


# 管理信息系统

1. 管理信息系统概论
2. 计算机网络技术基础
-  3. 企业计算模式
4. 管理信息系统的系统规划
5. 管理信息系统的系统分析
6. 管理信息系统的系统设计
7. 管理信息系统的系统实施
8. 管理信息系统运行维护与评价
9. 管理信息系统的项目管理

## 3 企业计算模式

本章介绍计算机应用系统的三种计算模式，即：

(1) 单主机计算模式

(2) 客户/服务器计算模式

(3) 浏览器/服务器计算模式

讨论其各自的结构特征及对管理信息系统发展的影响。

# 什么是“企业计算模式”？

计算机应用系统中**数据与应用**（程序）的分布方式称为企业计算机应用系统的计算模式，有时也称为企业计算模式。

# 3 企业计算模式

## 3.1 单主机计算模式

3.2 客户机/服务器计算模式

3.3 浏览器/服务器计算模式

\*3.4 客户机/服务器模式的技术实现

\*3.5 浏览器/服务器模式的技术实现

## 3.1 单主机计算模式

1985年以前，计算机应用一般是以单台计算机构成的单主机计算模式。

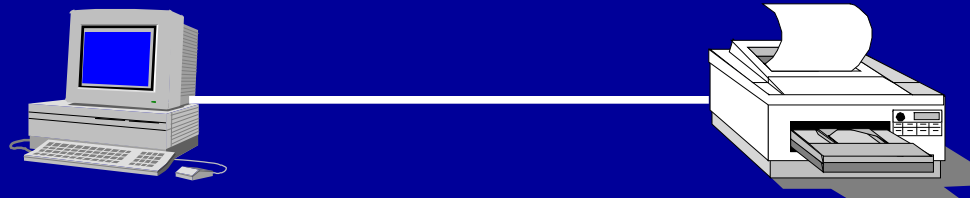
单主机计算模式又可细分为两个阶段：

- 单主机—单终端
- 单主机—多终端

## (1) 单主机—单终端计算模式

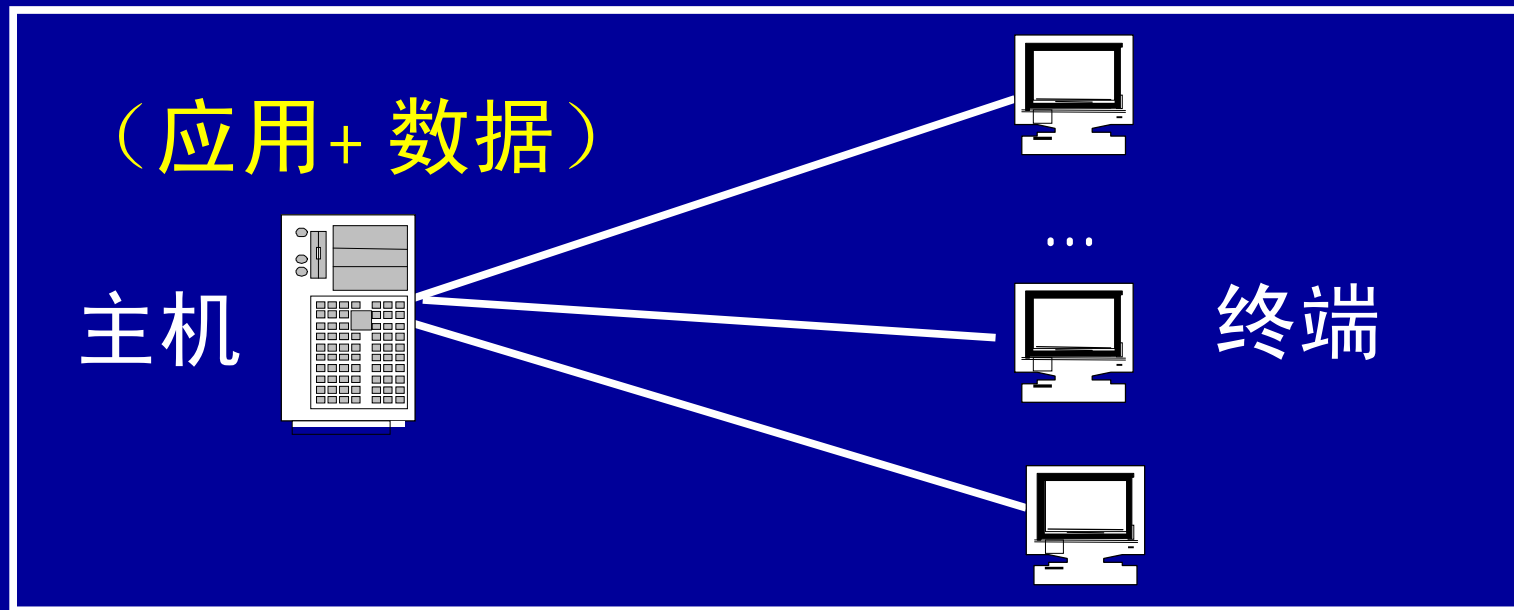
这是单主机计算模式的早期阶段，操作系统为单用户操作系统，一般只有一个控制台，且仅限于单项应用，如劳资报表统计等。

(应用+数据)



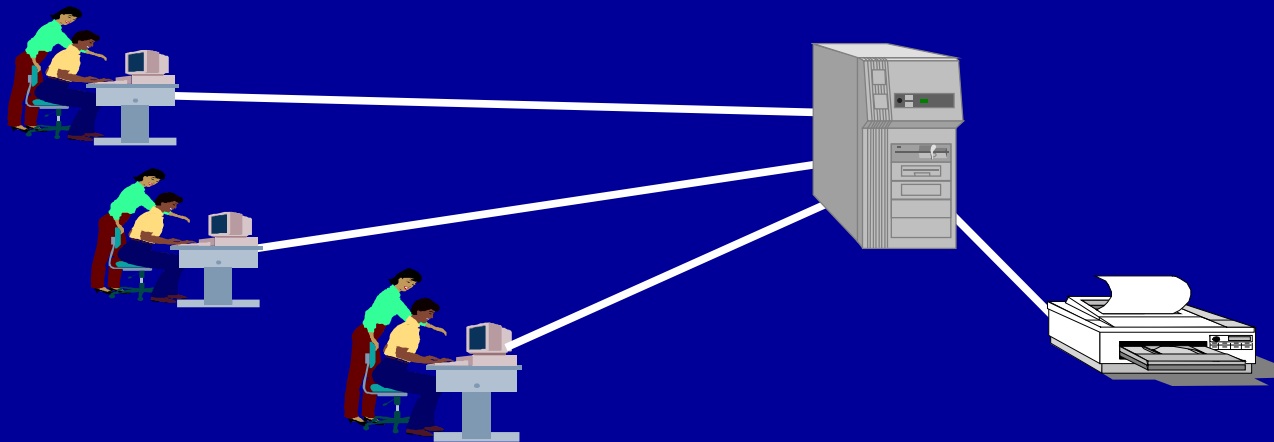
## (2) 单主机—多终端计算模式

分时多用户操作系统的研制成功，以及计算机终端的普及使早期的单主机计算模式发展成为单主机—多终端的计算模式。



# 单主机—多终端计算模式的工作方式

在单主机—多终端的计算模式中，用户通过终端使用计算机。每个用户都感觉是在独自享用计算机的资源，主机分时轮流为每个用户提供服务。





# 3 企业计算模式

3.1 单主机计算模式

 3.2 客户机/服务器计算模式

3.3 浏览器/服务器计算模式

\*3.4 客户机/服务器模式的技术实现

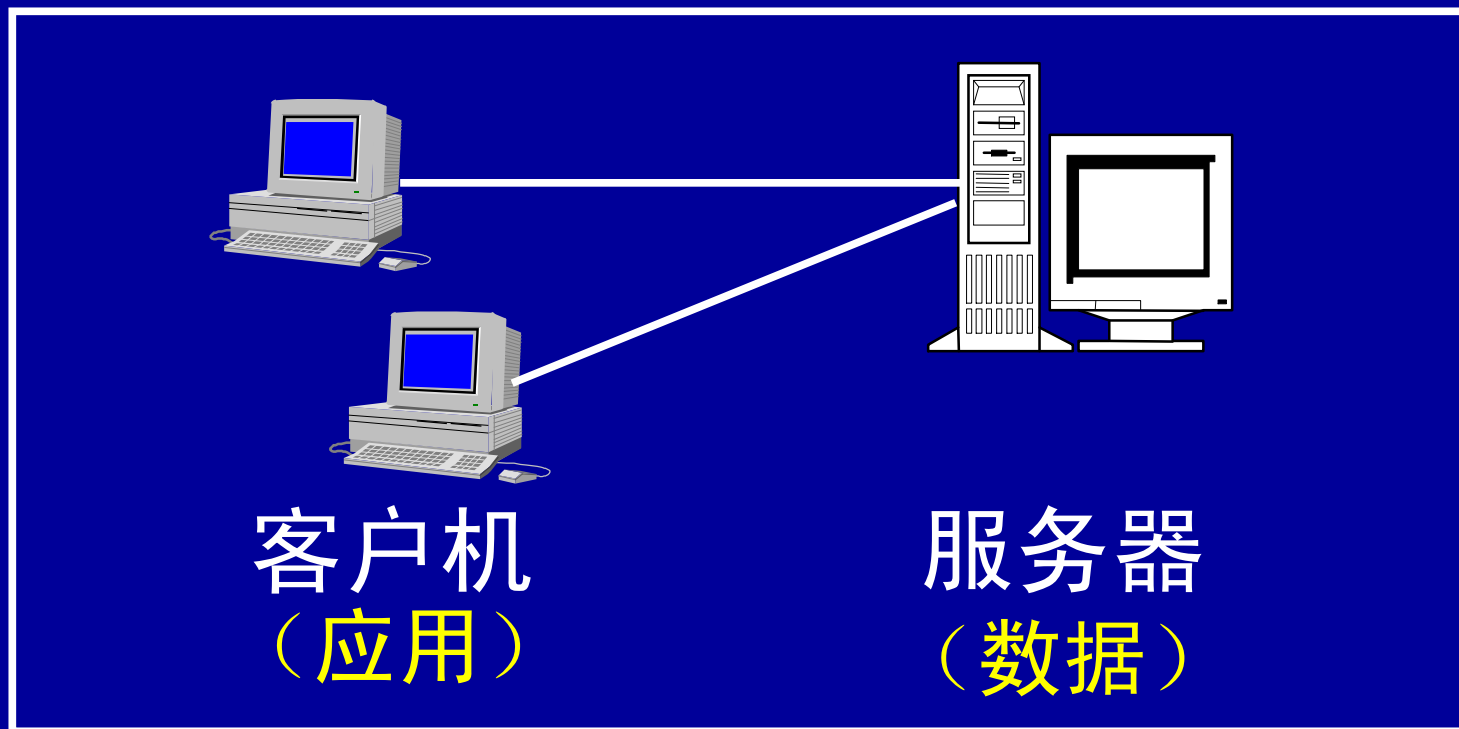
\*3.5 浏览器/服务器模式的技术实现

## 3.2 客户/服务器计算模式

计算机网络的出现后，用户通过网络共享计算机资源，计算机之间通过网络可协同完成某些数据处理工作。在这种背景下形成了分布式**客户/服务器**（**Client/Server—C/S**）计算模式。

