

一种基于CCM的多媒体中间件架构

韩健, 贺樑

(华东师范大学计算机应用研究所, 上海 200062)

摘要: 分析了数字家庭中多媒体应用领域存在的问题, 提出一种基于 CCM 构件技术的数字家庭多媒体中间件 DHMM, 其可以很好地支持各种数字设备间的多媒体数据的传输、管理、协同、保存、回放等功能, 有效地降低上层应用程序的开发难度, 提高开发效率。通过实验验证了该架构的可行性及有效性。

关键词: 多媒体中间件; CORBA 构件模型; 构件; 实现架构; 数字家庭

Framework of Multimedia Middleware Based on CCM

HAN Jian, HE Liang

(Institute of Computer Applications, East China Normal University, Shanghai 200062)

【Abstract】 This paper analyzes the issues in the application development of multimedia for digital home. A new middleware is raised which is based on CORBA Component Model(CCM) and named as Digital Home Multimedia Middleware(DHMM). DHMM can finely support the transmission, management, synchronization, storage and playback among the devices in digital home, thereby simplifying the development of high-level application and increasing the development productivity. The application in example system verifies the feasibility and validity of the framework.

【Key words】 multimedia middleware; CORBA Component Model(CMM); component; implementation framework; digital home

1 概述

家庭中的数字设备已越来越普及。随着科学技术的不断发展, 这些设备的功能得到了巨大的增强。但是设备间的互联与协作却进展缓慢。随着网络技术和无线网络在家庭中的普及, 物理层上设备间的互联已经不成问题。另一个方面, 这些数字设备正朝着智能化的方向发展。很多设备具有可编程性, 如数字电视机顶盒、手机, 甚至还有高端的网络冰箱。由此推断普适计算环境正逐步在家庭中形成。这些设备虽然在物理上通过网络连接在了一起, 可彼此的应用还是非常的独立。除了简单的以电脑为中心的协作, 很难彼此形成一个整体, 其主要原因是缺少有效的中间件架构。这为设备应用程序的开发带来了巨大的困难。本文借助构件技术设计了数字家庭多媒体中间件(Digital Home Multimedia Middleware, DHMM), 以解决数字家庭在多媒体娱乐中会遇到的问题。

2 数字家庭示例

图1给出了一个数字家庭的多媒体系统。

为了分析的需要, 本文简化了设备节点的个数, 并选用 MPEG2 作为实验的媒体格式。但是该架构具有很好的可扩展性, 可以支持大量的节点, 并只需要添加相应的模块(构件)就可以支持更多的格式。

在客厅内有一台大尺寸的等离子电视, 它与一台安装有 Linux 系统 PC 相连接。PC 中含有一块 VWeb 的模拟电视卡, 具有实时 MPEG2 的编码能力。电视卡与本地的有线电视(模拟)链接, 可以接收大量电视频道。

在书房内有一台高性能的 PC 与 Internet 相连接。同时可以接收卫星电视的信号, 如 DVB(Digital Video Broadcasting)等数字信号。PC 具有很好的性能, 可以将各种类型的多媒体

码流转换成 MPEG2 的码流。

在卧室内配有机顶盒的电视机。在实验的系统中, 机顶盒选用基于 Sigma Design 的 EM8621L 平台。其具有解码 MPEG1,2/4 及 WMV 等多种格式高清(1080P)码流的能力。

厨房中也同样有一台液晶电视, 其内部嵌入了笔者开发的机顶盒硬件, 在功能上与卧室内的相同。

以上系统通过 100 Mb/s 以太网相链接 构成家庭局域网。SB2 通过 54 Mb/s(802.11g)连入家庭局域网。

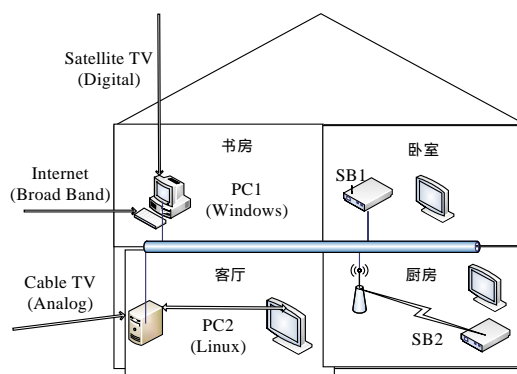


图1 数字家庭的多媒体系统

这时会有如下需求:

