

Pocket PC 数据库开发技术及其在 建材零售点业务中的应用

乔平安, 杨立法

(西安邮电学院 计算机系, 陕西 西安 710121)

摘要: Pocket PC 是一种新的移动计算平台, 基于该平台的数据库应用程序的开发具有很大的实用价值. 论文结合自身实践, 介绍了 Pocket PC 开发环境的搭建方法、Pocket PC 数据库特点, 重点探讨了 eVC 中 Pocket Access 数据库应用开发过程, 并以建材行业零售点业务管理系统的设计为例, 给出 Pocket PC 数据库应用开发的一些典型结果.

关键词: Pocket PC; Windows CE; ADOCE; 数据库应用; 建材市场

中图分类号: TP311.52 **文献标识码:** A **文章编号:** 1007-855X(2007)03-0035-05

Database Development Technology of Pocket PC and Its Application in Retailing Business of Building Materials

QIAO Ping-an, YANG Li-fa

(Faculty of Computer, Xi'an University of Post and Telecommunication, Xi'an 710121, China)

Abstract Pocket PC is a new kind of computation platform based on which the development of database application program is of practical significance. In combination with the practice, the establishment method of Pocket PC development environment and the features of database of Pocket PC are introduced, with the emphasis put on the development process of Pocket PC database application in eVC. In addition, some typical results of Pocket PC database application development are given, with the example of retailing business management of building materials.

Key words Pocket PC; Windows CE; ADOCE; database application; building materials market

0 引言

Pocket PC 是以 Windows CE (简称 Win CE) 为操作系统的掌上电脑, 体积小, 重量轻, 携带方便. 以 Pocket PC 为平台的数据库, 具有“移动数据, 随身携带”^[1~5]的特点. Win CE 的界面与桌面 Windows 界面极为相似, 但 Win CE 并不是桌面 Windows 的一部分或缩减版, 同时由于 Pocket PC 属于移动设备, 具有 PC 机所不具备的特点, 如显示屏小、内存有限等, 这就使得两个平台的程序开发有所不同. 基于对 Pocket PC 数据库开发技术进行探讨, 并以建材行业进销存软件设计为例, 阐述了 Pocket PC 数据库应用开发过程.

1 Pocket PC 应用系统开发环境

1.1 Windows CE 及应用开发环境^[5,7]

Win CE 是微软公司开发的一种模块化、实时、抢先式、多任务嵌入式操作系统, 是专为信息设备、移动应用、消费类电子产品、嵌入式应用等非 PC 领域而设计的操作系统, 它面向只有有限资源的硬件系统, 具有紧凑高效可伸缩的特性. Win CE 的模块化设计方式使得系统开发人员和应用开发人员能够为多种多样的产品来定制 Win CE, 适应于各种嵌入式系统和产品.

收稿日期: 2007-01-30

第一作者简介: 乔平安 (1966-), 男, 讲师. 主要研究方向: 数据库及其应用、计算机技术及应用.

E-mail: paqian@tom.com

配置和调试 Windows CE 及其应用程序需要一台高性能开发工作站和一台目标调试设备(如 Pocket PC)。Platform Builder 提供了创建和调试 Windows CE 的集成开发环境,如向导和工具栏,支持活动模板库(ATL)、微软基本类库(MFC)和 Visual Basic,为支持的处理器家族(如 x86 Xscale, ARM, MIPS 等)提供编译器、内核调试器以及各种远程调试工具。Platform Builder 3.0 要求运行在 Windows 2000 或者以上版本的高性能开发工作站上。目标调试平台可以是经过适当配置以后的普通 PC 机,也可以是嵌入式微处理器模块。开发工作站通过以太网或者串、并行口与目标平台连接,并将 Windows CE 操作系统及其应用程序下载到目标平台。下载完成之后,Windows CE 在目标平台启动。在开发工作站上,可以使用 Platform Builder 提供的调试工具查看 Windows CE 的运行情况。

Win CE 程序开发语言主要是 eVB (Embedded Visual Basic), eVC (Embedded Visual C++), 它们是 VB、VC 的子集。程序开发工具 Embedded Visual Tools (简记为 eVT, 包括 eVB, eVC) 由微软免费提供, 免费下载。和 VC 与 VB 的特点一样, 用 eVC 开发的应用程序具有运行速度快、程序占用空间小等特点, 但其开发周期长, 开发难度较大; 用 eVB 开发相对简单, 但在组件开发、网络通信等方面存在缺陷。论文主要讨论基于 eVC 的应用开发。