# 客户/服务器计算模式的结构

在客户/服务器模式中，网络中的计算机被分为**服务器**和**客户机**两大类。



# 客户/服务器模式应用的开发工具

C/S模式客户端应用程序的开发工具可分两类：

(1) 针对特定数据库管理系统的开发工具（如Oracle的Developer 2000）。

(2) 对大部分数据库系统都适用的前端开发工具（如Power Builder, Visual Basic, Visual C, Delphi, C++Builder, Java等）。

# 客户/服务器计算模式的优点

- (1) 成熟
- (2) 网络开销低
- (3) 应用开发工具丰富

## 客户/服务器计算模式的缺点

(1) 每一个客户机都必须安装并正确配置相应的数据库客户端驱动程序。这样，应用程序（也必须安装在客户机上）才能访问数据库。

(2) 由于应用程序被分散在各个客户机上，使系统的维护困难且容易造成不一致性。

# 3 企业计算模式

3.1 单主机计算模式

3.2 客户机/服务器计算模式

 3.3 浏览器/服务器计算模式

\*3.4 客户机/服务器模式的技术实现

\*3.5 浏览器/服务器模式的技术实现

## 3.3 浏览器/服务器模式

浏览器/服务器（Browser/Server—B/S）模式是在C/S模式的基础上发展而来的。

导致B/S模式产生的源动力来自不断增加的业务规模和不断复杂化的业务处理请求。



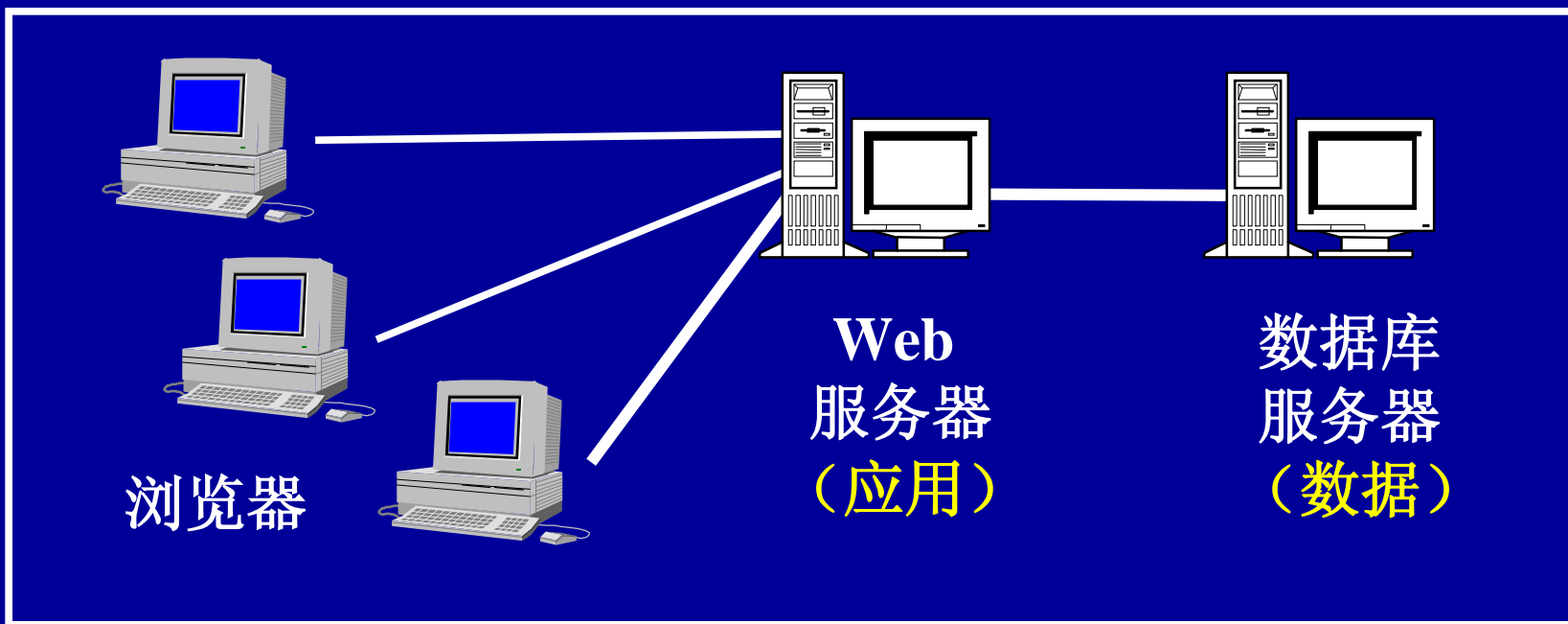
# 计算机应用的三层结构

解决复杂的业务处理要求的方法是在传统C/S模式的基础上，增加中间应用层（商务逻辑层），由原来的两层结构（客户/服务器）变成三层结构：



# 浏览器/服务器计算模式的结构

B/S模式是三层应用结构的一种实现方式，其具体结构为：浏览器/Web服务器/数据库服务器：



# 浏览器/服务器计算模式的特点

## 优点:

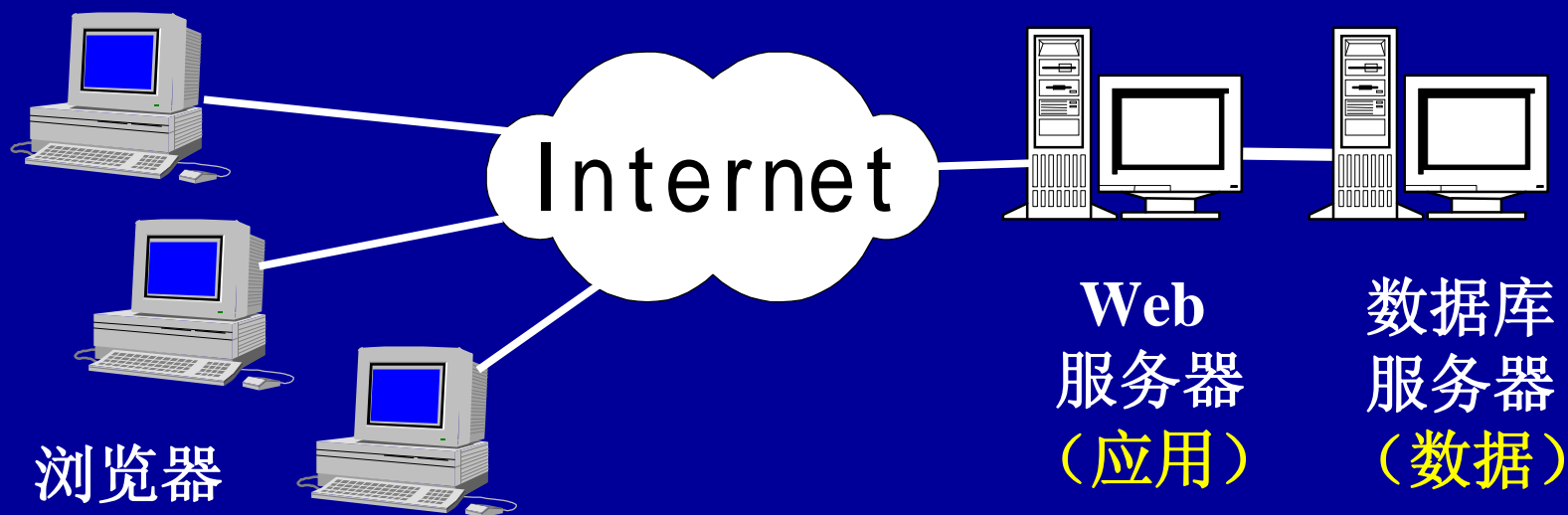
- (1) 客户端可做到零维护。
- (2) 应用（程序）维护方便。
- (3) 可使系统的潜在用户方便地使用系统。

