

多媒体统一处理平台产品化定制的设计与实现^①

张桂周 叶明 (南京航空航天大学 信息科学与技术学院 江苏 南京 210016)

摘要: 给出了基于网络多媒体开放系统的多媒体统一处理平台的产品化定制设计需求以及软件实现方法,介绍了该定制方法承担的任务,重点介绍了如何定制所需要的各类产品。该方法设计合理,使用方便,具有很强的实用性,对于计算机行业中其他领域产品的定制也具有指导作用。

关键词: 网络多媒体开放系统; 多媒体统一处理平台; 数据管理服务器; 媒体交换服务器; 视频管理服务器

Design and Implementation of Customized Multimedia Processing Platform Product

ZHANG Gui-Zhou, YE Ming

(College of Information Science & Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract: This document describes the customized design and software implementation method of the multimedia processing platform based on IP MultiMedia open system. It introduces the tasks undertaken with the customized design method, focusing on how to customize products as required. The method has features like excellent design, being easy-to-use, and high practicability. It also provides guidance to the product customization in other fields of the computer industry.

Keywords: IP multiMedia open system; multimedia processing platform; VM(Video Management Server); DM(Data Management Server); MS(Media Server)

随着计算机网络的飞速发展,现代社会越来越多的通过网络来实现互连,达到交互信息与资源共享的目的。由于对视频监控方面的需求越来越多,因此基于IP的视频监控产品越来越受市场青睐。

1 引言

多媒体统一处理平台是基于网络多媒体开放系统的多媒体解决方案。网络多媒体开放系统是IP领域的多媒体开发平台。该平台组件间使用采用SIP、RTSP/RTP、SNMP等标准协议,符合相关的国际、国内标准,保证业务、功能、界面、内容的高度统一化和标准化;提供多层次的SDK接口;在“多媒体基础模型”的基础上更进一步,提供“多媒体中间件”的功能。它将通过抽象具有相似特点的功能来建立各种平台组件框架,通过提供“脚本规则”或“插件规范”或接口“实现”的替换功能给用户以实现用户对组件模型的充分定制化;该系统是SOA的开放框架,

实现平台所有功能全方位的开放,实现“开发即开放”。多媒体统一处理平台在网络多媒体开放系统的基础上面对它提供的各个模块进行定制,从而定制出不同款的产品满足市场需求。

2 产品化定制概述

2.1 项目介绍

多媒体统一处理平台产品化定制承担其相关产品的安装、状态查询、卸载及配置信息的获取;目前网络多媒体开放系统平台提供功能模块,多媒体统一处理平台根据需求定制网络多媒体开放系统平台中的相关功能模块,最终定制出向用户提供的产品,本文中介绍的产品主要有VM、DM、MS,当然它也可以定制EC、DC编解码器以及其他基于IP的多媒体产品。各个产品需要安装、升级、状态查询、卸载及配置信息的获取,为实现这些功能,所以启动产品化定制项目。

^① 收稿时间:2009-08-17;收到修改稿时间:2009-09-25

2.2 设计方案介绍

产品化定制方案如下:

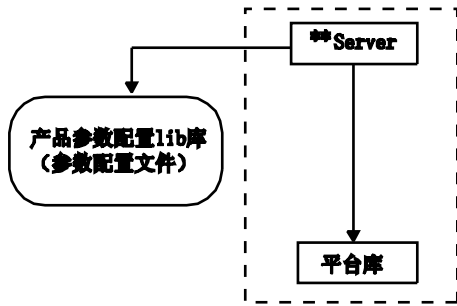


图1 产品化定制方案总图

说明: 椭圆形部分由产品化定制时实现; 长方形部分(虚框内)由平台提供。

操作过程:

- (1) 平台 `**server` 调用产品参数配置 lib 库, 获取到平台所需要的参数;
- (2) 平台 `**server` 调用网络多媒体开放系统平台库, 从产品参数配置 lib 库中获取到的参数传送给平台库, 完成整个服务的启动。

产品化定制主要任务:

- (1) 产品参数配置 lib 库的编写;
- (2) 产品化各产品安装升级启动、状态查询及卸载脚本的编写;
- (3) 守护进程;
- (4) 产品定制于 CentOS 操作系统中。