1.2 eVC 集成开发环境

eVC 和 VC 虽然在界面、语法和开发方式上十分相似, 但由于和桌面 Windows 系统之间不同的设计理念和硬件上的差别, 在编码和各种控件的使用方法上有很大不同。与传统桌面开发工具 VC++ 相比, eVC 具有嵌入式开发的特点, 如表 1 所示。

eVC 集成开发环境的界面除了一组特定工具外, 与 VC 集成开发环境相似。其中“WCE configuration”工具栏中可以选择设备类型 (Pocket PC, PPC, HPC)、仿真器类型 (Pocket PC 仿真器、PPC 仿真器、HPC 仿真器) 和 CPU 类型。工具菜单中多出了一些远程工具, 可以用来访问仿真平台, 比如添加一个文件到模拟器, 从模拟器复制一个文件到本机硬盘等。

ATL 主要用于基于 COM 组件技术的 OLE 和 ActiveX 支持, eVC 中的大多数 ATL 类与 VC 的 ATL 类相同, 但有 13 个不同或不支持。另外 eVC 中的 MFC 可以看作 VC 中 MFC 的一个子集, 但 eVC 的 MFC 中有 9 个类是独有的, 比如 CCeDBDatabase, CCeDBRecord, CCeDBProp, CCeDBEnum 这 4 个类支持 Win CE 的数据库功能, 但在 VC 的 MFC 中是不存在的, 对于开发者来说, 从开发资料、帮助文档就可以了解开发支持库的变化。

1.3 Pocket PC 开发平台的搭建

1) 安装 Pocket PC 开发环境。对于 Pocket PC 应用系统的开发, 目标平台可以是实际的 Pocket PC 设备, 也可以是微软 Platform Builder 自带的模拟环境 (运行在开发工作站上的 Pocket PC 仿真软件)。所以, 开发是基于这种模拟环境进行的。

假定开发工作站上的操作系统为 Windows 2000 应严格按照以下顺序建立 Pocket PC 开发环境: 安装 eVC4 及其 Service Pack 4 注意不要改变安装的相对路径, 否则开发时 eVC 找不到所需文件; 安装 Microsoft Pocket PC 2003 SDK (Pocket PC 的开发包); 安装模拟器。Pocket PC 的开发包中包含了 Pocket PC 仿真器, 但只支持英文。若要支持中文, 可直接安装 Emulator Images for Pocket PC - CHS.msi (中文仿真器)。

在整个安装过程中, 不要改动 PB (Platform Builder) 和 eVC 的安装路径, 当安装向导询问选择 CPU 时, 只选择 x86 和 emulator 即可。

表 1 eVC 4.0 与 VC++ 6.0 比较

Tab 1 Comparison between eVC 4.0 and VC++ 6.0

Visual C++ 6.0	eVC 3.0
Win32 API	Win32 API 的子集
C 运行时库	标准 C 运行时库
C/C++ 编译器	● 为 (X86) 模拟器提供的 X86/C++ C 编译器
通用开发工具	● 为 ARM 设备提供的 LARM C/C++ 编译器 ● 为 MIPS 和 MIPSFP 设备提供的 LARM C/C++ 编译器 ● 为 PowerPC 设备提供的 CLPP C/C++ 编译器
支持活动模板库	● 为 SuperH 设备提供的 SHCL C/C++ 编译器
支持微软基础类	使用和 Windows 环境类似的开发工具, 还附加了一些为
支持标准模板库	Windows CE 设备提供的远程开发调试工具, 并具有能生成 Windows CE 格式资源的资源编辑器, 能生成 4 色位图等 Windows CE 格式的资源。 Windows CE MFC 是标准 MFC 类库的子集, 支持大多数 ATL 类, 但不支持 STL。

2)配置虚拟网卡(微软自带的虚拟网卡)。进入控制面板,安装新硬件,安装 Microsoft 虚拟网卡;在网络连接中,找到对应的虚拟连接,在属性对话框 TCP/IP 的属性页,配置其 IP 地址,要与原网卡(或 eVC 中)的 IP 地址和网关一致。