## 缺点:

要求网络有较高的传输速度。

# B/S模式改变了计算机应用系统面貌

B/S模式极大地扩大了管理信息系统的功能覆盖范围，可使系统的潜在用户方便地使用系统，革命性地改变了计算机应用系统面貌。



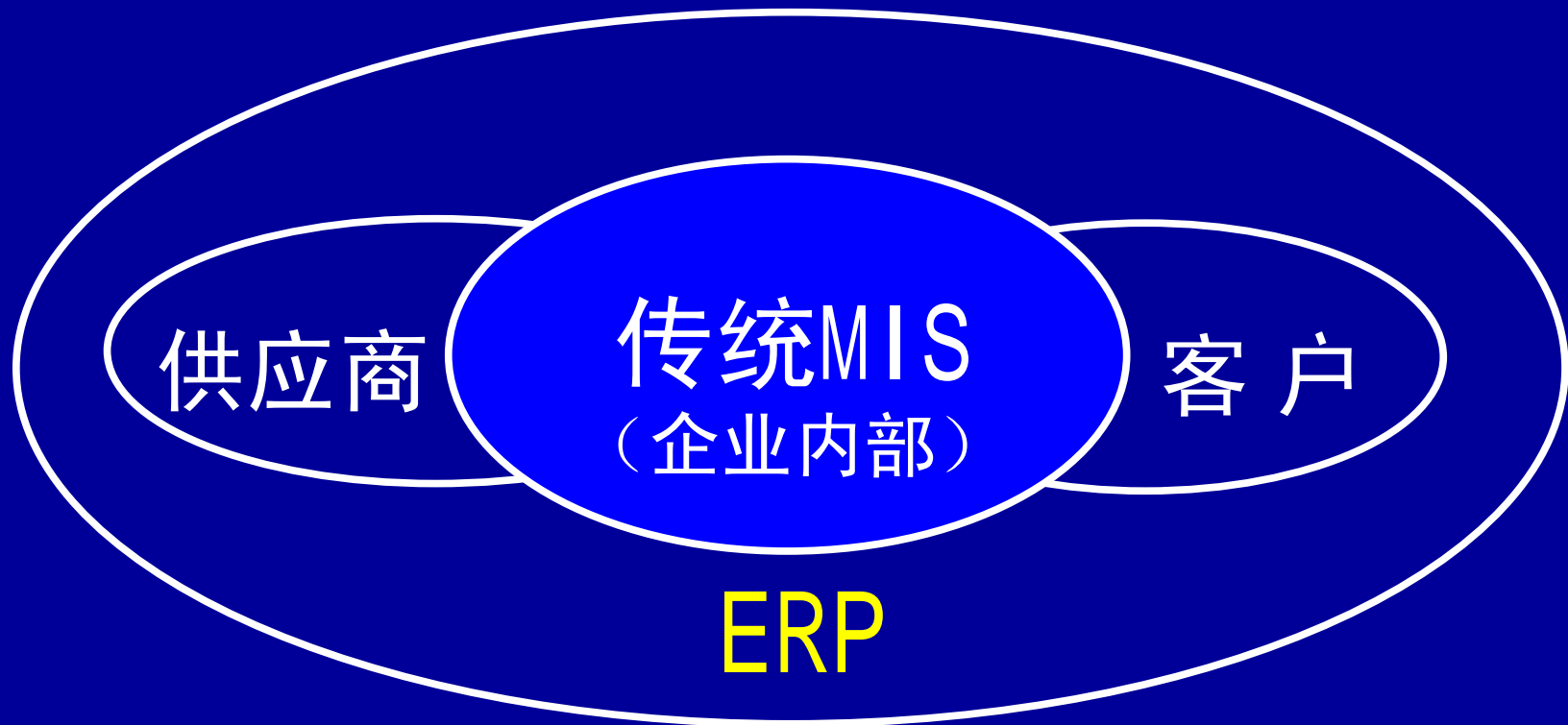
# B/S模式从技术上保证需要向潜在用户提供服务的应用的实现

(1) 企业资源规划系统 (Enterprise Resource Planning System —ERP)。

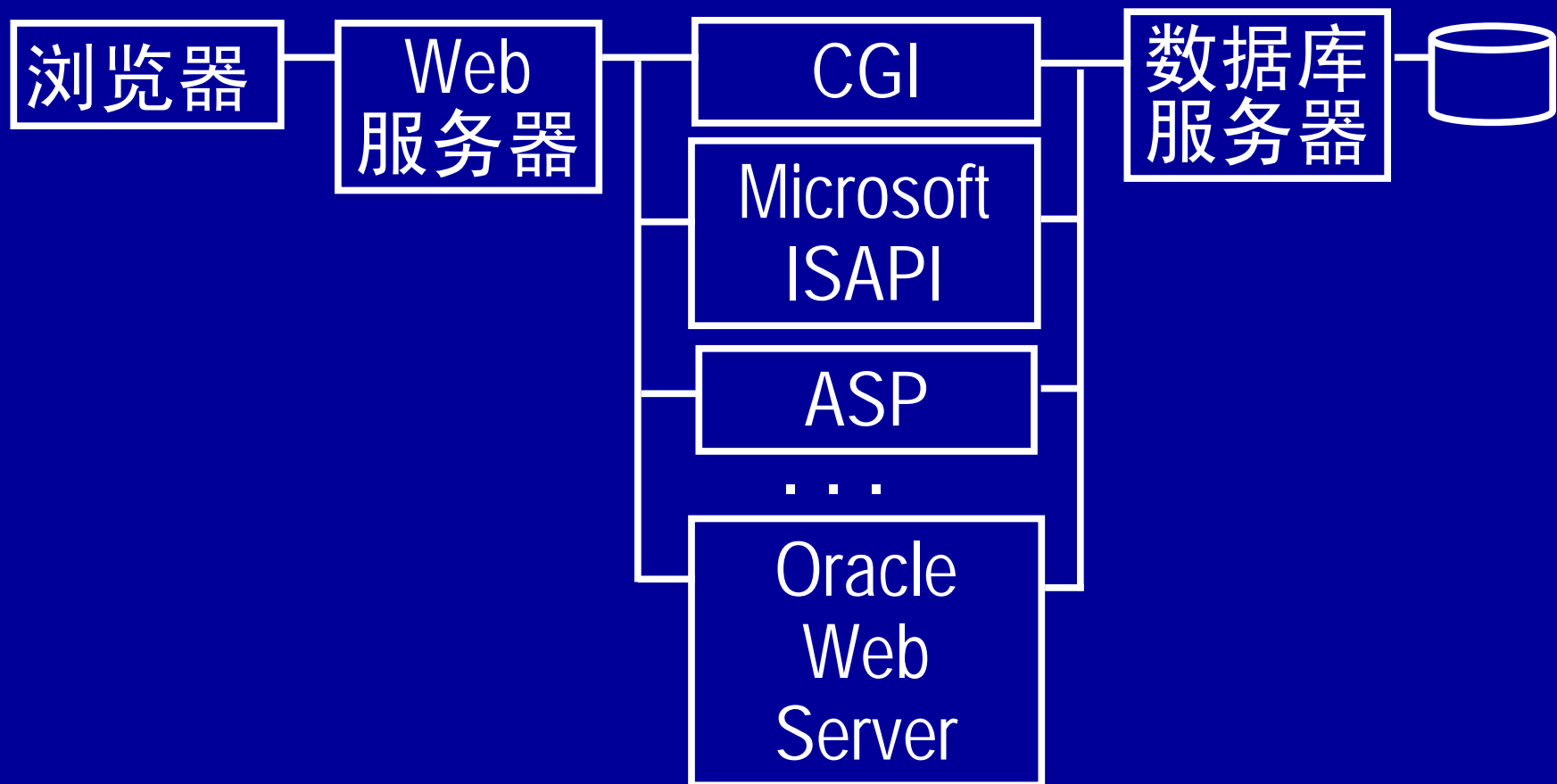
(2) 客户关系管理系统 (Customer Relation Management—CRM)。

(3) 电子商务 (Electronic Commerce—EC)。

# ERP系统与传统MIS的关系



# B/S模式中应用程序 通过CGI或中间件访问数据库



# 3 企业计算模式

3.1 单主机计算模式

3.2 客户机/服务器计算模式

3.3 浏览器/服务器计算模式

 \*3.4 客户机/服务器模式的技术实现

\*3.5 浏览器/服务器模式的技术实现



## 3.4 客户/服务器模式技术实现

- (一) 客户/服务器模式的工作原理
- (二) 通过交互式查询工具实现数据库访问
  - (1)本地访问
  - (2)远程访问
- (三) 通过开发工具的专用接口实现数据库访问
- (四) 通过开放式数据库接口实现数据库访问

## （一）客户/服务器模式的工作原理

前台应用程序（客户）提供用户界面，接收用户输入，向服务程序（服务器）发出相应的工作请求（Job）；后台服务程序负责响应（Answer）前台应用程序的请求，将结果返回给前台。



**注意：** 客户端应用程序和服务程序既可在同一台计算机上，也可在不同的计算机上。

## （二）通过交互式查询工具 实现数据库访问

数据库管理系统厂商一般都提供一种称为“交互式查询工具”的客户端软件来实现对后台服务器端数据库中数据的访问。

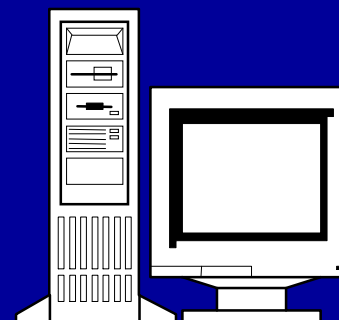
下面以Oracle数据库厂商提供的交互式查询工具SQL\*Plus为例，介绍C/S模式下的数据访问的实现技术。

## (1) 本地访问

最简单的一种情况是用户从安装了数据库管理系统软件的计算机（一般称其为数据库服务器）本地直接访问数据库。

### 实验环境

操作系统: Windows NTServer4.0,  
数据库管理系统: Oracle7.3 for NT,  
交互式查询工具: SQL \*Plus 3.2



或 { 操作系统: Windows 98,  
数据库管理系统: Personnel Oracle7.3,  
交互式查询工具: SQL \*Plus 3.2

## (1) 本地访问—运行SQL\*Plus后的登录界面

Oracle提供的查询工具SQL\*Plus是一种交互式的数据访问工具。用户可应用该工具实现与数据库的连接、创建数据库表、向数据库表添加数据记录、修改数据库记录及对数据库表的统计查询等操作。

登入

用户名称(U):

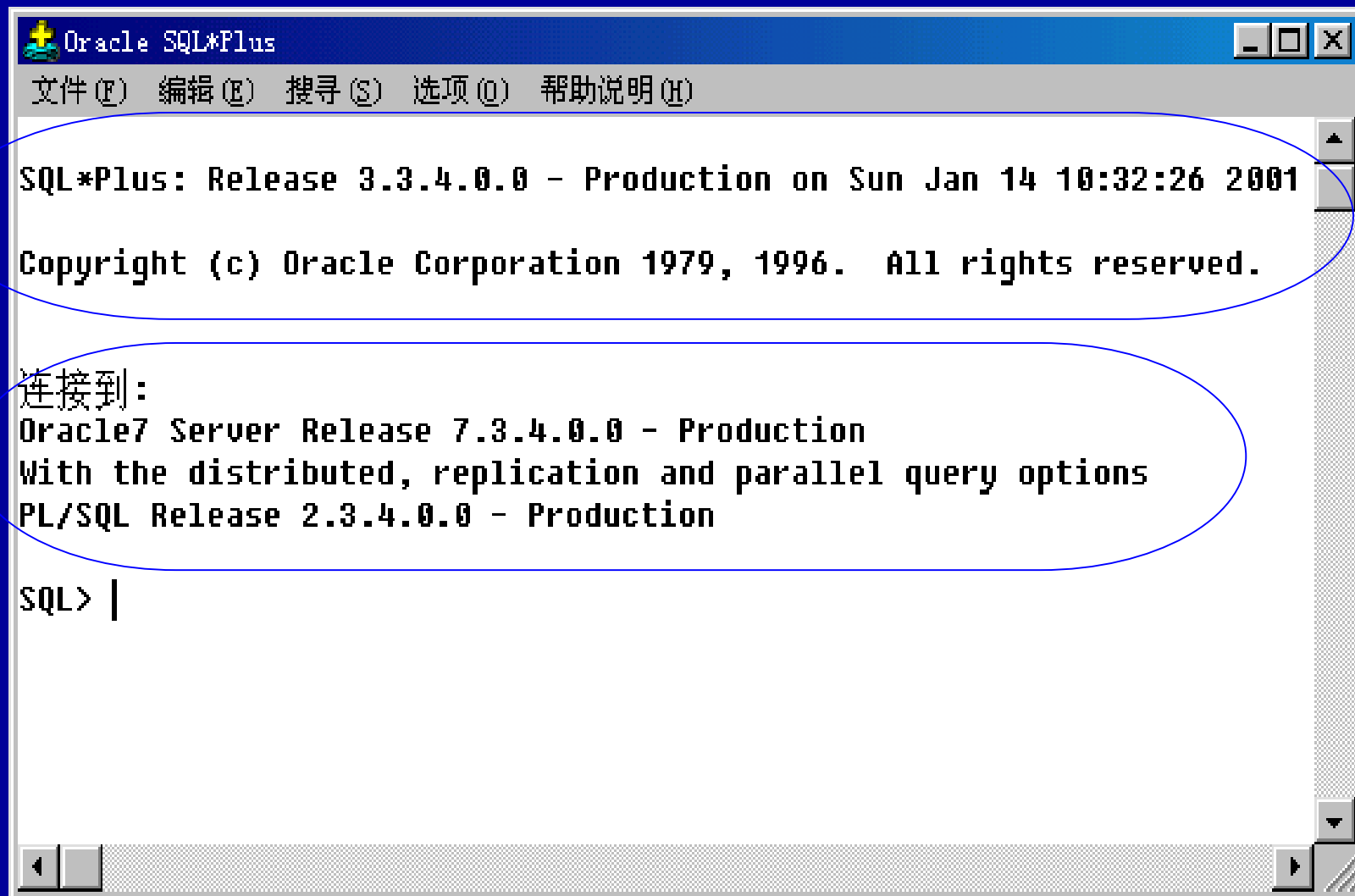
口令(P):