(1) 在任意电视机(SB1, SB2, PC1, PC2)上观看喜欢的节目。不必关心节目是来自有线电视还是卫星电视或 PC。

基金项目: 上海市信息化专项资金资助项目“无线数字电视平台及其核心软件系统”(沪信息委产[2006]316号)

作者简介: 韩健(1982-), 男, 硕士研究生, 主研方向: 普适计算, 中间件技术, 多媒体技术; 贺樑, 副教授、博士

收稿日期: 2007-07-10 **E-mail:** jhan@ica.stc.sh.cn

(2)在任意的电视机上,可以通过遥控器选择需要录制的节目。不必关心节目会录制到哪台电脑上,系统会根据信号源选取存储的位置。

(3)用户可以将观看的喜好或其他相关信息保存到系统中。在不同的电视机上,都可以通过菜单访问到以上信息。

(4)系统需要支持多用户同时使用。

总结以上需求,系统将会遇到如下的挑战:

(1)需要具有访问、控制并整合来自不同设备的多媒体码流的能力。

(2)需要能够同时支持多个节点。

(3)需要屏蔽信息源的异构性。

(4)需要屏蔽由上下文环境所带来的复杂性。

(5)需要屏蔽由动态配置带来的复杂性。

3 设计思路及 CCM 介绍

为了达成以上目标,需要在架构设计中引入中间件。中间件可以有效地屏蔽普适计算环境带来的复杂性,从系统的层次上提供对异构性的屏蔽。在中间件模型中,越来越多的软件开发人员选择基于构件的开发思路。引入构件可以很好地模块化系统,从而降低系统的耦合度,实现分而治之。更重要的是,构件实现了代码的高层次重用,为快速开发与部署提供了可能。基于构件技术的中间件能够很好地屏蔽系统的复杂性,为应用程序的开发带来巨大的便利。

未来普适计算环境中的应用开发都会使用这种思路。构件技术作为比较成熟的技术将在未来的普适计算中广泛地使用。在实际中可能需要为不同的普适计算场景设计不同的中间价架构,如在家庭的多媒体应用中设计 DHMM。但是在中间件的设计中,使用构件技术将成为标准。

目前实现的构件技术主要有OMG组织的CORBA构件模型(CORBA Component Model, CMM)^[1]、Microsoft的COM^[2]技术以及SUN的EJB^[3]。其中,CCM是为了与Microsoft Transaction Server (MTS)/COM+, .Net(平台专有的)和Java(语言专有的)竞争提出的。CCM实现了CORBA与Java, COBOL, COM/DCOM, C++, Ada, Smalltalk等的无缝集成。有关CCM与EJB互操作的详细信息见文献[4]。

CCM 构件平台体系结构如图 2 所示,其主要组成技术包括 4 方面的内容:定义构件接口特性的抽象构件模型;定义构件实现的构件实现框架;为构件定义其运行环境的容器编程模型及定义构件如何打包、组装、部署的组装部署模型。

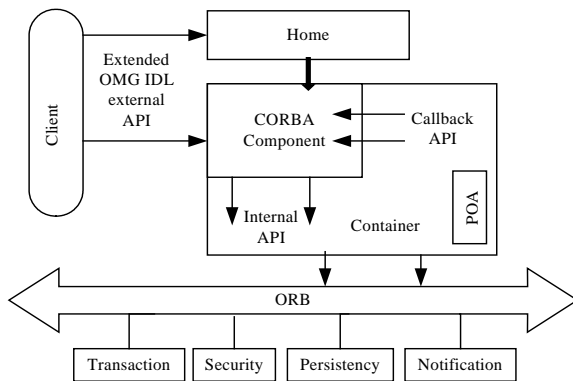


图 2 CCM 构件平台体系结构

CCM抽象构件模型提供了定义CCM构件外观特征的方法,见图 3。根据此构件模型,构件设计者可利用接口定义语言IDL定义所需构件,经过IDL 编译器生成CORBA 的Stub

