维普资讯 http://www.cqvip.com

OSGI的分析和实现及其改进思路*)

熊 江
(重庆三峡学院计算机科学系 重庆万州404000)

摘 要 从消费类电子产品到各种工业设备,嵌入式系统已渗透到我们日常生活的各个角落,嵌入式设备的内存容量限制了用户并发使用多个服务的个数,本文借助操作系统中的虚拟内存思想,提出了虚拟OSGI框架模型,很好地解决了嵌入式设备上的内存对OSGI框架的限制。

关键词 OSGI,普及计算,虚拟OSGI框架,虚拟内存

The Analysis & Realization of OSGI and Improvement on It

XIONG Jiang
(Department of Computer Science, Chongqing Three Gorges College, Wanzhou 404000, China)

Abstract From consumer electronics to different kinds of industrial equipments, embedded system has played an important role in our daily life, the capacity of memory of the embedded equipment has restrained users from erupting the number of using a lot of services simultaneously. Based on virtual memory thought of operating system, this paper has put forward virtual OSGI Framework model , has well solved the restriction on OSGI Framework of memory on the embedded equipment.

Keywords Open service gateway initiative, Pervasive computing, Virtual OSGI framework, Virtual memory

1 引言

在21世纪这个普及计算(Pervasive Computing)时代,输入设备种类越来越多,数据的交易量增加,获取数据的方式也越来越多,手持设备、车载设备、办公室电脑、家庭电脑所有的信息都通过网络交互,信息交流使我们的工作生活变得更加简单,它将带来人机交互的无缝连接。嵌入式系统和手持设备的种类越来越多,尺寸也越来越小,使人们的期望越来越大,希望提供的服务更多更有弹性。

开放服务网关组织(OSGI:Open Service Gateway Initiative)是一个非营利性组织,致力于提供从WAN到LAN及其它设备间多种服务传送的开放式规范,OSGI框架规范就是该联盟制定的一个规范,至今已有包括IBM、Hp、Cisco、Motorola、Nokia在内的80多个公司已声明完全支持其规范。

OSGI框架规范的最大特点是其开放性、易扩展性。系统具有良好的动态装御机制,即在整个系统无须下线的情况下能够按应用的需要终止、启动、安装及御载软件的某些部分。OSGI框架可以通过现有的Java标准如Jini或者非Java标准HAVI全部或部分地集成到现有的系统如机顶盒、路由、家居网关、警报系统、cable modem等。OSGI组织已经发布了一套软件规范,该规范定义了一个开发式的软件框架。它允许不同的服务软件能够在服务网关上装载和运行。

2 OSGI的不足

嵌入设备和手持设备上都很少配置大的内存,绝大多数没有硬盘,尽管随着社会的发展,内存和硬盘的费用已经越来越低,但假使给这些设备配备大容量的内存,仍然有可能因为动态可装载的应用耗尽它的内存,就像现在拥有大容量内存的PC机也因为用户运行过多的应用程序而耗尽了内存。OSGI服务框架显然没有考虑这个问题,这给人们使用这些服务带来了一定的限制。

OSGI框架提供一个运行时的框架,负责导入、安装、激活、执行以及卸载服务在OSGI中,这些服务是一组内建的,用户可配置的应用,称为Bundle。一个Bundle包含一个配置描述文件(称为Bundle Manifest),一些Java类和资源。Bundle是OSGI服务框架的一个核心元素。可是,OSGI框架恰恰缺乏对此有效管理,即使有一些管理,那也是局部的。

Bundle一般可分为可中断的Bundle和不可中断的Bundle,如果一个不可中断的Bundle因为运行时间持续过长就可能会因为其他的Bundle不能及时运行而导致问题的发生,这在实时系统领域尤为突出,而嵌入设备和手持设备很多就具有实时的要求。在OSGI框架描述文件中,还没有相关的语句。

OSGI服务框架的设计目标就是为那些小内存设备对象服务的,并没有考虑上述问题,随着人们生活水平的不断提高,各种嵌入式设备,应用,服务铺天盖地的涌进人的生活,这些问题将变得越来越突出,人们由于这些内存和硬盘的限制而无法同时享受到所希望的服务。所以,作者提出了虚拟OSGI框架的思想,来解决此问题。