主机字符串(H):

确定 取消

本地访问  
本项不填

# (1) 本地访问—SQL\*Plus登录后显示版本信息



The screenshot shows a terminal window titled "Oracle SQL\*Plus". The menu bar includes "文件(F)", "编辑(E)", "搜寻(S)", "选项(O)", and "帮助说明(H)". The main text area displays the following information:

```
SQL*Plus: Release 3.3.4.0.0 - Production on Sun Jan 14 10:32:26 2001  
Copyright (c) Oracle Corporation 1979, 1996. All rights reserved.  
  
连接到:  
Oracle7 Server Release 7.3.4.0.0 - Production  
With the distributed, replication and parallel query options  
PL/SQL Release 2.3.4.0.0 - Production  
  
SQL> |
```

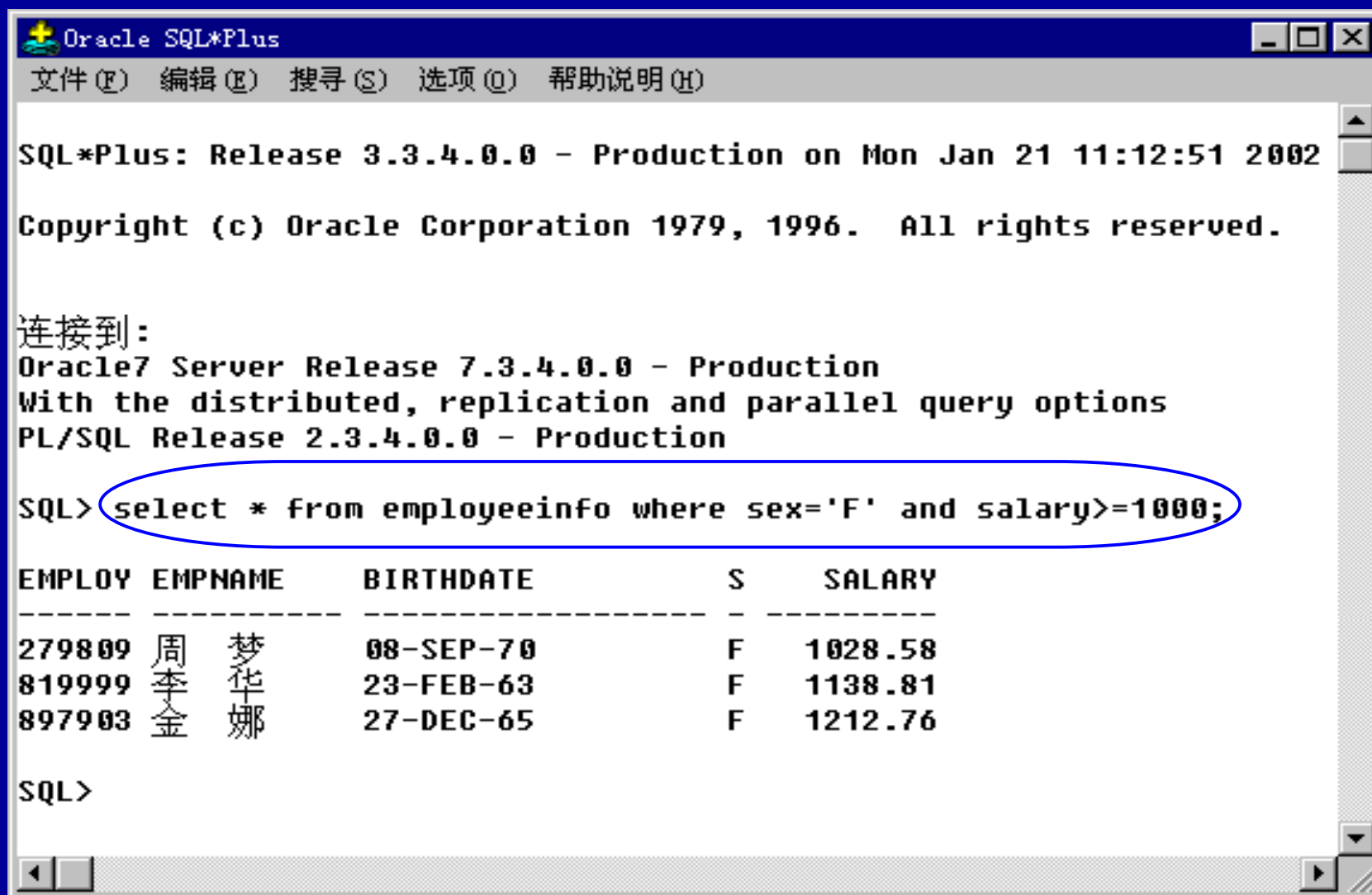
Two blue ovals are drawn on the image: one around the first two lines of text, and another around the "连接到:" section and its details.

## (1) 本地访问—创建数据库表并添加数据记录

在 SQL\*Plus 下，建立数据库表 employee info，并对表添加以下数据记录。

Emno	Empname	Birthdate	Sex	Salary
277052	王伟怀	19-JUL-69	M	1117.29
279809	周 梦	08-SEP-70	F	1028.58
456789	廖卫东	18-DEC-79	M	856.76
576868	谢 敏	22-SEP-76	F	962.73
807956	徐瑞忠	24-MAY-62	M	1168.36
809139	黄建伟	11-JAN-67	M	1038.91

# (1) 本地访问—查询数据库记录



The screenshot shows the Oracle SQL\*Plus interface. The title bar reads "Oracle SQL\*Plus". The menu bar includes "文件 (F)", "编辑 (E)", "搜寻 (S)", "选项 (O)", and "帮助说明 (H)". The main window displays the following text:

```
SQL*Plus: Release 3.3.4.0.0 - Production on Mon Jan 21 11:12:51 2002  
Copyright (c) Oracle Corporation 1979, 1996. All rights reserved.  
  
连接到:  
Oracle7 Server Release 7.3.4.0.0 - Production  
With the distributed, replication and parallel query options  
PL/SQL Release 2.3.4.0.0 - Production  
  
SQL> select * from employeeinfo where sex='F' and salary>=1000;
```

The query results are displayed in a table format:

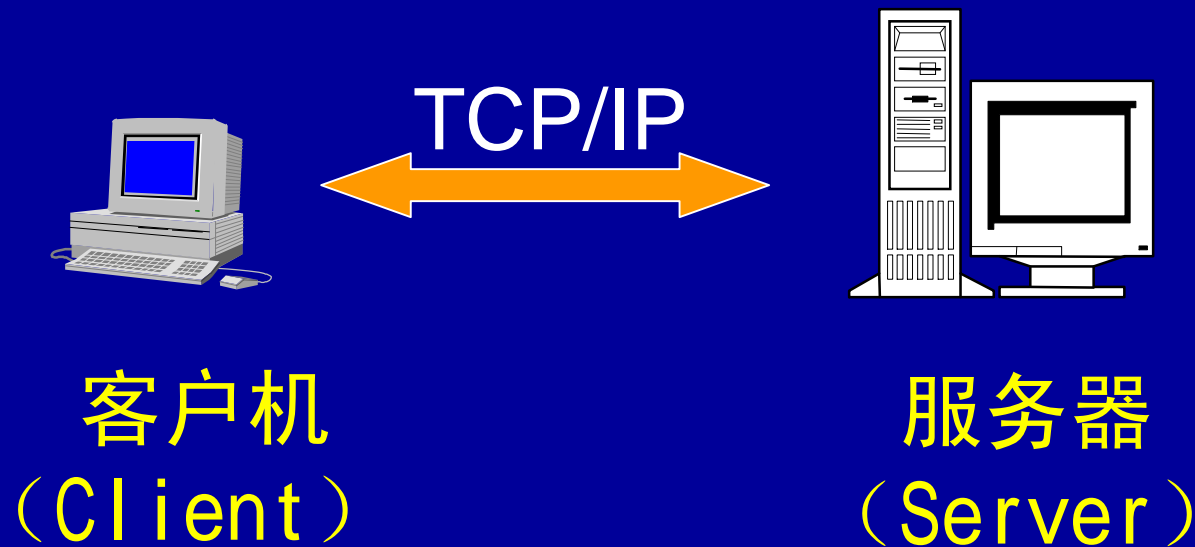
EMPLOY	EMPNAME	BIRTHDATE	S	SALARY
279809	周 梦	08-SEP-70	F	1028.58
819999	李 华	23-FEB-63	F	1138.81
897903	金 娜	27-DEC-65	F	1212.76

The SQL prompt "SQL>" is visible at the bottom of the window.

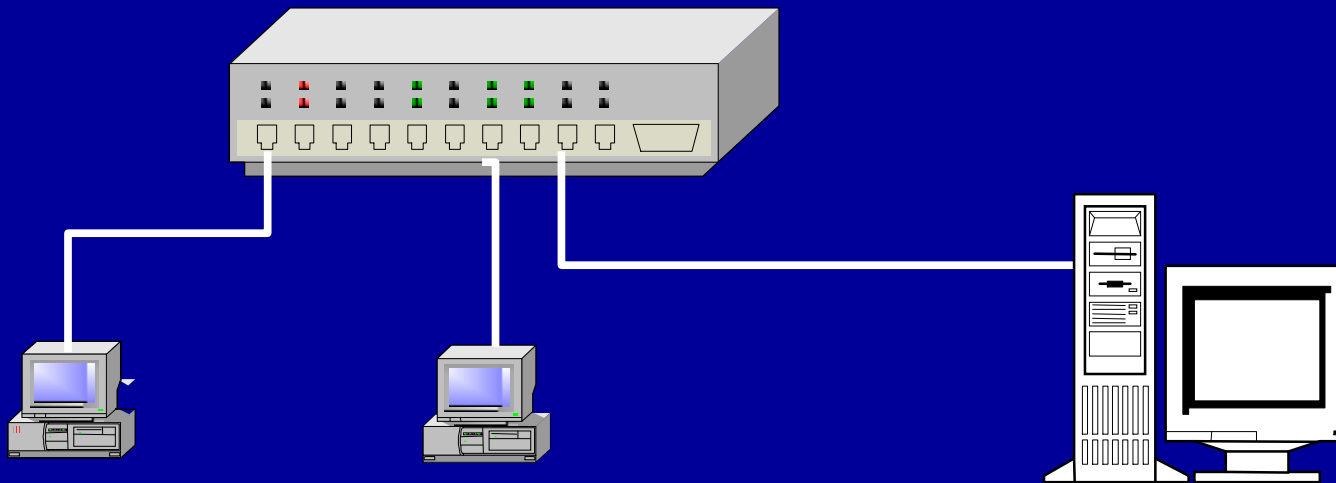


## (2) 远程访问

这里对数据库的远程访问是指客户端程序（Client）及服务程序（Server）不在同一台计算机上的C/S模式对数据库的访问形式。



## (2) 远程访问—实验环境



### 客户机

Windows98  
(TCP/IP协议)  
SQL\*Net  
SQL\*Plus

### 服务器

Windows NTServer4.0  
(TCP/IP协议, IP地址:  
192.168.63.12 )  
Oracle7.3 for NT

## (2) 远程访问—建立数据库别名

在客户端计算机上用SQL\*Net建立一个数据库别名(Database Alias),并对其进行参数配置。

```
Database Alias:  myorasvr  
  
Protocol:  TCP/IP  
  
Host Name:  192.168.63.12  
  
Database Instance:  ORCL
```