和Skeleton。其中,构件通过剖面(Facet)和事件槽(Event Sink)定义构件向外提供服务的接口,通过接插口(Receptacle)和事件源(Event Source)定义构件依赖于其他构件提供服务的所需服务接口,通过属性(Attribute)定义构件在运行时可配置的特性。构件需要定义Home,通过工厂模式实现对构件的生命周期管理。在构件的连接依赖关系中,剖面、构件接口或其他接口和接插口的连接刻画了构件之间通过持有对方引用而建立起来的同步紧耦合方式的连接关系;事件槽和事件源的连接则刻画了构件之间通过事件通道建立起来的异步松耦合方式的连接关系。CCM构件模型的独特之处就在于构件提供服务和所需服务的对称接口定义^[5]。利用这种对称的接口定义并借助于对构件连接依赖关系的描述,可以实现构件的组装重用。

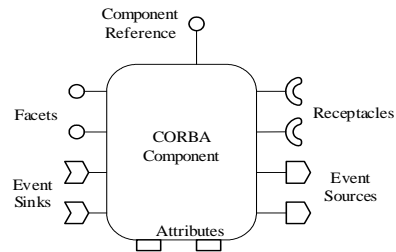


图 3 CCM 构件外观特征

CCM 具有很多的优越性。主要特点如下:

(1)复用性。对于任意实体,复用性依赖于对 2 个方面问题的清晰描述:该实体提供什么服务,需要什么服务。只要完备提供实体所需功能,并正确使用其提供的功能,实体就可以被复用。CCM 构件通过对称的接口定义,对所需服务及策略有清晰的描述,更易于被复用。

(2)接口定义的对称性。CCM 构件在接口层面上支持对提供服务和需求服务的定义。对称的接口定义,连同对构件间连接依赖关系的描述可以清晰地表达应用的软件体系结构。借助于体系结构信息,可以实施对应用的高级管理功能,如动态配置,即在运行时刻对系统进行修改甚至是扩展,从而满足技术的发展、客户需求的变化及系统对 QoS 的要求。

(3)对分布式应用部署的支持。对于一个由 CCM 构件组装成的分布式应用来说,构件良好的自包含性带来了丰富的自描述信息,从而可在构件组装部署模型的支持下实现应用的自动、分布式部署,解决基于对象的分布式应用无法方便、灵活地部署在分布式环境中的问题。

(4)松耦合性。CCM 构件之间的调用关系是通过对称的接口定义以及对连接依赖关系的声明,在部署组装时建立起来的。建立起来的连接关系也可以在不需要的时候撤销。基于松耦合性,CCM 构件可以更方便地支持一些高级的使用和管理功能,如容错、负载平衡和动态配置。

(5)软件开发的简单快速、高质量。构件开发者只需关注业务逻辑的实现,大量代码可以自动生成。而且,基于构件的应用开发过程被清晰地划分为 3 个阶段:构件开发,构件组装和构件部署。显然,基于构件的分布式应用开发过程大大简化了基于传统对象的应用开发过程,提高了软件开发效率,保障了软件质量。

4 DHMM 的设计

基于以上的分析,笔者认为 CCM 很适合构建该数字娱乐系统。其可有效地屏蔽由系统异构带来的复杂性。同时从系统的角度,提供事务、安全、通知等高级功能,为系统中

各节点通信方面的开发带来很大的便利。

如图 4 所示, 本文的数字家庭多媒体中间件基于 CCM 技术。其中, 实现了配置解析构件(configuration XML parser component, CxPC)、多媒体构件 (multimedia component, MMC), 控制解析构件(control parser component, CPC)。

