

在 VC++6.0 中使用 Oracle AppWizard 向导

本文介绍了在 VC++6.0 中使用 Oracle AppWizard 编写 Oracle 数据库应用程序的方法,重点讨论了生成程序时西文界面的汉字替换问题。

河南油田勘探开发研究院 袁里 李春霞 霍春亮
河南省唐河县张店镇 袁丁

引言

随着 Oracle 大型数据库系统在各行各业的广泛应用,基于 Oracle 的 Client/Server 模式的数据库应用软件将大量增多,Oracle 公司也在其产品 Oracle8i Client for Win9x 中提供了一个基于 VC++ 的 Oracle 客户应用程序向导——Oracle AppWizard for MS VC++,用该向导产生的应用程序的最大优点是不使用 ODBC,直接挂在 VC++ 下,用 oradcl 数据源控件和 DBGrid 控件直接访问 Oracle 数据库,不仅使 Oracle 应用程序的开发方便、快捷,而且效率很高,越来越受到 Oracle 数据库应用软件开发人员的青睐。但由于 Oracle AppWizard 生成的软件界面是西文的,而且数据库表的域名大都是汉语拼音码或西文缩写,使得应用程序的界面很不友好,限制了 Oracle AppWizard 的应用,本文结合一个实例讨论了用 Oracle AppWizard 生成程序的方法,并解决了几个西文的汉字替换问题。

用 Oracle AppWizard 生成应用程序

用 Oracle AppWizard 生成 Oracle 数据库应用程序,

必须要有满足 Client/Server 模式的网络环境,Oracle7.0 以上的数据库服务器,安装了 Visual C++ 5.0 以上的编程环境以及挂在 VC++ 下 Oracle 客户端产品 Oracle8i Client for Win9x。具体操作如下:

1. 启动 VC++6.0,输入新项目名 Example,选择 Oracle AppWizard for MFC(exe)选项,单击 OK 按钮。
2. 指定服务器端正在运行中的数据库系统的一个用户名和相应的口令字,以及数据库的网络服务器名(也称数据库别名),单击 NEXT 按钮。
3. 选择数据库的表格类型和对数据库表的操作,可选的表格类型有单记录表格显示、多记录表格显示、主一从表格显示;可选的数据库表的操作有添加新记录、改变现有记录、删除现有记录。选择后单击 NEXT 按钮。
4. 选择登录用户的表及域名,选择后单击 NEXT 按钮。
5. 以后的步骤同 MFC AppWizard 向导相同,本文不再详述。

对产生的应用程序进行编译、链接、运行,运行结果如图 1、图 2 所示。

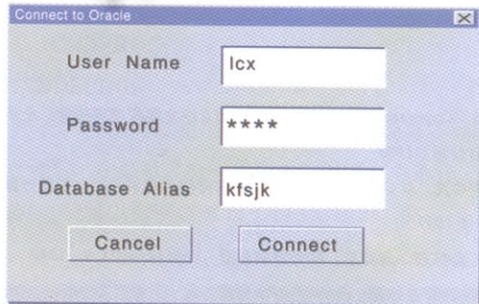


图 1

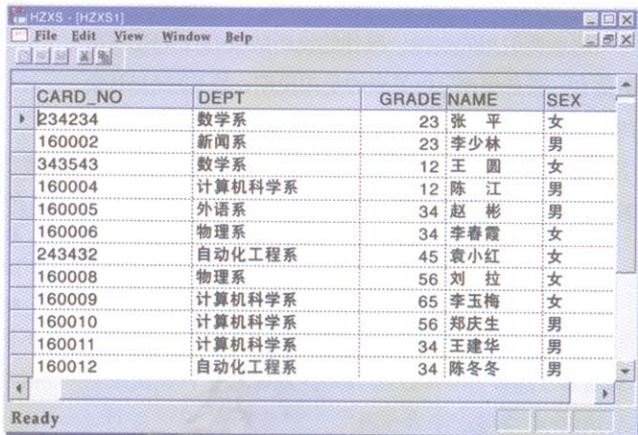


图 2

应用程序界面的汉字替换问题

在用Oracle AppWizard生成数据库应用程序时,在回答What language would you like you resources in?时,该向导屏蔽了其他语言的支持,只有“英语 [美国] (APPWZENU.DLL)”一种选择项,造成在界面上添加汉字时出现怪字符;另外,如图2所示,数据库表的域名大都是汉语拼音码或西文缩写,使得界面很不直观,也要进行替换,下面介绍几种解决方法。

1. 汉语言支持问题, 具体操作如下

(1)首先检查硬盘中VC++安装目录中的DevStudio\SharedIDE\bin\ide下有无动态链接库文件Appwzchs.dll(简体中文的语言支持文件),如果没有则从VC++光盘中的DevStudio\SharedIDE\bin\ide拷入。

(2)在项目工作间(Workspace)中选择资源(resources)标签,打开各类资源,逐个右键单击每一资源项,选择资源属性,将Language属性设置为Chinese(P.R.C)。