注意,在配置完虚拟网卡后,一定要关闭掉防火墙,否则软件无法下载到模拟器。

3)中文显示.进行以下操作。打开 eVC4 选择 tools- > configure platform manager- > pocket pc 2003 emulator;会出现: Transport TCP/IP Transport for Windows CE Configure.. (点击此处显示模拟器需要配置的 IP 地址) Startup: Emulator Startup Server Configure.. (点击此处执行 2); 完成以上操作后,在 image 栏中选择 dsppc 2003 se 在 eVC4 工具栏中,在 project- > Setting- > Resources- > language 中选择: 中文 [中国]。

2 eVC 中数据库应用开发

eVC 中开发数据库应用有两种方法,一是使用 SQL Server CE,二是使用 eVC 自带的 Pocket Access

2.1 SQL Server CE

SQL Server CE (SQL Server 2000 for Windows CE 的简称)是 Pocket PC 上推荐使用的嵌入式数据库,使用 .sdf 文件.它与 SQL Server 2000 桌面版之间可以实现良好的数据同步及互操作,使用一些诸如 Replication (复制)、RDA (Remote Data Access, 远程数据访问)和执行 SQL 查询处理工作的特性。

SQL Server CE 在安装时要选上 Development Tools (主要完成 RDA 和 Replication 功能)及 Server Tools (主要完成 IIS 支持与 SQL Server 2000 的连接功能)。同时安装 ADOCE 3.1,通过它对数据进行操纵.在远程数据存取时,通过 PUSH 将本地的数据上传到服务器上,而用 PULL 将服务器数据下载到本地.需要注意的是, PUSH 方式必须保证本地表和远程数据表的结构一致 (PUSH 方式需要一个数据开始点,即保持本地表和远程数据表的结构一致,这需要在联网的条件下通过 PULL 方式实现),而 PULL 方式必须是本地没有同样名字的表。

2.2 Pocket Access

Win CE 自带 Pocket Access 数据库,使用 .cdb 文件,但没有实际的 Pocket Access 软件. Win CE 自带的数据库类有: CMountDBVol() (装配数据库卷), CCreateDatabase() (创建数据库), COpenDatabase() (打开数据库), CSeekDatabase() (查找或移动记录)。

PC 上的 Access 数据库格式为 .mdb 与 Pocket PC 上使用的 .cdb 不兼容,在目标设备上可以通过 DesktopToDevice() 函数和 DeviceToDesktop() 函数来完成转换,但在仿真器上不能使用以上函数,在此可以对 VOADO 类的代码进行修改来实现。

VOADO 类需要在构造函数中提供数据库的路径名,但此时还没有建立数据,可以在 Connection 类中添加 void SetConnectDB(LPCTSTR pszDBName) 和 void ClearConnect() 两个函数,作用是设置连接的数据源以及断开连接,具体实现如下:

```
void CConnection::SetConnectDB(LPCTSTR pszDBName)
{
    if(m_Conn)
    {
        m_Conn->close();
        m_Conn->put_Provider((LPTSTR)(TEXT("cedb")));
        m_Conn->Open((LPTSTR) pszDBName, TEXT(""), adOpenUnspecified);
        m_stdProvider= pszDBName; } }

void CConnection::ClearConnect()
{
    if(m_Conn)
    {
        m_Conn->Close();
        m_Conn->put_provider((LPTSTR)(TEXT("cedb")));
        m_Conn->Open(TEXT(""), TEXT(""), TEXT(""), adOpenUnspecified);
        m_stdProvider= TEXT(""); } }
```

这样就可以解决在没有任何数据源的情况下建立数据库.创建表及向表中插入初始数据需通过 VOADO 进行,但存储数据的文件选用了 XML 格式.选择 First Objective Software 的 Cmarkup 类可进行 XML 解析。

2.3 ADOCE 访问数据库^[56]

ADOCE (ActiveX Data Objects for the Windows CE) 是桌面 ADO 的子集,该子集专门用于 Win CE. ADOCE 包括自己内部的数据库提供者,可以访问本地 Win CE 设备的数据库,可以使用 OLE DB 支持的数据源. ADOCE 的数据访问模型如图 1 所示。

ADOCE 提供了以下对象: Connection 对象, 用于创建和存取数据库和数据表, ADOCE 可连接到 Windows CE 数据源和 OLE DB Provider 数据源; Recordset 对象, 用于表示从实际数据库、数据表组合而成的虚拟记录集, 其中有许多的方法和属性, 主要用于管理已经获得的记录集合; Field 对象, 用于表示字段; Errors 对象, 包含各种错误信息。

