票信息定制化搜索框加入到用户的桌面上，或把某个视频加入到来电提醒中。甚至也可以创建一个综合了用户网站信息、SMS、日程及相册等内容在内的完整的 Widgets 桌面。

## 4.2 软件构件的动态下载、按需加载

基于 Elastos 灵活内核、URL 自动构件寻址技术，可实现软件模块的点击下载，按需计算（On Demand），支持无缝计算（Seamless Computing）。先启动应用，再启动服务的软件模型，会从根本上改变普适计算平台的可移动性，增强计算平台的鲁棒性。

CAR 构件技术，为 C/C++ 语言添加契约导向（面向接口）编程描述，对目标软件模块进行封装，按照规范实现软件工厂化生产，支持不同版本软件构件的互操作，允许不同生产厂家软件的相互替换，实现不同语言（JAVA、C# 等）构件相互调用。在软件制造商之间互不公开源代码的前提下，以目标代码为单位，动态拼装不同厂家的软件构件，完成更大的功能模块。只要不同厂家的软件构件符合 CAR 构件的接口标准，就可以通过 URL 等全球唯一标识技术，将软件模块直接在因特网范围编址，如：[//www.elastos.com/car/drivers/tcpip.dll](http://www.elastos.com/car/drivers/tcpip.dll)。

### 4.3 灵活的动态可配置机制与自适应性

Elastos 构件运行平台与 CAR 构件技术，具有较强的计算环境适应能力。CAR 构件技术通过 XML、URL 和元数据对任务进行描述、分解和分布式管理。在普适计算网络上不仅可以实现对进程或构件的调度，还可以实现通过 XML 描述来对它们进行动态组合，这意味着对应的应用引擎可以与操作系统无缝衔接与升级。

以网络自适应性为例，受到所处网络环境的限制，网络的连接性和远程资源对终端来说都是动态变化的。普适计算终端的环境是变化的，可得到的资源，也是变化的，并且这种变化是动态的，不可预期的。这就要求终端具有自适应的环境处理能力。

在 CAR 编程模型中，支持泛类编程技术，它使 Elastos 构件运行平台的动态自适应性更加灵活。generic 被称为“泛类”，定义这样一种类的目的是为了抽象出各个一般类所具有的共同类或接口特性，并且在定义这个泛类的 New 函数时，可以根据硬件条件的变化自动选择创建不同的具体的类。只要这些具体的类都是从这个泛类继承而来，创建这些具体的类时直接使用泛类的 New 函数就能完成，客户端不需要关心到底是创建哪个具体的类。

Elastos 对“类对象”（Class-Object）进行编程，重载创建构件的 New() 操作。比如，创建网络构件的时候，动态检查当前什么通讯协议信号最强，然后采用 GSM、WiFi、TD-SCDMA 等构件之一作为网络构件。CAR 构件编程模型支持透明网络切换，比如：从室内 WiFi 切换到室外 TD-SCDMA 网络。比如支持当前计算设备与周围环境的多种通信模式自选择性地沟通，组成 P2P、P2S、Push-on-Call（PoC）等多种协作网络。

在普适计算环境下，大多数的用户终端是系统资源和计算能力都有限的小型移动通讯终端，它们在设计方面存在以下需求：功耗低，重量轻和物理尺寸比较小。根据这些需求设计出来的设备所拥有的资源在以下方面存在不足：外存（通过磁盘、FLASH 等提供）容量，物理内存大小，处理能力，缓存大小等。

利用 Elastos CAR 编程模型开发的软件模块，特别适合实时、高效、低资源消耗的系统软件应用。不用任何修改就可以跟 JAVA、.NET 等技术实现的 Web 服务通过 SOAP 进行互操作。因为 Elastos 软件模块的运行效率和相互之间的调用操作，在效率上可以与传统的 C/C++ 软件模块相媲美。因此，Elastos 软件既具有 JAVA 软件的灵活性，同时还具有 C/C++ 软件的效率和代码的精炼。

### 4.4 操作系统仿真技术

操作系统仿真就是在一个宿主操作系统之上，利用软件或固件的手段建立一个虚拟层，模拟另外一种寄生操作系统的行为和功能的<sup>[9]</sup>。在一个底层操作系统之上，可以同时实

现多个彼此独立的仿真操作系统。在这些被仿真的操作系统之上可以运行它们原来固有的应用程序，而不需要对这些应用程序进行修改。

操作系统仿真技术又可以大致分为两种：静态源代码级编译仿真和动态二进制模块运行仿真。静态仿真技术是在宿主操作系统的开发平台上实现一个或多个模拟寄生操作系统的程序调用接口（API）的函数库。为寄生操作系统编写的应用程序的源程序可以经过编译之后与那些仿真函数库链接，形成可以在宿主操作系统之上运行的目标代码。动态仿真技术就是能够使寄生操作系统的可运行目标代码直接拿到宿主操作系统上运行的技术。

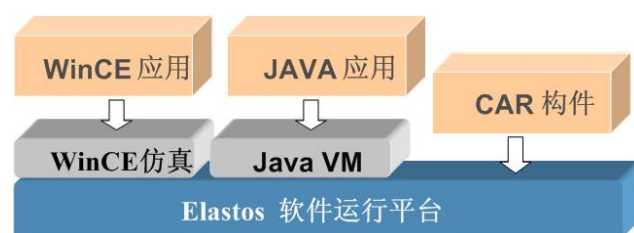


图 2 Elastos 已经支持 JAVA、WinCE 应用程序仿真运行

在 Elastos 操作系统之上实现了对 WinCE 的静态源代码级仿真，也就是把为 WinCE 开发的应用程序源代码拿来，重新跟 Elastos 的函数库链接就可以运行于 Elastos 之上。在 Elastos 之上的 WinCE 代码的二进制兼容也已经初步实现。