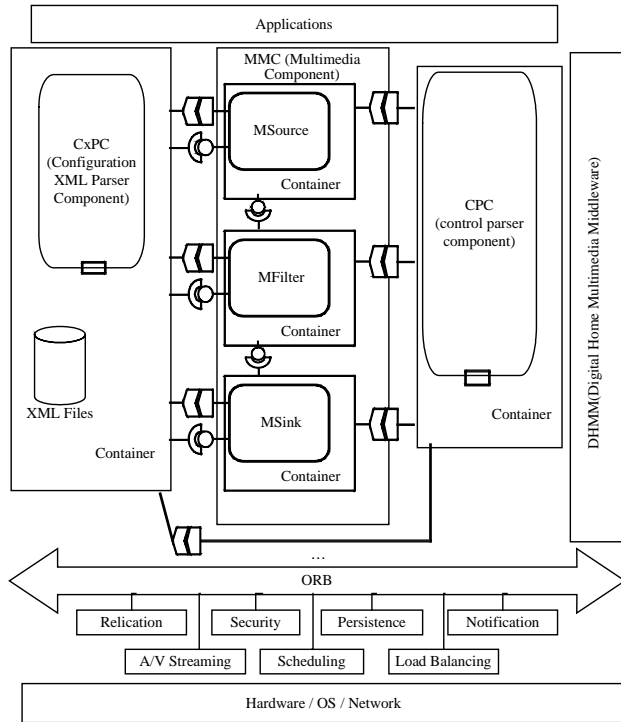


图 4 数字家庭多媒体中间件架构

4.1 多媒体构件

多媒体构件是数字家庭多媒体中间件中的关键构件, 其作用是处理多媒体应用相关的内容。本文的实验系统包括视频的捕获、视频编码/解码、多媒体码流传输、码流的回放与保存等功能。这些功能由硬件模块及驱动模块或其他软件模块提供, 但是各自的 API 都不相同, 这为系统的开发带来了很大的困难。在系统中, 使用 CCM 的接口设计技术 IDL 语言定义这些模块的接口。这使模块可以在不同系统中重用, 并可以在应用层被方便地调用。其中, 通过 A/V streaming 技术可以方便地将多媒体信息编码成码流, 在基于 CCM 的构件间传送。这为多媒体数据传输方面的开发带来了极大的便利。在目前的设计中包括 3 种功能不同的构件类型: Msource, Mfilter, MSink。

MSource 是处理多媒体数据输入的构件, 用于接收各种输入设备的原数据并加工成为其他构件能够处理的数据格式。MSource 提供了事件源与 CxCP 相连接。MSource 会向 CxCP 提供当前可用的多媒体信息源的信息, 同时提供多个不同的 CCM 剖面, 以此提供各种不同的功能。MSource 还会通过刻面向 MFilter 提供需要进一步处理的原数据(raw data)。具体的数据格式通过 IDL 文件中的剖面定义给出。总之, MSource 是对 DHMM 中多媒体数据源的封装。

MFilter 是多媒体数据的处理构件, 用于处理来自 MSource 的各种多媒体数据, 主要负责多媒体数据的编解码及转码的工作。在 MFilter 中也通过事件源同 CxCP 相连接, MFilter 会提供可用的编解码器的具体信息。另外还有一个刻面向 MSink 提供解码或编码后的数据。总之, MFilter 是 DHMM 中多媒体数据的加工厂。

MSink 是多媒体数据的输出构件。多媒体数据的输出一般有 2 个方向: 显示终端或其他设备的输入构件(MSource)。当面向显示终端的时候, MSink 一般是一个渲染构件。其功能是将 MFilter 解码后的数据转换成显示设备能够回放的内容。当面向其他设备的输入构件时, MSink 将是一个网络传输的代理。可以实现将 MFilter 编码后的码流通过 A/V streaming 技术传输到其他设备的 MSource。同时可以实现 QoS 等高级功能。综上所述, MSink 是 DHMM 中多媒体数据的消费节点。

4.2 配置解析构件

配置解析构件相当于数字家庭多媒体中间件中的数据库及管理系统。它会记录当前系统中的上下文信息, 包括所捕获的用户信息, 通过 XML 文件保存在各个设备的节点上。如果通过传统的开发方式, 收集及同步这些数据是一件难事。通过构件技术, 可以将这些复杂性屏蔽在中间件的特殊模块中。通过成熟的可重用的构件, 可以在应用层方便地获取这些信息。在 DHMM 中, CxPC 的目的就是为应用层提供智能的上下文信息。