需要注意, ADOCE 不支持 ADO 的 Command 对象、Property 对象和 Property 集合, 所有的命令操作都包含在 Open 方法中。

使用 ADOCE 访问数据库方法如下 (以 PocketAccess 为例):

1) 建立与数据源的连接. ADOCE 采用 Connection 对象的 Open 方法实现数据源的连接, 语法为:

```
Connection Open (BSTR ConnectionString BSTR UserID, BSTR Password, long Options)
```

如: My_con= Conn1->put_Provider(TEXT("cedb"));

```
My_con= Conn1->Open(TEXT("M ydb cdb"), TEXT(""), TEXT(""), adOpenUnspecified);
```

2) 创建 Recordset 对象:

```
My_con= CLS DF romProg ID ( tsADOCE30RSProgID, & tC lsid);
```

```
My_con= CoCreateInstance( tC lsid NULL, CLSCTX_NPROC_SERVER, IID_Recordset ( LPVOID* )&Rs1);
```

```
varConn1. pd ispVal= Conn1; varConn1. vt= VT_D ISPATCH;
```

```
My_con= Rs1->put_ActiveConnection(V aiConn);
```

3) SQL 命令的执行和数据的处理

ADOCE 支持 SQL 的子集, 使用 SQL 命令可以简单地实现对数据库、数据表、记录的建立、查询和修改. ADOCE 支持的 SQL 命令与 ADO 支持的 SQL 命令基本一致^[3].

ADOCE 不支持命令对象 (Command), 所有的 SQL 命令操作都包括在 Recordset 对象的 Open 方法中, 语法为:

```
Open(VARIANT Source, VARIANT ActiveConnection, CursorTypeEnum CursorType, LockTypeEnum LockType, LONG Options) (其中第一个参数可以是 SQL 命令).
```

3 建材行业零售点进销存管理软件设计与实现

3.1 需求分析

目前, 在建材行业零售点业务中, 主要存在以下问题:

1) 从订货到实际销售, 期间需要经过很多流程, 其中要填制大量的表单. 在填制这些表单时, 需重复录入很多的信息. 例如, 建材名称、生产厂家、规格等在订货、入库、销售、库存及出库记录中反复出现, 每次重复录入, 降低了工作效率.

2) 建材品种繁多、价格多变, 难以准确地完成销售开票任务, 因而造成错账、漏账.

3) 每天的销售情况, 不能及时反映, 零售商不能实时有效地了解经营情况和出现的问题.

4) 在库存管理方面, 由于库存品种繁多, 手工统计耗时、耗力, 有时还出现库存统计不准确的现象, 这便出现了库存建材积压、缺货商品不能及时补进.

根据以上需求分析, 建材行业零售点进销存管理软件应该具备的主要功能有:

1) 对客户和建材商品的基础信息进行维护管理. 能够完成信息的添加、删除和修改, 并按一定条件查询有关信息.

2) 对商品的销售进行管理. 实现产品销售, 销售记录, 查询, 修改, 删除, 打印, 退货, 销售统计, 出库警戒等.

3) 库存商品查询. 可按不同字段和条件查询库存商品信息.

4) 查询统计功能. 既可按日期查询, 也可以按用户选择的字段和输入的内容查询, 并在查询过程

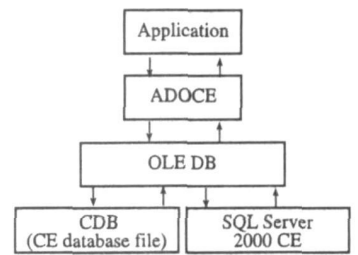


图1 ADOCE 的数据访问模型
Fig.1 Data access model of ADOCE

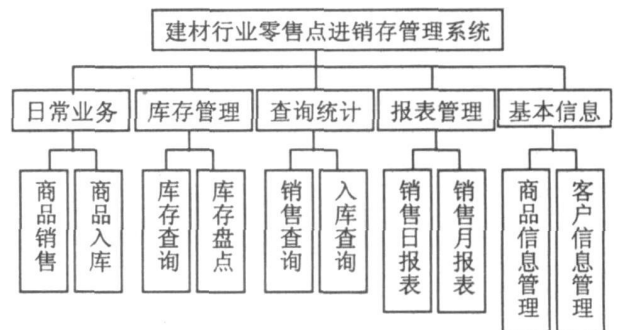


图2 系统功能结构图
Fig.2 Application funtion architecture

中统计销售数量和销售金额.

系统的功能结构如图 2 所示.