## (2) 远程访问 — 在客户端运行SQL\*Plus

运行SQL\*Plus。注意：在“主机字符串”填入的是我们已定义的服务器别名myorasvr。

登入

用户名称(U): stu

口令(P): \*\*\*

主机字符串(H): myorasvr

确定 取消

远程访问时  
本项指明访  
问哪一个数  
据库服务器

## (2) 远程访问

当我们确认后，SQL\*Plus将显示版本信息及SQL提示符，表明用户正确连接到数据库，可以访问数据库了。

此时，若用户在SQL提示符后键入上文的SQL命令，将得到同样的结果。

对数据库的本地访问和远程访问在建立客户端与服务器端的连接之后是一样的。

## （三）通过开发工具的专用接口 实现数据库访问

### 数据库的高级用户普通用户

■ **高级用户**：具备较高的计算机及数据库管理的实际操作水平。一般采用交互式数据库访问工具（如前文所述的SQL\*Plus）进行数据库性能的调整、控制、维护等管理工作。

■ **普通用户**：仅关注与自己业务相关的数据和操作，希望能方便直观地录入、查询、维护所负责业务领域内的数据。普通用户用由开发人员编制的的应用程序来完成他们的数据处理工作。这些应用程序的开发，一般借助特定的开发工具来完成。

# 常用的应用程序开发工具

- Visual Basic(简称VB)
- Visual C++(简称VC)
- Power Builder(简称PB)
- Delphi

# Power Builder 6.0 访问数据库

这里以Power Builder 6.0 为例，介绍用Power Builder 6.0编制开发的客户端应用程序对数据库的访问。

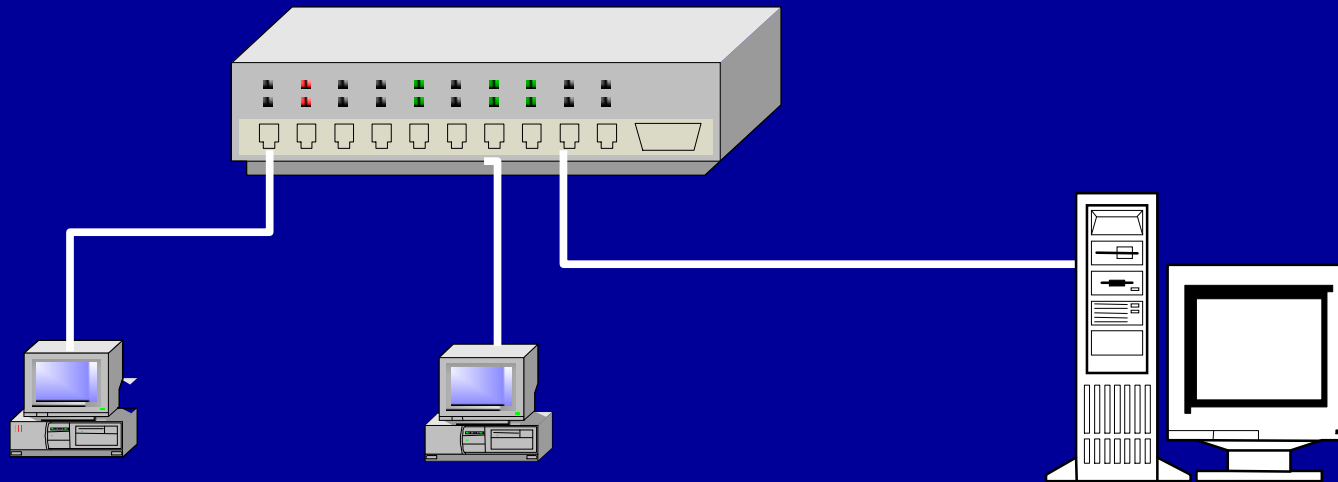
PB6.0中对数据操作是通过数据窗口（DataWindow）来实现的。

我们以前文对员工信息的查询及修改为例，可以按如下步骤给出其实现过程，读者可通过该过程了解C/S模式应用程序开发中的数据访问原理。



# Power Builder 6.0 访问数据库

—实验环境



## 客户机

Windows98

(TCP/IP协议)

SQL\*Net

Power Builder 6.0

## 服务器

Windows NTServer4.0

(TCP/IP协议, IP地址:

192.168.63.12 )

Oracle7.3 for NT

# Power Builder 6.0 访问数据库

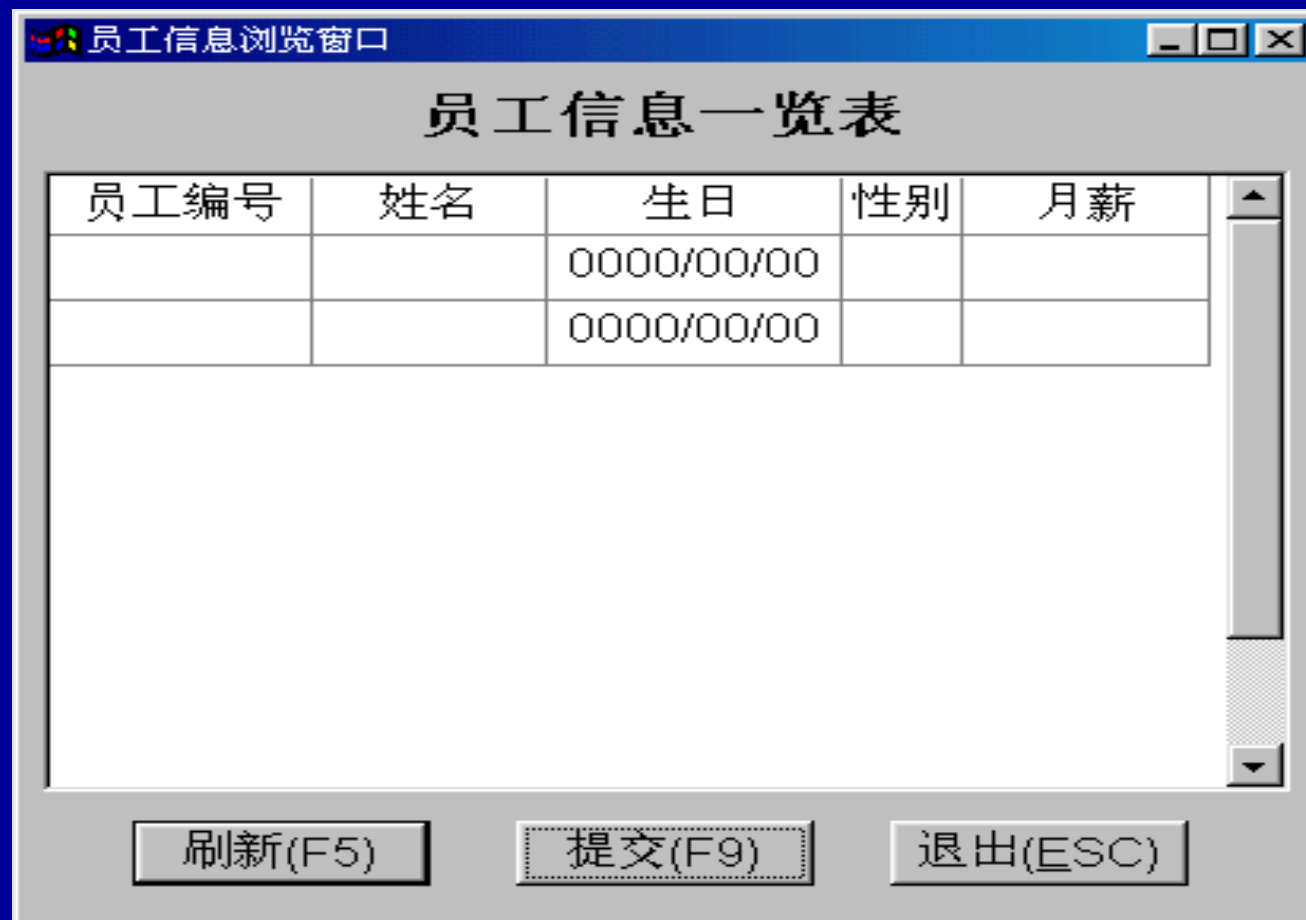
## —建立数据库别名

假设在客户端计算机上用SQL\*Net已了建立一个数据库别名(Database Alias),并对其进行参数配置。

```
Database Alias:  myorasvr  
  
Protocol:  TCP/IP  
  
Host Name:  192.168.63.12  
  
Database Instance:  ORCL
```

# 用 Power Builder 6.0 访问数据库的步骤

Step1: 在 PB6.0 中，创建数据窗口 dw\_employeeinfo



## Step2: 编写窗口的open事件的程序代码

```
sqlca.dbms='O73'  
sqlca.logid='stu'  
sqlca.logpass='stu'  
sqlca.servername='@myorasvr'  
connect ;  
if sqlca.sqlcode <> 0 then  
    MessageBox ("错误", "数据库联接错误," + "程序将退出: ~r~n" +  
        sqlca.sqlerrtext)  
    close(this)  
else  
    dw_employeeinfo.settransobject(sqlca)  
end if
```

**Step3: 在“刷新”按钮的click事件中添加如下程序代码:**

```
dw_employeeinfo.retrieve ( )
```

该程序代码的作用是: 从数据库中读取数据并覆盖数据窗口中数据。

**Step4: 在“提交”按钮的click事件中添加如下程序代码:**

```
dw_employeeinfo.update ( )
```

该程序代码的作用是: 将数据窗口中所做的修改写回数据库。

**Step5: 在“退出”按钮的click事件中添加如下程序代码:**

```
close(parent)
```

该程序代码的作用是关闭当前对象的“父”对象。本例中，当前对象为“退出”按钮，其“父”对象即为窗口。

## 程序运行过程（1）

打开主窗口，执行窗口“Open”事件中的程序代码。

该段代码将建立与数据库的连接，并指定数据窗口dw\_employeeinfo使用该数据库连接访问数据库。

此时数据窗口中还没有数据，因为此时仅仅连接了数据库，但并没有发出读取数据请求。

## 程序运行过程（2）

当用户点击“刷新”按钮时，系统将运行“刷新”按钮的click事件中的程序代码。

数据窗口dw\_employeeinfo的retrieve方法将自动生成相应的SQL命令并发送至数据库连接指定的数据库，服务器端返回结果显示在窗口上。



# 用户点击“刷新”按钮后的结果

员工信息浏览窗口

## 员工信息一览表

员工编号	姓名	生日	性别	月薪
456789	廖卫东	1979/12/18	M	856.76
576868	谢敏	1976/09/22	F	962.73
980931	向济辉	1970/08/06	M	997.38
279809	周梦	1970/09/08	F	1028.58
809139	黄建伟	1967/01/11	M	1038.91
893448	江文济	1965/07/28	M	1043.18
891212	孙凯	1971/05/15	M	1069.94
277052	王伟怀	1969/07/19	M	1117.29