目前在本文的实验系统中实现了记录当前系统状况的功能, 包括有哪些可用的输入设备, 编解码模块及输出模块。这些信息是通过 CxPC 接口, 动态地从 MMC 那里获得的。在 CxPC 上安装有事件槽, 可以接收 MMC 上事件源发来的消息。当有新的 MMC 加入或当前构件发生变化时, 都会有事件产生, 而 CxPC 会记录下这些信息。CxPC 还会包含用户的状态信息, 如收看电影的书签。这些信息会从 CPC 那里获得。当用户具体操作系统时, 这些信息会自动被保存下来。这样应用层就可以获得更为丰富的信息构建高级的功能。

4.3 控制解析构件

控制解析构件是数字家庭多媒体中间件中控制信息的输入代理。在数字家庭中, 控制信息可能通过各种途径为 DHMM 获得。换言之, DHMM 需要支持各种输入设备, 这包括未来可能出现的外设。就本文的实验系统而言, 输入设备包括键盘、鼠标、红外遥控器及远端的控制程序等。对于系统而言, 这可能是一个 Xevent 或一次中断或是一个 UDP 的数据报。如果需要应用程序处理这些复杂且多变的输入源, 对于应用层的开发人员, 这是一件痛苦而乏味的事情。同样, 如果有新的输入外设加入, 对原有应用程序的修改将不可避免。因此, 需要在中间件中加入 CPC 这样的构件, 将输入与应用开发隔离。CPC 的作用就是捕获各种外设的输入数据, 将其转换成统一的事件。这样如果有新的外设加入, 只需要安装一个新的 CPC 到系统中, 系统就可以使用新的设备了。

在 CPC 中, 需要实现一个事件源, 它会与 CxPC 上的事件槽相连接。通过事件相应机制, CPC 可以动态地加入或移出系统。在加入的同时获取系统可支持的事件类型, 实现部分的智能消息映射。CPC 还可以通过 CxPC 的帮助与 MMC 中的构件建立事件响应机制。从而实现一些自动的功能。比如当 CxPC 中记录有自动录像的需求, CPC 可以自动完成这个功能(需要应用程序的设置)。

5 数字家庭娱乐中心

在作者参与的《无线数字电视平台及其核心软件系统》项目中, 分别在多媒体中心服务器(Linux)与家庭电脑(Windows)上运用了 CCM 技术, 并按照 DHMM 的设计实现了若干主要的构件, 搭建出 DHMM 的实验系统。通过实验, 证明了 DHMM 的设计在普适计算环境中的可行性。

实验环境与第2节中描述的场景类似。具有一台安装有Linux系统的PC作为多媒体中心服务器。另外还有一台安装有Windows系统的PC作为家庭个人电脑。实验用的LCD电视机与基于Sigma Design公司EM8621L芯片的机顶盒相连接。这些设备都通过802.11g无线网络相连接。在实验系统中实现了如下功能：

(1)将模拟电视节目通过多媒体中心服务器转码成为MPEG2的码流，并在机顶盒内回放。

(2)以MPEG2文件的格式，将模拟电视节目保存到多媒体中心服务器上。

(3)通过机顶盒，点播多媒体中心上的多媒体文件，包括各种格式的音频、视频及图片文件。

(4)通过家庭个人电脑将各种视频文件转码成为MPEG2的码流(软件)，并在机顶盒内回放。

在设计中按照IDL规范定义了以下主要的接口：

```
interface registerComponent {
//构件的注册接口
void register ();           //将构件注册到 CxPC 中
void update();             //更新 CxPC 中的信息
void destroy();            //注销 CxPC 中的信息
}
```

以下是实验系统中主要的构件定义：

```
component MSource_TVtuner
supports registerComponent {}; //模拟电视输入构件
component MFilter_MPEG2
supports registerComponent {}; //MPEG2 硬件编码构件
component MSink_DISPLAY
supports registerComponent {}; //显示构件
component MSink_Trans
supports registerComponent {}; //MPEG2 软件编码构件
component MCPC_RC
supports registerComponent {}; //遥控器事件映射构件
以下是构件宿主的定义：
home MSourceHome manages MSource{};
home MFilterHome manages MFilter{};
home MSinkHome manages MSink {};
home CxPCHome manages CxPC{};
home CPCHome manages CPC{};
```

(上接第194页)