3.2 数据库设计

根据系统主要功能要求, 数据库的逻辑结构设计如下:

- 1) User 登录信息表 (用户代码, 登录密码);
- 2) Customer 客户资料表 (单位, 客户类型, 客户地址, 邮政编码, 联系电话, 联系人);
- 3) GoodsInfo 产品信息表 (供应商名, 产品名, 入库吨位, 入库价格, 入库发票号, 操作人, 产品生产地, 备注信息);
- 4) Store 库存信息表 (产品名, 库存量, 销售价格);
- 5) Sell 销售记录表 (客户名, 产品名, 销售发票号, 操作人, 吨位, 日期).

本设计采用 Pocket PC 的本地数据库 Pocket Access

3.3 软件设计

根据建材行业的实际特点, 该应用软件利用 eVC 提供的强大的可视化功能, 为用户创建窗口式程序界面, 使用中通过窗体的相互调用, 完成用户建材的进、销、存的基本要求.

1) 界面设计

设计 Pocket PC 应用程序用户界面的指导思想是简单易用, 还要考虑如何显示数据和排放控件, 应避免在应用程序主窗口内使用大量的控件, 并尽可能留出足够的空间来显示数据. 图 3 示出了以控件形式列出的系统功能.



图3 系统功能
Fig.3 System function layout

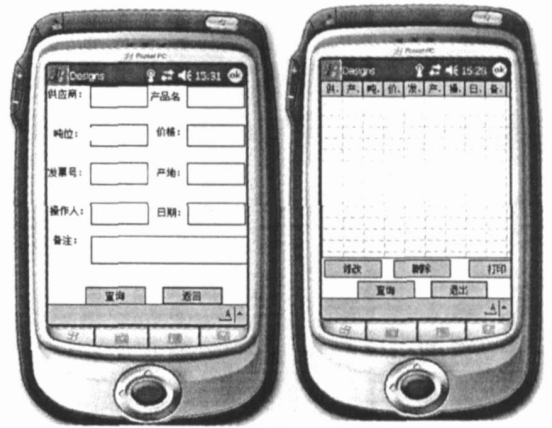


图4 库存管理
Fig.4 Stock management

2) 功能设计. 以入库产品数据库管理为例如图 4 若要对该数据项进行操作, 首先需查询出所要操作的项目, 左图是一个查询界面, 它的特点是多种方式查询: 查询的条件是以非空的一组内容为查询条件, 这样即可以实现模糊查询, 也可以实现所需要的项目. 点击查询, 即可得到右图的入库资料管理的界面.

3.4 应用程序运行

在 Pocket PC 上运行建材行业进销存软件, 需要注册与访问 Pocket Access 数据库相关的动态链接库. 首先在 eVC 中打开建材行业进销存软件的工作空间 (Open Workspace): Designs.vcw, 然后

- 1) 选择 EVC 菜单 Tools Remote File Viewer, 弹出对话框 Select a Windows CE Device
- 2) 选择 Pocket PC 2003 Emulator, 此时先出现 Pocket PC 画面, 随后弹出 Windows CE Remote File Viewer 窗口;
- 3) 在左边的树型结构中选择 Windows, 然后选择 File Export File (即工具条上的)
- 4) 找到并打开文件夹 Windows CE Share File
- 5) 分别打开 adoce31.dll adocedb31.dll adoceoledb31.dll adoxce31.dll ssce20.dll ssceca20.dll dllregister

最后回到 eVC 选择 Build Designs.exe 并运行该软件, 即可进入 Pocket PC 进销存软件的界面.

(下转第 43 页)

neighbor 192.168.1.10 send-label

●配置 BGP多路径共享 : maximum-paths bgp 5

●验证 6PE配置 :

show ipv6 cef 2001:0DB8:DDDD: /64 show bgp ipv6 2001:0DB8:DDDD: /48

3 总结

3.1 应用效果

通过在 IPv4 MPLS VPN Core 上对 PE 路由器的改造, 不仅保持原网络中 PE 路由器完成标签交换、器维护 IPv4-VRF、承担路由反射器等基本功能, 还可以实现以下效果:

●同时支持 IPv4 和 IPv6 能够进行 IPv4 和 IPv6 路由存储与转发, 实现 IPv6 sites 与 IPv6 sites IPv4 sites 与 IPv6 sites 的连接

●同时支持 IPv4 VPN 和 IPv6 VPN, 而且增加 IPv6 VPN 的配置和 IPv4 VPN 的配置相同;