刷新(F5)      提交(F9)      退出(ESC)

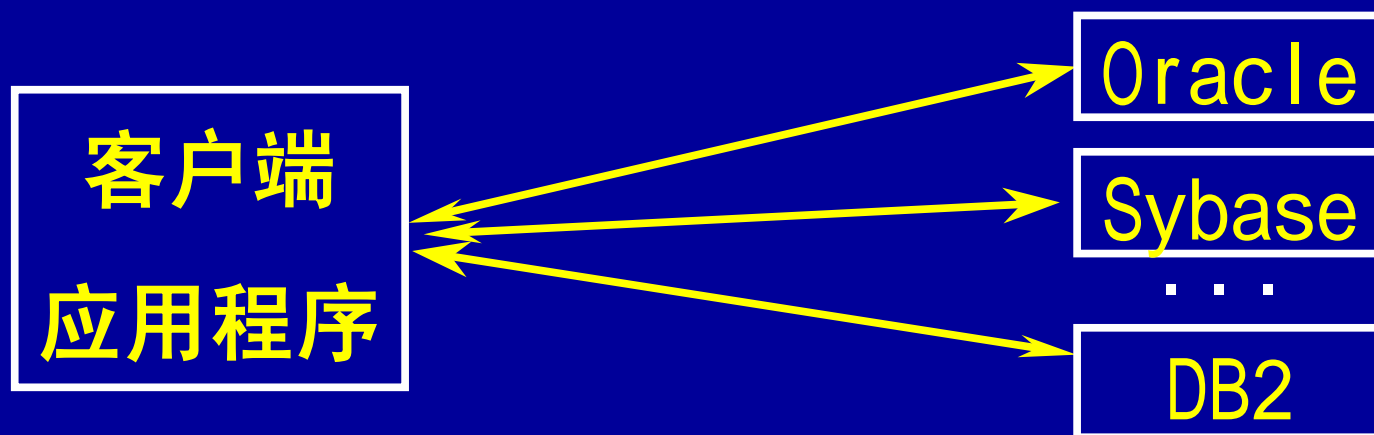
## 程序运行过程（3）

若用户对数据进行了修改操作，当点击“**提交**”按钮后，系统将修改的结果保存到数据库中。

当用户点击“**返回**”按钮时，系统将切断与数据的连接并关闭窗口。

## (四) 通过开放式数据库接口(ODBC)实现数据库访问

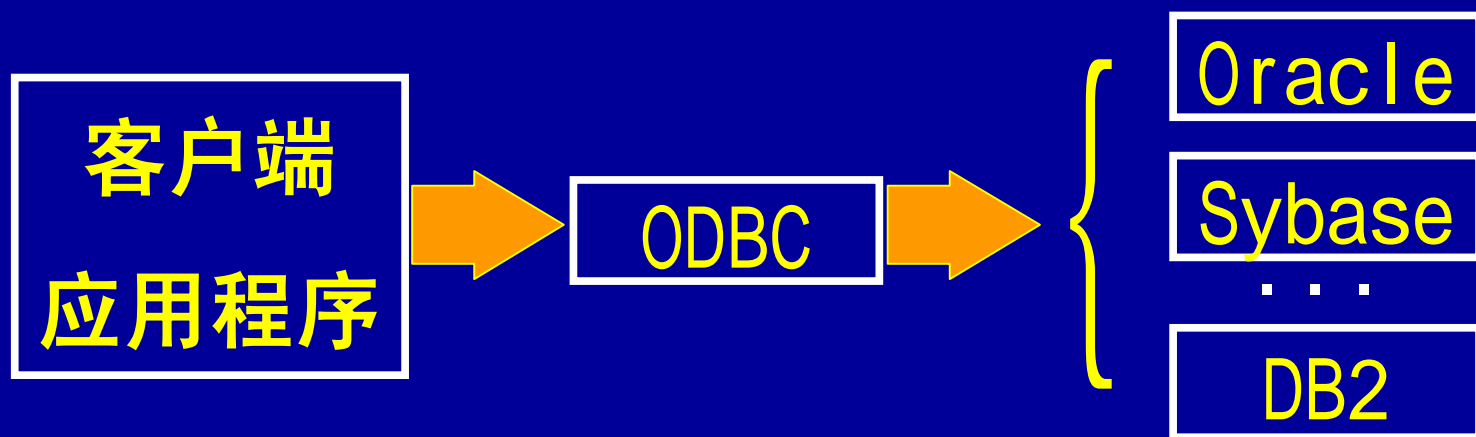
考虑到应用程序的通用性、可维护性、可扩充性等要求，有时同一程序必须处理与多种异质数据源的连接。



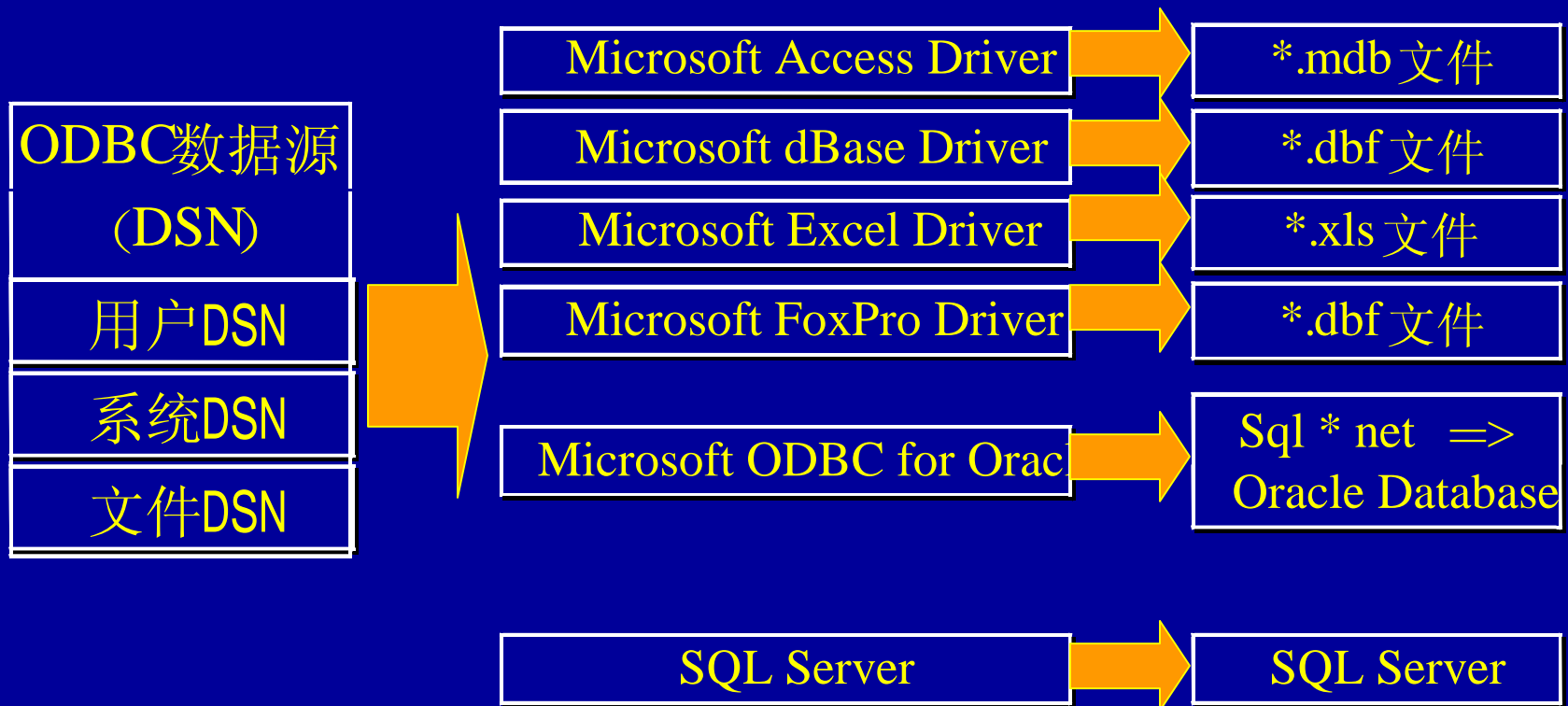
# 开放式数据接口ODBC工作原理

Microsoft 推出的开放式数据接口（Open Database Connectivity—ODBC）为异质数据库的访问提供了统一的接口。

ODBC提供了最大限度的操作弹性，一个应用程序可以通过一组通用的程序代码访问不同的数据库管理系统。



# ODBC工作原理示意图



# ODBC是一个封装器！

从上页图可以看出，ODBC实质上是一个封装器，它将对各种不同数据库的访问驱动接口封装在一起。而应用程序对数据源的处理是一致的，这样就屏蔽了不同驱动程序（Driver）之间的差异。