3 虚拟OSGI框架的设计

为了能够解决上述内存限制问题,同时又具有OSGI的所有优点和特性,作者借鉴操作系统中的虚拟内存思想提出了虚拟OSGI的概念。

虚拟内存(VM:Virtual Memory)是指高速磁盘上的一片存储空间,其功能是通过硬件、软件的办法,可以将其作为主

*)重庆三峡学院资助科研项目(No.20030102)。熊 江 讲师,主要研究方向为智能多媒体技术,微机接口技术。

维普资讯 http://www.cqvip.com

存储器的扩展的存储空间一样来使用，这就使得程序设计人员能够使用比主存储器实际容量大得多的存储空间来设计和运行空间一样来使用。在传统的虚拟内存机制中，一个局部的、二级存贮设备（例如硬盘）被用来存贮非活动的内存页，当这些非活动的内存被应用访问时，他们被重新调回主内存。

3.1 设计虚拟 OSGI 框架的思想

虚拟 OSGI 框架（Virtual OSGI Framework）是指创建了一个很大的内存空间运行更多的并发程序。但一个嵌入式系统是没有办法配备一个硬盘来实现虚拟内存机制的。为此，虚拟 OSGI 框架使用一个远端服务器去存贮非活动的内存页。然而，由于网络存在潜在的延迟，在内存页层次上进行交换是不实际的，这是因为设备不能容忍难以接受的响应时间，同时对在远端服务器上的内存页的管理也非常复杂。所以，本方案不采用在内存页这个层次上进行交换，而是在应用层交换内存，即将整个应用在不需要的时候交换到远端服务器，在需要的时候再从远端服务器交换回来。

从单个过程来看，交换应用比交换内存要花费更多时间。但是，由于交换发生在远端服务器和 OSGI 框架之间，交换内存页将使得正处于激活状态的应用程序不能响应。再者，OSGI 是建立于 JVM 上的，Java 的 byte-code 程序要比其他语言写成的程序要紧凑。另外，嵌入式设备上的“服务”的大小趋向越来越小，尤其是用来通信和控制的应用。在这些应用中，用户界面趋向于简单，从这几点来说，一个好的应用程序设计核 Java byte-code 的压缩效果能够使 OSGI 服务变得越来越小，完全可以获得可以接受的装载延迟，所以交换应用程序比交换内存页并不更需要更多的时间。因此，采用在应用层交换的方案是可行的。通过维护在远端服务器上的当前应用，并在需要他们时将它们交换进来。

虚拟框架的管理者提供一个大的 OSGI 配置平台，通过维护在远端服务器上的当前应用，并在需要它们时将其交换进来。

提出虚拟 OSGI 框架的目的是为了解决嵌入式设备上物理内存大小限制了 OSGI 框架上运行并发应用个数；提出虚拟 OSGI 框架目标是使得更多的并发应用程序运行在服务网关上。

3.2 实现虚拟 OSGI 框架的关键问题

实现虚拟 OSGI 框架的关键问题是实现应用层上的交换，这就要求虚拟 OSGI 框架必须能够知道何时内存被耗光，哪个应用应该被交换出去，哪个应用应该被交换进来。同时，虚拟 OSGI 框架还应该能够让服务提供者很容易的发布服务、管理服务等等。结合 OSGI 规范，用 UML 用例图模型对其功能进行描述，如图1来解决应用层上的交换问题。

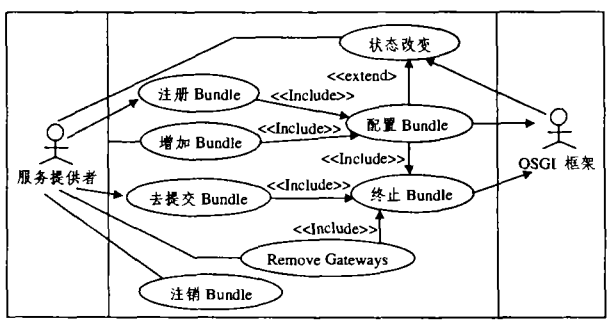

图1 虚拟 OSGI 框架系统的用例图

下面分别说明图1中的各个用例含义。

注册 Bundle：服务提供商注册一个新的服务 Bundle，这些 Bundle 将被配置到一台或多台服务网关上。对每一个注册了的 Bundle 将给定一个配置策略以指导如何管理 Bundle。这个用例将激活配置 Bundle 用例在目标服务网关上配置新的服务。