3 产品参数配置lib库的设计



图2 M 模块接口的 lib 库实现

根据平台提供接口设计产品参数配置 lib 库。

该平台提供的组件有多个模块。各个模块分别实现不同的功能, 在此不作描述。产品参数配置 lib 库要从配置文件中获取配置参数值, 然后将参数值传给这几个组件相关接口, 从而完成其功能。

由于 M 模块较之于其他模块来说又可细分, 所以实现方式略有不同, M 模块接口的实现结构图如图 2 所示。

对其他模块而言实现方式是一样的, 现在以 C 模块为例给出实现结构图如图 3 所示。



图3 C 模块接口的 lib 库实现

4 产品化定制软件设计

网络多媒体开放系统既可以选择 Linux、Post-GreSQL 等开源项目做为解决方案的运行平台, 显著降低运行成本, 也可以选择 Windows、Oracle 等商业系统满足超大规模应用的需求。

本文中涉及的产品化定制的实现是基于 Linux 系统的, 采用 shell 编程来实现, 通过脚本来完成产品的安装、升级、卸载、服务以及配置工具的编写。定制的 VM, DM, MS 均采用该方式来实现, 这几款产品的脚本程序实现大致相同, 只是它们定制的模块不同, 启动的进程也不同^[1,2]。

VM 是视频管理服务器, 它定制网络多媒体开放系统平台提供的五个模块, 并且 VM 产品还需要定制数据库, 以实现其所要求实现的功能。

DM 是数据管理服务器, 它定制网络多媒体开放系统平台提供的 M 模块中的一部分, 通过定制来实现数据管理的功能。

MS 是媒体交换服务器, 也是定制 M 模块中的一部分, 但与 DM 定制的部分不同。

由于各款产品的定制相似, 因此下面以 VM 为例来介绍下实现过程。

4.1 产品的安装

该脚本实现对 VM 的安装, 安装成功之后, 机器上面就能启动相应的服务, 实现相应的功能。主程序实现结构图如图 4 所示:

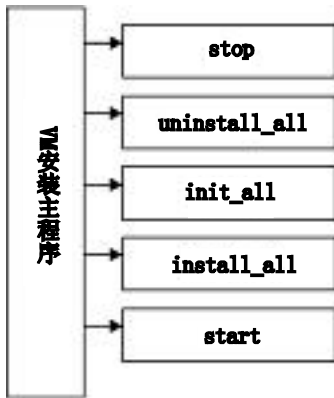


图 4 VM 安装脚本主程序调用函数

VM 安装主程序调用五个函数的顺序依次是 stop, uninstall_all, init_all, install_all, start。假设机器中已经安装了 VM, 那么在执行安装之前, 首先调用第一个函数 stop 用于停止正在运行的进程, 然后调用 uninstall_all 来卸载机器中已安装的产品。接收终端的参数输入, 然后调用 init_all, 完成安装之前的初始化工作。接着调用 install_all 执行产品的安装, 安装好之后调用 start 来启动所需的各个进程。

如果没有安装, 则直接接收终端的参数输入, 然后调用 init_all, install_all, start。

4.2 产品的升级

该脚本实现产品从老的版本升级到新版本。主程序实现结构图如图 5 所示:

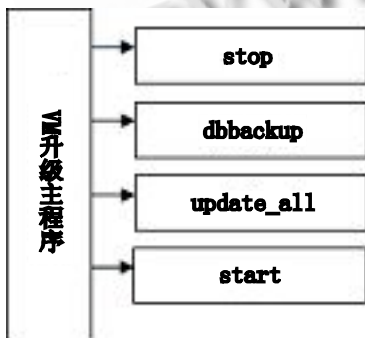


图 5 VM 升级脚本主程序调用函数

VM 升级主程序调用四个函数的顺序依次是

stop, dbbackup, update_all, start。假设机器中已经安装了 VM, 那么在执行升级之前, 首先判断是否安装了该产品, 如果没有安装则直接退出升级。如果安装了, 则判断版本号, 升级版本较低则直接退出升级; 升级的版本号比较高, 则执行下面的操作。首先调用第一个函数 stop 用于停止正在运行的进程, 然后调用 dbbackup 来备份配置数据。接下来调用 init_all, 完成安装之前的初始化工作。接着调用 update_all 执行产品的升级。升级好之后调用 start 来启动产品的各个进程。

4.3 产品的卸载

该脚本实现产品的卸载。主程序实现结构图如图 6:

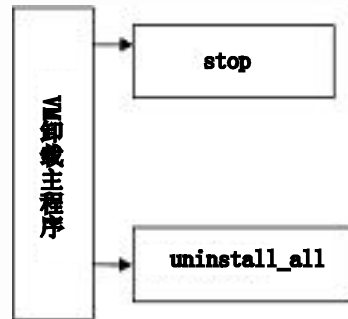


图 6 VM 卸载脚本主程序调用函数

VM 卸载主程序调用两个函数的顺序依次是 stop, uninstall_all。首先该脚本会调用函数 stop 停掉已启动的服务进程, 然后, 调用 uninstall_all 执行产品的卸载。

4.4 产品的服务脚本

主程序实现结构图如图 7:

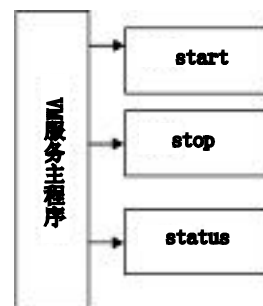


图 7 VM 服务脚本主程序调用函数

服务脚本主程序实现功能是, 输入启动标识 start

直接调用 **start** 实现进程的启动；输入停止标识 **stop** 调用 **stop** 实现进程的停止；输入重新启动标识 **restart** 先调用 **stop**，再调用 **start** 实现进程的重新启动；输入查询标识 **status** 调用 **status** 实现进程的查询。

4.5 配置工具

该脚本可以帮助用户设置配置参数，也可以查询已设置的配置参数，并打印出来。该脚本还能够查询已安装产品的版本号，帮助用户确认自己安装的版本^[3]。

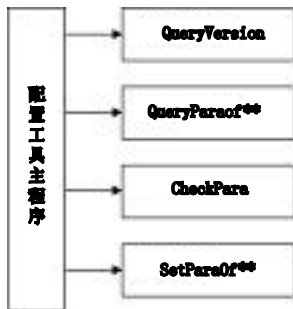


图 8 配置工具主程序调用函数

其中，**QueryVersion** 用来查询产品的版本号；**QueryParaof**** 用来查询产品的配置参数，**代表不同的产品，如果是 VM，则该函数为 **Query-Paraofvm**；**CheckPara** 用来验证输入的参数值是否合法；**SetParaOf**** 则用来设置配置参数的值。

4.6 守护进程

守护进程的主要功能是实时监控产品的各个进程，它从守护进程配置文件中读取相应的产品参数，然后检测进程状态，当产品进程被停掉时，守护进程会将其拉起。

4.7 应用举例

由于产品化定制实现的过程相似，而其中以 VM 为核心，因此以 VM 来介绍该文定制的产品。经过了

上面的定制之后，我们就可以在网络多媒体开放系统平台的基础上定制出视频监控系统中处于核心地位的产品 VM 了。VM 是整个系统整体解决方案的核心产品，是用于管理视频监控业务和控制呼叫的中心服务器。其实现的功能主要有两大类：其一，进行呼叫控制和信令转发；其二，用于进行用户管理和设备管理，比如添加删除用户和设备，实现各项控制功能，同步系统数据等等。执行安装包中的安装脚本之后，便可将 VM 安装到服务器上面，完成上述的各项功能。

5 结语

经过几个月的学习与实际项目的参与，对产品化定制整个流程有了一个比较深入的认识，在整个项目的进展期间，对 Linux 系统的 shell 程序的编写有了比较深入的理解，也基本了解了平台的整个框架体系，并在这个平台的基础上，定制出了三款不同的产品。该产品化定制能够在已有平台各个模块存在的基础上，根据用户实际需要来推出不同的产品，此设计的意义不仅仅是这几款产品的定制，而是提供了一种行业解决方案，基于 IP 的多媒体产品如何能够比较简单的推出，推广开来，其他计算机相关产品也可以根据该方法来定制。

参考文献

- 1 Tansley D. 张春萌译. LINUX 与 UNIX shell 编程指南. 北京:机械工业出版社, 2002. 89-100.
- 2 汤荷美, 董渊. Linux 基础教程(1)操作系统基础. 北京:清华大学出版社, 2005. 141-153.
- 3 索贝尔, 杨明军, 王凤芹译. Linux 命令、编辑器与 Shell 编程. 北京:清华大学出版社, 2007. 93-115.