Elastos 应用的 x86 二进制代码在 Windows XP 和 PC Linux 之上运行的效率与现有的 Windows XP 和 Linux 二进制代码的运行效率几乎完全一样，Elastos 使用的 C/C++编译器与 Windows 和 Linux 使用的 C/C++编译器是完全一样的。也就是说，只要 C/C++源程序相同，不论是用 Elastos 开发环境还是用 Windows 开发环境产生的二进制可执行代码都是一模一样的。

因此，使用 Elastos 构件运行平台作为普适计算环境的系统支撑软件将能在一定程度上抵消各种系统之间的差异性。

## 4.5 面向方面的编程

Elastos 提供的基于二进制的 AOP 编程环境，可以灵活的实现基于构件级别的代码的动态插入、动态聚合<sup>[11]</sup>。在普适计算环境中，强调上下文敏感性。如同种计算服务可以随着人所处的环境不同而采取不同的服务标准或服务策略。在 Elastos AOP 编程模型中，我们将这些标准和策略抽象成一个一个的方面构件，它们可以动态地附加到进入它们所在计算环境的计算实体上，使这些计算实体具有这些方面构件所代表的特征或功能。

Elastos 构件运行环境可以将一个一般 CAR 构件类与一个或多个 Aspect 方面 CAR 构件类动态聚合起来，从而生成一个具有两个或多个 CAR 构件类所有接口实现的新构件类。一般来说 Aspect 方面是一个 Context 上下文语境的特征。所谓的具有环境特征或失去环境特征，就是 Context 上下文语境会为对象动态地聚合或拆卸一个或多个 aspect 对象。这些 aspect 对象是 Context 上下文语境在 CAR 文件中定义时指定的。

一个对象如果进入了上下文语境，那么该对象将具有此语境的特征，一旦对象离开了语境，环境特征就会失去。但该对象很有可能又进入了另外一个语境，拥有新的环境特征。当一个用户进入了某个商店，那么他就应该具有这家商店的顾客的特性，就具有了消费行为。



我们可以把“消费行为”看成是一个 aspect 对象,那么“用户进入了某个商店”就是一个上下文语境。只要有消息能够触发“用户进入了某个商店”这个条件,就可以通知应用中的构件对象进入“用户进入了某个商店”上下文语境,并聚合“消费行为” aspect 对象,这样应用就为用户接下来的消费准备好计算服务。当用户离开了商店,则用户就暂时不再有在这家商店的“消费行为”,也就是离开了“用户进入了某个商店”的上下文语境,则应用中的构件对象将与“消费行为” aspect 对象动态拆卸。随着构件对象的环境(context)的改变,构件对象所具有的特征(Aspects)也可能是不一样的。因此,随时聚合、随时拆卸是 Elastos 面向方面的聚合模型的最大特点。

## 5. 结束语

普适计算是一种突破了桌面系统的计算模式。它为人提供以人为本的更自然的人机交互环境和更灵活、更广泛的计算服务接入方式,使人们能随时随地透明地获得各种计算服务。普适计算反映了人们对信息服务的更高需求,展现了信息空间、物理空间及人类活动三者无缝融合的前景,是计算机科学与技术发展的必然趋势。

Elastos 构件运行平台,及其支持下的 CAR 构件技术,具有较强的移动计算环境适应能力。CAR 构件技术所提出的构件编程模型和运行模型能够很好地支持普适计算环境的实现。面向 WEB 服务设计的 Elastos 网络构件运行平台能够为普适计算这一新型的分布式计算模式提供强大支持。

### 参考文献:

- [1] 上海科泰世纪技术有限公司,Elastos 资料大全 [EB/OL].<http://www.koretide.com.cn> . 2006-10
- [2] Buyukkokten O. , Garcia2Molina H. Power browser; Efficient web browsing for PDA s [A ]. In : Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'00 ) [C ]. Hague, Netherlands, 2000. 430 - 437.
- [3] 岳玮宁,董士海,王悦,等. 普适计算的人机交互框架模型 [J]. 计算机学报 , 2004 , 27 (12) : 1657 - 1664.
- [4] Andrew C. Huang, Benjamin C. Ling, Shankar Ponnekanti. Pervasive computing: What Is It Good For? [C ]. The International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE) in conjunction with ACM MobiCom'99, April 1999.
- [5] 罗俊伟,秦晓,陈思功等. 普适计算中基于上下文触发的事务模型,小型微型计算机系统,2004,25(8):1542-1545.
- [6] Brian D Noble .Mobile Data Access[DB/ OL] . Carnegie Mellon University , <http://www.cs.cmu.edu/afs/cs/project/coda/Web/docdir/bnoble2thesis.pdf> . 1998 .
- [7] Gupta R,Talwar S,Agrawal D P.Jini Home Networking:A Step Toward Pervasive Computing. IEEE Computer,2002-08
- [8] 上海科泰世纪技术有限公司 , CAR 构件技术与编程模型 [EB/OL]. <http://www.koretide.com.cn>. 2006-10
- [9] 上海科泰世纪技术有限公司,Elastos 技术白皮书 [EB/OL]. <http://www.koretide.com.cn>. 2006-10
- [10] Robert Clemenzi. DLL Hell. [DB/ OL ]. [http://mc-computing.com/Languages/DLL\\_Hell.htm](http://mc-computing.com/Languages/DLL_Hell.htm). 2007-3-26
- [11] 郭强,范典,陈榕等.CAR 中基于上下文 AOP 机制的实现[J].计算机应用研究,2005 年第 10 期:P220-222.