增加 Gateways：服务提供商指定一台或多台服务网关接受已经存在的、已注册过的 Bundle。

删除 Gateways：服务提供商指定一台或多台服务网关不再运行已经存在的、已注册过的 Bundle。

配置 Bundle：这是一个抽象的用例。它被其他用例激活去配置一个新的或处于睡眠状态的 Bundle。

终止 Bundle：这是一个抽象的用例。它被其他用例激活去终止一个已注册的、可能正在一台或多台服务网关上运行的 Bundle。

通知状态改变：OSGI 框架通知本地 AdminBundle，它将通知虚拟框架管理（Virtual Framework Manager，VFM），驻留在网关上的 Bundle 已经改变了它的生命周期状态。如果运行在所有目标网关上的 Bundle 的所有实例报告状态改变了足以引起 VFM 上的主 Bundle 改变状态，VFM 将利用 URL（来自最初的请求）发送给服务提供商一个通知。

注销 Bundle：服务提供商请求 VFM 注销它先前注册的 Bundle。

3.3 虚拟 OSGI 框架结构模型

虚拟 OSGI 框架不仅需要和服务提供者打交道，还需要同 OSGI 进行通信，对服务进行管理，提供 OSGI 规范中的所有功能，从结构上来看，它主要由以下4个组成部分组成：虚拟框架管理（Virtual Framework Manager，VFM）、管理 Bundle、服务提供接口和网关接口。

它们之间的关系如图2所示。

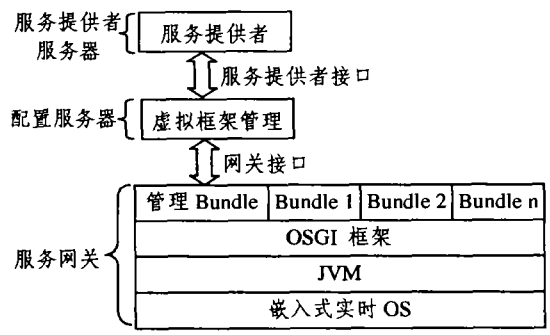

图2 虚拟 OSGI 框架组成

虚拟框架管理（Virtual Framework Manager，VFM）是服务提供商用来请求配置服务的一个远端服务器。OSGI 服务被打包在称为 Bundle 的配置单元中，它由包含了组成服务的 Java 类的 JAR 文件和一个描述 Bundle 属性的称为 Manifest 的文本文件。当应用需要交换的时候，VFM 使用由服务提供商提供的配置策略来讲应用交换进来或者交换出去。VFM 介于服务提供者和 OSGI 框架之间，是它们之间的桥梁，是虚拟 OSGI 框架体系结构的主要部分。

管理 Bundle（Admin Bundle）是运行在目标 OSGI 框架上的一个 OSGI 服务。它扮演着一个远程管理代理的角色，方便配置和管理 OSGI 的 Bundle。它也能够监控通过局域网连接在服务网关上的消费类设备发出的请求或者消息。例如，Admin Bundle 可能检测到从 IP 电话发来的 VoIP 消息；然后

维普资讯 http://www.cqvip.com

她通知 VFM，VoIP 服务必须交换进来，以提供服务。

服务提供接口支持在服务提供商和 VFM 之间进行数据交换和数据请求。

网关接口支持在 VFM 和 Admin Bundle 之间进行数据交换和数据请求。

结束语 利用 OSGI 的应用动态装载能力，虚拟 OSGI 框架不仅能够动态按需装载应用程序，而且解决了嵌入式设备上内存对 OSGI 框架的限制，还初步解决了普适计算模式中的应用框架问题。本文提出了虚拟 OSGI 框架模型，但是没有将它具体实现，只是给出了虚拟 OSGI 框架结构模型。在 OSGI 标准下，实现虚拟 OSGI 框架，构建普适计算应用的平台，将是作者下一步迫切需要进行研究的工作。