表2 MDE, FPDCGA 与 PSONE 的比较结果

算法	min(f(x))	min(f(x))平均值	方差
MDE	-0.803 619 104 060 374 07	-0.803 619 093 396 667 26	0.000 201 073
PSONE	-0.803 619 000 000 000 00	-0.803 604 000 000 000 00	0.000 200 000
FPDCGA	-0.802 747 000 000 000 00	-0.796 500 000 000 000 00	0.005 600 000

4 结束语

利用EA求解复杂的高维约束优化问题时，对约束条件的处理是关键。本文基于函数 e^x 的一般性质，提出了一种不需要任何因子的简易罚函数法SPFM，避免了在罚函数法中因确定罚因子而带来的困难。鉴于DE是一种全局搜索能力极强的算法，将SPFM和DE相结合，提出了一种求解难约束优化问题的改进差分演化算法MDE。对Bump问题的求解结果表明，利用MDE求解约束优化问题不但编码简单，而且运算速度和计算结果更优。

在实现中使用了Vanderbilt 大学和华盛顿大学共同开发的CIAO (Component Integrated ACE ORB)^[6]作为CCM的实验平台。这个系统为CCM构件技术在DRE(Distributed Real-time & Embedded)应用的开发做了大量的优化。通过实验，取得了比较好的效果。

6 结束语

本文分析了数字家庭中多媒体娱乐应用程序开发所面临的问题。其主要表现为环境的异构性与系统的异步性所带来的设计复杂、实现困难等问题。尝试通过构件技术，构建适合家庭环境的多媒体中间件，为上层应用的开发提供良好的环境。研究并设计了称为DHMM的中间件架构，并在《无线数字电视平台及其核心软件系统》项目中测试了架构的可行性。由于目前的试验系统还不完善，因此只实现了部分功能。但是就现有的结果来看，DHMM具有很好的抽象能力及扩展性。使用DHMM可以很好地减少应用程序的开发难度，降低开发中模块间的耦合度，为应用程序的开发及部署带来了很大的便利。

CCM技术本身具有很好的开放性，世界上有很多组织正尝试让CCM技术在DRE环境中更好地工作。CIAO是一个很好的尝试，并已经取得了良好的效果。不过这方面的研究还只是处在起步阶段，以后将进一步研究与此相关的问题，并将其应用到中间件架构的设计中。

参考文献

- [1] Merle P. CORBA 3.0 New Components Chapters[Z]. (2001-11-03). <http://www.omg.org/cgi-bin/doc?ptc/01-11-03.pdf>.
- [2] Microsoft. Microsoft Component Object Model Protocol COM 1.0 Specification[Z]. 1996-05.
- [3] Sun Microsystems. JavaTM 2 Platform Enterprise Java Beans Specification. V2. 1: Final Draft 2/8/2002[Z]. 2002-02.
- [4] OMG. OMG CORBA Components, Integrating with Enterprise Java Beans[Z]. (2003-08-07). <http://www.omg.org/corba>.
- [5] Bálek D. Connections in Software Architecture[D]. Physics Department of Software Engineering, Charles University, 2002.
- [6] Vanderbilt. CIAO Overview[Z]. (2005-07-06). <http://www.cs.wustl.edu/~schmidt/CIAO-intro.html>.

参考文献

- [1] Michalewicz Z. 演化程序——遗传算法和数据编码的结合[M]. 周家驹, 何险峰, 译. 北京: 科学出版社, 2000.
- [2] 贺毅朝, 王彦祺, 刘建芹. 一种适于求解离散问题的二进制粒子群优化算法[J]. 计算机应用与软件, 2007, 24(1):157-159.
- [3] Storn R, Price K. Differential Evolution for Multi-objective Optimization[J]. Evolutionary Computation, 2003, 11(4): 8-12.
- [4] 贺毅朝, 寇应展, 陈致明. 求解 TSP 问题的位置-次序编码差分演化算法[J]. 计算机应用, 2007, 27(3): 630-632, 641.
- [5] 李敏强, 寇纪松, 林丹, 等. 遗传算法的基本理论与应用[M]. 北京: 科学出版社, 2003.
- [6] 李炳宇, 萧蕴诗, 吴启迪. 一种基于粒子群算法求解约束优化问题的混合算法[J]. 控制与决策, 2004, 19(7): 804-807, 812.