2. 数据库域名的汉字替换

数据库域名的汉字替换有两种方式,第一种利用控件DBGrid提供的接口,构造一个域名类CFields来实现。首先把控件DBGrid对应的类CmsDgridCtrl加入到项目中,然后用MFC ClassWizard向导添加DBGrid控件类CmsDgridCtrl的类对象m_grid,类CmsDgridCtrl中的成员函数LPDISPATCH GetColumns(const VARIANT& Index)可得到控件DBGrid的域名,用构造的类Cfields接管控件DBGrid的域名信息,通过其成员函数void SetDataField(LPCTSTR lpszNewValue)进行汉字替换,其实现代码如下:

```
// 构造域名类 CField
class CField : public COleDispatchDriver
{
public:
// 构造函数,完成与控件DBGrid的接口,接管数据库表的域信息
CField(LPDISPATCH pDispatch) : COleDispatchDriver(pDispatch) {}
public:
CString GetDataField(); // 得到当前域名
void SetDataField(LPCTSTR lpszNewValue); // 设置新域名
};
// 域名类 CField 实现函数
```

```
#include "stdafx.h"
#include "Field.h"
CString CField::GetDataField()
{
    CString result;
    InvokeHelper(0x5, DISPATCH_PROPERTYGET, VT_BSTR, (void*)&result, NULL);
    return result;
}
void CField::SetDataField(LPCTSTR lpszNewValue)
{
    static BYTE parms [] = VTS_BSTR;
    InvokeHelper(0x5, DISPATCH_PROPERTYPUT, VT_EMPTY, NULL, parms, //0x6
        lpszNewValue);
}
// 应用程序视类的初始化函数
void CHZXSView::OnInitialUpdate()
{
    m_pDynaset = &GetDocument()->m_HZXSDynaset;
    m_pDynaset->OpenQuery(GetDocument()->m_database);
    CFormView::OnInitialUpdate();
    m_dataControl.SetRecordset((LPDISPATCH)(m_pDynaset->Internal()));
    GetFieldName("BORROWERS"); //把BORROWERS表的域信息调入内存 //篇幅所限,实现函数略。
    m_grid.SetCaption ("借阅者情况表"); //向DbGRID控件加标题
    SetFieldName(); //用汉字替换数据库表的西文域名
}
// 汉字替换数据库表的西文域名的实现函数
void CHZXSView::SetFieldName()
{
    ODynaset set(GetDocument()->m_HZXSDynaset); //定义动态记录集并与
// 当前动态记录集相连接
    int FieldCount=set.GetFieldCount(); //读取字段个数
    for(int i=0;i<FieldCount;i++){
```

```

const char *FdName;
FdName=set.GetFieldName(i); //按查询语句
的顺序读取字段名
// 把 CField 类与 DBGRID 控件相连接
Cfield column(m_grid.GetColumns
(COleVariant(FdName)));
// 数组 FieldName [] 为域名缓冲区, 格式为: 8 字符
西文域名加汉字域名,
// 如 CARD_NO 借书证号
// 在视类初始化时由函数 GetFieldN a m e
("BORROWERS ")调入内存,
// FieldNumber 为域名个数
for(int j=0;j<FieldNumber;j++){ // 在域名缓冲
区查询
    CString temp1=FieldName [j];
    CString temp2=FdName;// 当前西文域名
    temp1=temp1.Left (7);//取出缓冲区中西文域名
    temp1.TrimLeft ()
    temp1.TrimRight ();
    if(temp1==temp2){//与当前西文域名比较
        CString temp3=FieldName [j]; // 相等,
取出相应的汉字域名
        temp3=temp3.Right (temp3.GetLength ()-
temp3.Find (' '));
        temp3.TrimLeft ();
        temp3.TrimRight ();
        column.SetDataField (temp3);// 实现替换
        break;
    }
}
}

```

的顺序读取字段名

// 把 CField 类与 DBGRID 控件相连接

Cfield column(m_grid.GetColumns (COleVariant(FdName)));

// 数组 FieldName [] 为域名缓冲区, 格式为: 8 字符西文域名加汉字域名,

// 如 CARD_NO 借书证号

// 在视类初始化时由函数 GetFieldN a m e ("BORROWERS ")调入内存,

// FieldNumber 为域名个数

for(int j=0;j<FieldNumber;j++){ // 在域名缓冲区查询

CString temp1=FieldName [j];

CString temp2=FdName;// 当前西文域名

temp1=temp1.Left (7);//取出缓冲区中西文域名

temp1.TrimLeft ()

temp1.TrimRight ();

if(temp1==temp2){//与当前西文域名比较

CString temp3=FieldName [j]; // 相等,

取出相应的汉字域名

temp3=temp3.Right (temp3.GetLength ()-temp3.Find (' '));

temp3.TrimLeft ();

temp3.TrimRight ();

column.SetDataField (temp3);// 实现替换

break;

}

}



图 3

以下实例的运行结果如图 3、4 所示。

另一种汉字替换方法利用数据库系统的 SQL 语言来实现, ORACLE 的标准 SQL 语句 SELECT 有设置特定域的别名功能, 结合本实例如下:

```

select CARD_NO AS 借书证号, NAME AS 姓名, DEPT AS 系别, GRADE AS 年级, SEX AS 性别 from LCX.BORROWERS;

```

然后把修改过的 SELECT 语句加入到应用程序中即可实现汉字替换。这种方法实现较简单, 因对每一个表、每一个域都具体操作, 造成编程工作量大, 程序冗长, 代码执行效率低, 代码不灵活, 不利于代码的修改及程序的维护, 在应用软件开发中很少采用。

结束语

本程序在 Visual C++6.0 环境下调试编译通过, 客户端操作系统为 Windows98, 并且安装了 Oracle8i Client for Win9x, 服务器端为 SUN 工作站, 操作系统为 Solaris 2.7, 数据库服务器为 ORACLE 8.1.5。

用本文讨论的 Oracle AppWizard 向导生成的基于 Oracle 数据库系统的应用程序具有容易实现、代码紧凑、代码执行效率高等特点, 比通过 ODBC 生成的应用程序速度提高了很多, 在对多表大规模数据量查询时, 几乎没有等待时间, 而且软件开发周期也大幅度减少, 是一种较好的 ORACLE 数据库应用软件开发工具。■



图 4