## 应用ODBC访问数据库的必要条件

- 支持该数据库的ODBC驱动程序已经正确安装
- 用户通过ODBC管理器正确配置了对该数据库的连接

## 应用ODBC访问数据库的两个条件的建立

关于第一个条件，大多数数据库管理系统在安装时均会自动装载其自身的ODBC驱动程序，而且很多专门的第三方的ODBC驱动程序可以从网上下载安装。

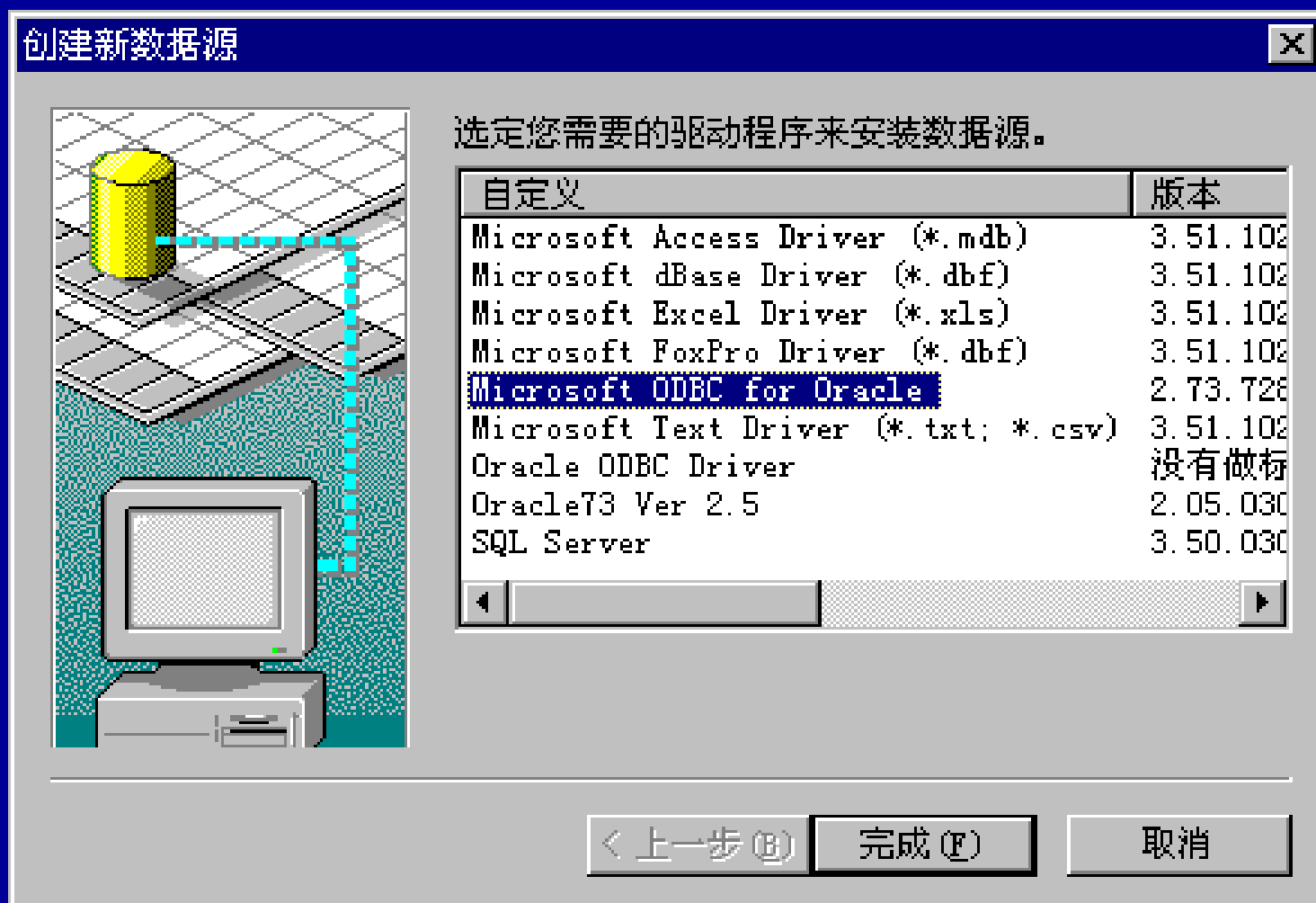
对于第二个条件，在Windows NT中，可通过控制面板中的ODBC数据源管理器创建（添加）及配置ODBC数据源。



# ODBC数据源管理器



# 创建新数据源窗口



# 设置Microsoft ODBC for Oracle DSN窗口



至此，我们就为myorasvr建立了一个名为Orasvr的ODBC连接。

用ODBC连接“OraSvr”定义数据库连接

对前面的例子，如果希望用PB6.0编写通过ODBC实现上例中对数据库访问的应用程序，只需修改Open事件程序代码中的数据库连接定义，其它程序代码不变。

## 用PB6.0的专用接口“073”定义的数据库连接

```
sqlca.dbms='073'  
sqlca.logid='stu'  
sqlca.logpass='stu'  
sqlca.servername='@myorasvr'  
connect ;  
if sqlca.sqlcode <> 0 then  
    MessageBox (“错误”，“数据库联接错  
        误，”+ “程序将退出： ~r~n” +  
        sqlca.sqlerrtext)  
    close(this)  
else  
    dw_employeeinfo.settransobject(sqlca)  
end if
```

## 用ODBC连接“Orasvr”定义的数据库连接

```
sqlca.dbms='odbc'  
sqlca.DbParm="Connectstring='DSN  
=orasvr;UID=stu;PWD=stu;' "
```

```
connect ;
```

```
if sqlca.sqlcode <> 0 then
```

```
    MessageBox ("错误", "数据库联接错  
    误,"+ "程序将退出: ~r~n" +  
    sqlca.sqlerrtext)
```

```
    close(this)
```

```
else
```

```
    dw_employeeinfo.settransobject(sqlca)
```

```
end if
```

## 注释:

与通过开发工具专用接口实现数据库访问不同的是，用ODBC访问数据库时，数据库连接定义中的dbms项为“ODBC”，而不是“073”。

也就是说，数据库访问请求不是直接通过SQL\*Net发给了服务器，而是先交给ODBC，ODBC再根据不同的数据源（DSN）采取相应的处理。

# 3 企业计算模式

3.1 单主机计算模式

3.2 客户机/服务器计算模式

3.3 浏览器/服务器计算模式

\*3.4 客户机/服务器模式的技术实现

 \*3.5 浏览器/服务器模式的技术实现



## 3.5 浏览器/服务器模式技术实现

在Windows环境中B/S模式中，应用（程序）及访问数据库的实现技术主要有：

- ①公共网关接口CGI
- ②Internet数据库连接器IDC
- ③先进数据库连接器ADC
- ④JAVA/JDBC语言编程
- ⑤Oracle WebServer
- ⑥动态服务器页面ASP

## ① 公共网关接口（CGI）

CGI（Common Gateway Interface）

是较早的B/S应用实现技术，适用于多种服务器平台，如UNIX、Windows等。

但用CGI开发B/S应用比较困难，占用服务器资源也比较多。

## ② Internet数据库连接器IDC(Internet Database Connector)

IDC 集成在 Internet Server API (ISAPI) 中，充分利用了DLL（动态链接库）技术，易扩充，但编程较CGI更为复杂，只适用于小型数据库系统。

另外，目前ISAPI还不具备跨平台的功能，只限于Windows NT平台。

### ③ 先进数据库连接器ADC (Advanced Database Connector)

ADC提供了ActiveX Control来访问数据库，其数据查询操作是在用户端的浏览器上执行的，为此ADC需要将服务器端数据库中部分记录先下载到用户端，故系统开销较大、响应慢，但对一些特别频繁且重复性的数据库查询操作具有很好的反应速度。

## ④ JAVA/JDBC语言编程

JAVA语言是一种面向对象、易移植、多线程控制的语言，通过JDBC去连接数据库。

用JAVA/JDBC编写的软件可移植性强，适用于多种操作系统，但其**执行效率和执行速度**还不太理想，不适于建立高效、高速的应用。

## ⑤ Oracle WebServer

这是一种典型的由数据库厂商提供与Web服务器集成在一起的中间件。

## ⑥ 动态服务器页面ASP

ASP是微软公司推出的Web应用开发技术，用于**动态网页**及其对**数据库访问**的应用开发，编程灵活、简洁，具有较高的性能，是目前访问Web数据库比较好的选择。

限于篇幅，**本节仅以ASP为例**，简单介绍B/S模式的实现技术。

## 3.5 浏览器/服务器模式技术实现

### 3.5.1 ASP工作原理

3.5.2 用ASP实现与前端浏览器的交互

3.5.3 通过ASP实现对数据库的访问

3.5.4 用ASP实现B/S模式应用实例



## 3.5.1 ASP工作原理

(1)ASP的含义

(2)ASP的工作方式

## (1) ASP 含义——Active

ASP 使用了Microsoft的Active X 控件技术。ActiveX控件技术是现在Microsoft软件的重要基础，它采用封装对象、程序调用对象的方式，简化编程，加强程序间合作。

ASP本身封装了一些基本组件和常用组件，有很多公司也开发了很多实用的第三方组件，只要在服务器上安装这些组件，就可以方便快速地建立B/S应用。

## (1) ASP 含义——Server

ASP 可以运行在服务器端，这样就不必担心浏览器是否支持ASP所使用的编程语言。

ASP的编程语言可以是 VB Script 和 JavaScript，其中 VB Script 是VB的一个简集，会VB的人可以很方便的快速上手。

## (1) ASP 含义——Pages

ASP返回给客户端的是标准的HTML页面，  
可以正常地在常用的浏览器中显示。

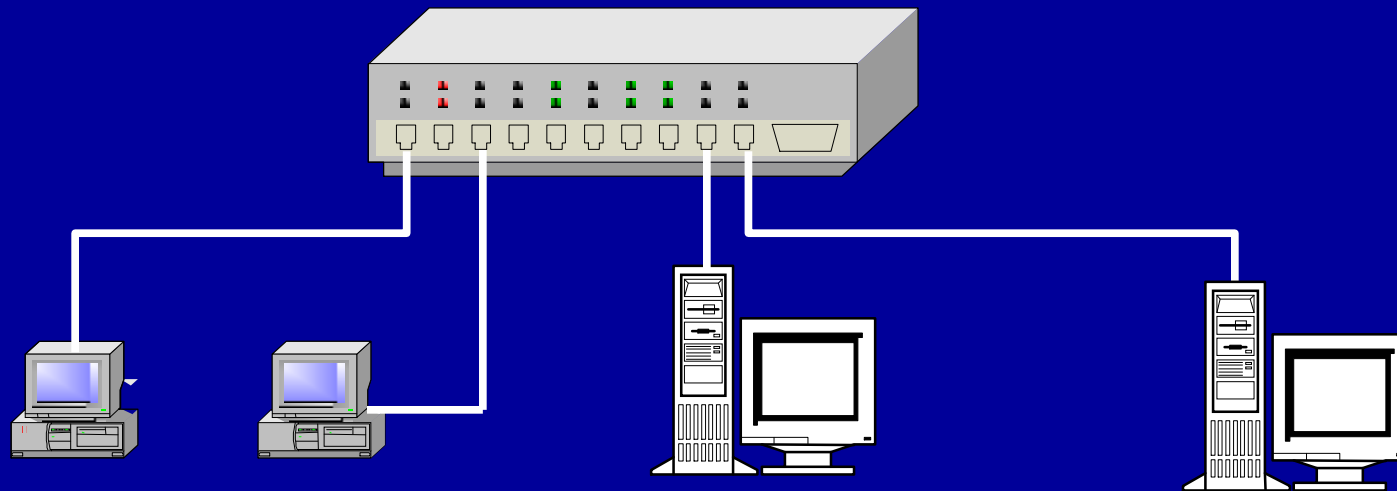
浏览者查看页面源文件时，看到的是  
ASP生成的HTML代码，而不是ASP程序代码，  
这样还可以防止别人抄袭程序。

## 常用的ASP运行环境

- Windows NT Server4.0+ IIS3.0及以上
- Windows NT Workstation4.0
  - + Peer Web Services 3.0及以上版本
- Windows 9x + PWS(Personal Web Server)