参 考 文 献

1 王诚. 计算机组成原理. 北京：清华大学出版社，2001

2 Szymanski J W. Embedded internet technology in process control devices，IEEE 2000

3 Chen K. Programming Open Service Gateways With Java Embedded Server. US：Addison Wesley pub co (sd). 2001

（上接第191页）

大，类的数目过多会影响设计的成本和延长软件开发周期。下面介绍通过类的分解和合并来确定类的合适的规模大小。

4.1 分解

所谓的分解就是将规模较大、类自我的依赖度低的类逻辑分解成若干个子类，分解后的每个子类都应该具有一定的逻辑意义，并且每个子类内部依赖度较高，子类间的依赖度较低。具体分解算法如下：

```
1)计算待分解类 B 自我的相对依赖度 rd
2)IF 类 B 自我的相对依赖度 rd 较小
    按照类内部的逻辑关系将类 B 初步分解为子类 B1，B2
  ELSE
    类 B 不能分解，跳到 5
  ENDIF
3)B=B1，转到1
4)B=B2，转到1
5)结束
```

如图2所示，类 B 规模较大，自我的依赖度较低，通过分解算法将类 B 分解成子类 B1，B2，…，Bn。

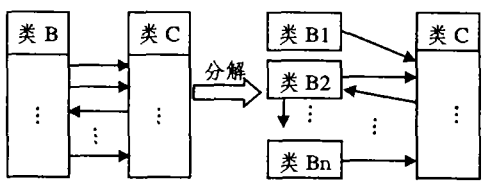

图2 将类 B 分解并降低其依赖

4.2 合并

所谓的合并就是将类之间的依赖度高、规模较小的两个类合并成一个类，使得原先错综复杂的类之间的依赖关系被隐藏。但是，需要注意一点：合并后的类的规模不应该过大。显然，类的合并过程是类的分解过程的逆过程。具体合并算法如下：

```
1)计算类 B 和类 C 的相对依赖度 rdB，rdC
2)IF rdB 且 rdC 较大
    合并类 B 和类 B 为类 BC
ENDIF
```

如图3所示为类 B 和类 C 的合并过程。

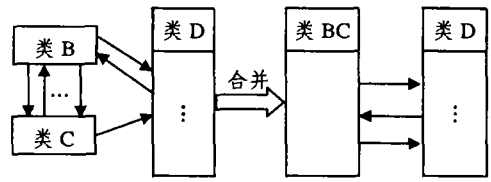

图3

结束语 依据类的依赖关系来分解或合并类的方法，只是为我们在类的分析和设计阶段提供指导方法，我们不可能也没有必要给出一个类具体的依赖度为多少才合适，或者给出依赖度合理的范围。设计者可以在不同的情况，针对具体不同的类，综合利用这两种方法，确定一个规模大小、依赖度都比较合适的类。

参 考 文 献

1 Priestley M. Practical Object-Oriented Design With UML. McGraw-Hill Companies，Inc，2000. 111～152

2 Booch G，Rumbaugh J，Jacobson I. Unified Modeling Language User Guide(in Chinese). Beijing：China Machine press，2001

3 Rumbaugh J，Jacobson I，Booch G. Unified Modeling Language Reference Manual(in Chinese). Beijing：China Machine press，2001

4 周之英. 现代软件工程(中). 科学出版社，2000. 358～428

5 方菲，等. 面向对象软件回归测试技术研究. 软件学报，2001，12(3)：72～75

6 Jones C. Applied Software Measurement，Mcgraw-hill，1986. 36～126

7 Fenton N E，Software Metrics，A Rigorous Approach，New York：Chapman & Hall，1991. 45～78

8 Chidamber R，Kemerer F. A metrics suite for object-oriented design. IEEE Trans on Software Engineering，1994，20：45～66