●IPv4 MPLS VPN Core 对 IPv6 Sites 透明, 即 IPv6 Sites 不需要关心 MPLS 标签交换;

●对于 IPv6 用户端而言, 不需要关心网络中其他 IPv6 Sites 的变化 (由 PE 路由器进行维护)。

3.2 不足及展望

论文未对在 IPv4 MPLS VPN 部署 6PE /6VPE 以承载 IPv6 带来的网络流量和 6PE 路由性能问题进行研究和阐述^[6]。同时, 6VPE 方式对于 PE 路由器的性能, 特别是信令开销比价大。

参考文献:

- [1] 华为 - 3Com 技术有限公司. IPv6 技术 [M/OL]. 北京: 清华大学出版社, 2004
- [2] van Peepeljak MPLS 和 VPN 体系结构 CCIP 版 [M]. 北京: 人民邮电出版社, 2003
- [3] Cisco Systems Inc. Implementing IPv6 over MPLS [M]. January, 2004
- [4] Hinden R. RFC 2460 IPv6 Addressing Architecture [M]. April 2003
- [5] Conta A, Deering S. RFC 2473 Generic Packet Tunneling in IPv6 Specification [M]. December 1998
- [6] Raghunathan R. Ed. RFC 4022 IP Version 6 Management Information Base for the Transmission Control Protocol [M] March 2005
- [7] Carpenter B, Moore K. RFC 3056 Connection of IPv6 Domains via IPv4 Clouds [M]. February 2001
- [8] 沈俊鑫. MPLS VPN 与 IPsec VPN 融合在电子政务中的应用 [D]. 昆明: 云南大学出版社, 2006
- [9] 沈俊鑫. MPLS VPN 技术及产品探究 [D]. 昆明: 云南大学出版社, 2006
- [10] Jeremy D C. Cisco Systems draft-ietf-l3vpn-bgp-ipv6-06 [M]. USA: Cisco Press February 2005
- [11] 吴晟, 苏庆堂, 罗斌. 基于 Socket 和多线程技术的并发服务器的研究 [D]. 昆明: 昆明理工大学出版社, 2006

(上接第 39 页)

4 结束语

基于 Pocket PC 数据库应用软件与基于桌面的数据库应用软件有所不同, 它要受到 Pocket PC 资源的限制, 要做到简单实用. 本文所讨论的有关方法对基于 Windows CE 应用平台的数据库应用程序开发具有一定的参考意义.

参考文献:

- [1] Chris M. Windows CE 权威指南 [M]. 北京: 中国电力出版社, 2001
- [2] Douglas B. Windows CE 程序设计 [M]. 北京: 北京大学出版社, 2001
- [3] 黄聪明. Pocket PC 数据库应用程序设计 [M]. 北京: 清华大学出版社, 2002
- [4] 田东风. Windows CE 应用程序设计 [M]. 北京: 机械工业出版社, 2003
- [5] 范盛荣. Windows Mobile 应用程序开发实践 [M]. 北京: 科学出版社, 2006
- [6] 孟宪国, 胡建旺, 王丙坤, 等. ADOCE 技术在 eVC++ 中的应用研究 [J]. 微计算机信息 (测控自动化), 2004 20(5): 64-65
- [7] 张新房, 吕跃刚, 徐大平, 等. 使用 Platform Builder 配置 Windows CE 操作系统 [J]. 单片机与嵌入式系统应用, 2002 (7): 8-11

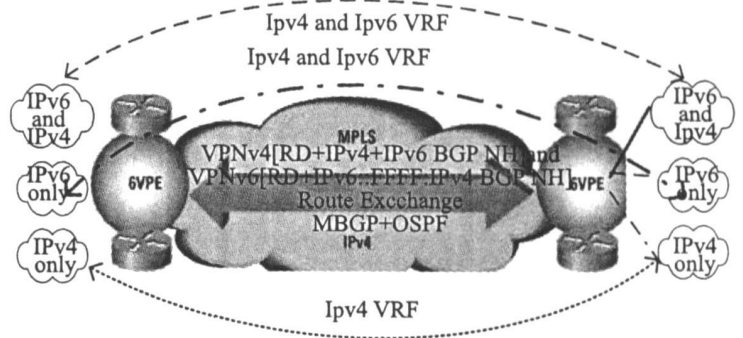


图3 6VPE逻辑图
Fig.3 6VPE logical fig