其中Windows NT Server上的IIS功能最强，提供了对ASP的全面支持，是创建高速、稳定B/S模式应用的最佳选择。

# 基于 Windows NT Server4.0+ IIS3.0 的 B/S 模式—实验环境



## 客户机

Windows98  
(TCP/IP协议)  
IE4.0

## Web Server

Windows NTServer4.0  
(TCP/IP协议, IP地址:  
192.168.63.30)  
IIS3.0

## DB Server

Windows NTServer4.0  
(TCP/IP协议, IP地址:  
192.168.63.12)  
Oracle7.3 for NT

## (2) ASP工作方式

B/S模式的技术实现包含两个方面：

- ① **浏览器** 与 **Web 服务器** 的交互
- ② **Web服务器** 对 **数据库服务器** 的访问

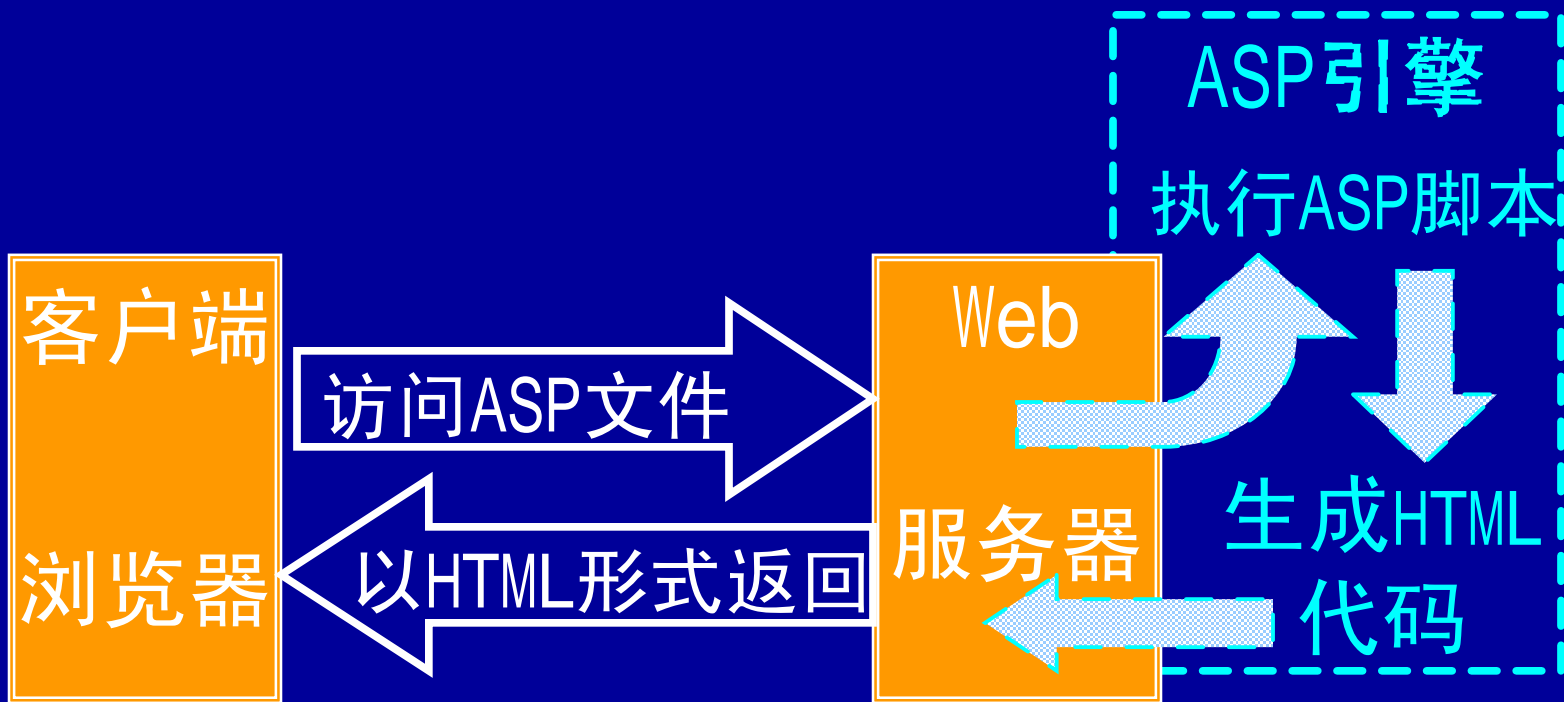


## ① 前端 浏览器 与 Web 服务器的交互

在本教程的第二章中，我们曾简单介绍用于网页编制的超文本标识语言（HTML），但用HTML编制的Web应用，其交互性较差，ASP弥补了标准HTML的这一不足。

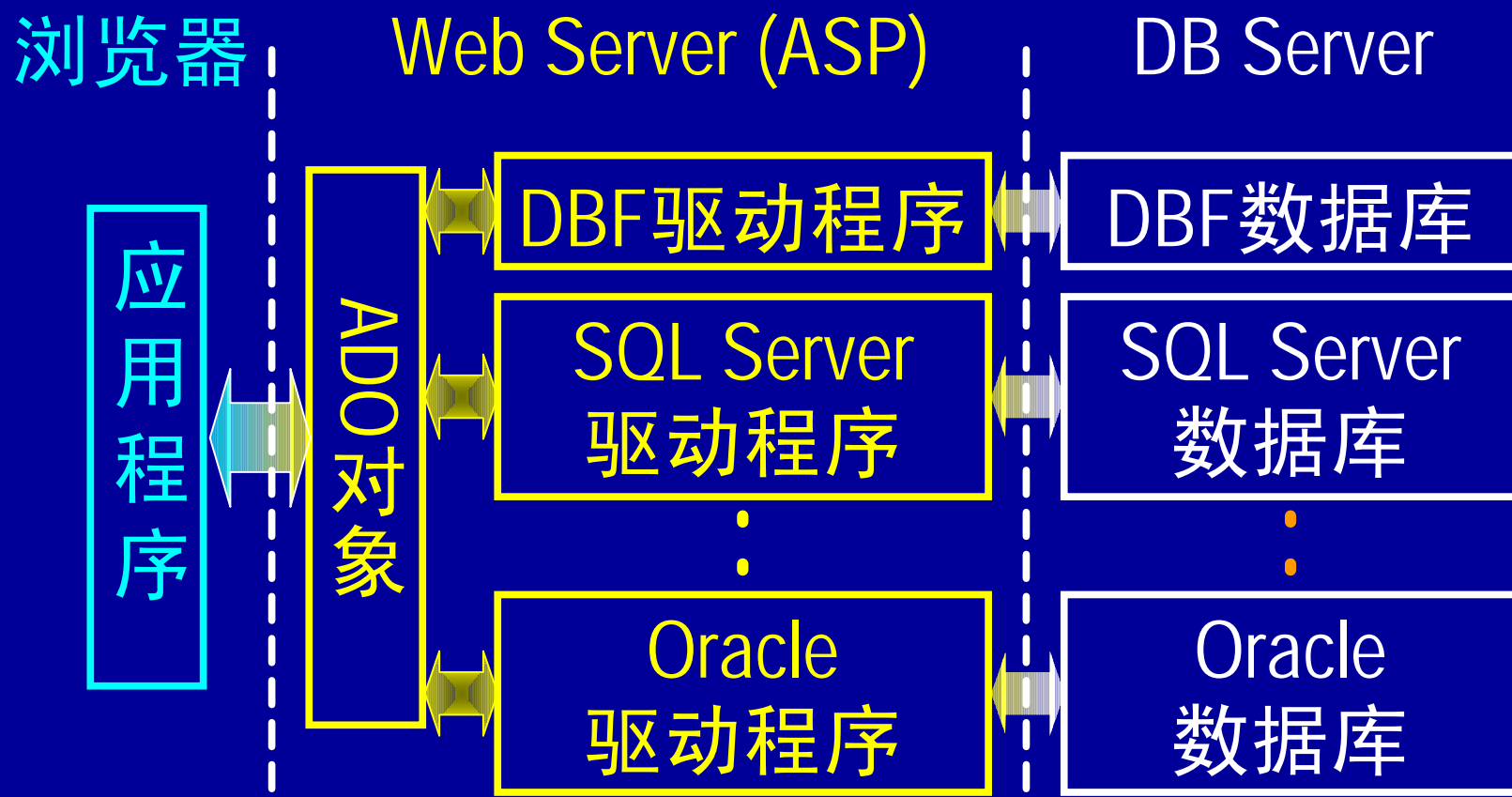


# 浏览器与Web服务器的交互中 ASP响应客户端(浏览器)工作示意图



## ② Web 服务器对数据库服务器的访问

应用程序通过 ADO对象及数据库驱动的驱动对数据库进行存取。



## ② Web 服务器对数据库服务器的访问(续)

ASP用来存取数据库的对象统称为ADO对象（Active Data Objects）。ADO主要有三种：

Connection 负责打开或连接数据库。

Recordset 负责存取数据表。

Command 负责对数据库执行动态查询命令和执行数据库的存贮过程。

只依靠 ADO 对象还是无法存取数据库的，还必须具有数据库存取的驱动程序。

## 3.5 浏览器/服务器模式技术实现

3.5.1 ASP工作原理

 3.5.2 用ASP实现与前端浏览器的交互

3.5.3 通过ASP实现对数据库的访问

3.5.4 用ASP实现B/S模式应用实例

## 3.5.2 用 ASP 实现与前端浏览器的交互

ASP 可根据用户输入的不同信息而应答不同的页面内容（生成动态页面），并将生成的HTML文件返回给前端浏览器。

假设：文件welcome.htm和welcome.asp放在Web Server（IP地址为：192.168.63.30）的aspdemo目录下。

当在浏览器地址栏里输入<http://192.168.63.30/aspdemo/welcome.htm>时，该文件将被执行（解释）显示在浏览器窗口中。

# 文件welcome.htm的内容

行	HTML 代码	注解
1	<html>	
2	<head>	
3	<title> 欢迎来到 ASP 世界 </title>	HTML 页面标题栏
4	</head>	
5	<body bgcolor = "#cccc00">	
6	<form action = "welcome.asp" method = "POST">	定义表单
7	<table width = "280" border = "0" align="center">	
8	<tr>	
9	<td colspan = "2" >您的名字:	
10	<input type = "text" name = "name">	定义文本输入框
11	</td>	
12	</tr>	
13	<tr>	
14	<td colspan="2"> 您的性别:	
15	<input type="radio" name="sex" value=1>男	定义单选钮
16	<input type="radio" name="sex" value=0>女	
17	</td>	
18	</tr>	
19	<tr>	
20	<td align = "center">	
21	<input type="submit" name="Submit" value=" 进入 ">	定义 submit(提交)按钮
22	</td>	
23	</tr>	
24	</table>	
25	</body>	

# 文件 Welcome.htm 的页面效果



## 文件 welcome.asp 的内容及执行过程

运行 Welcome.htm。其“进入”按钮把表单发送给 Welcome.asp。

WebServer 将 Welcome.asp 交给专门的 ASP 处理程序。

ASP 处理程序扫描 Welcome.asp，执行用 `<%` 和 `%>` 括起来的代码。

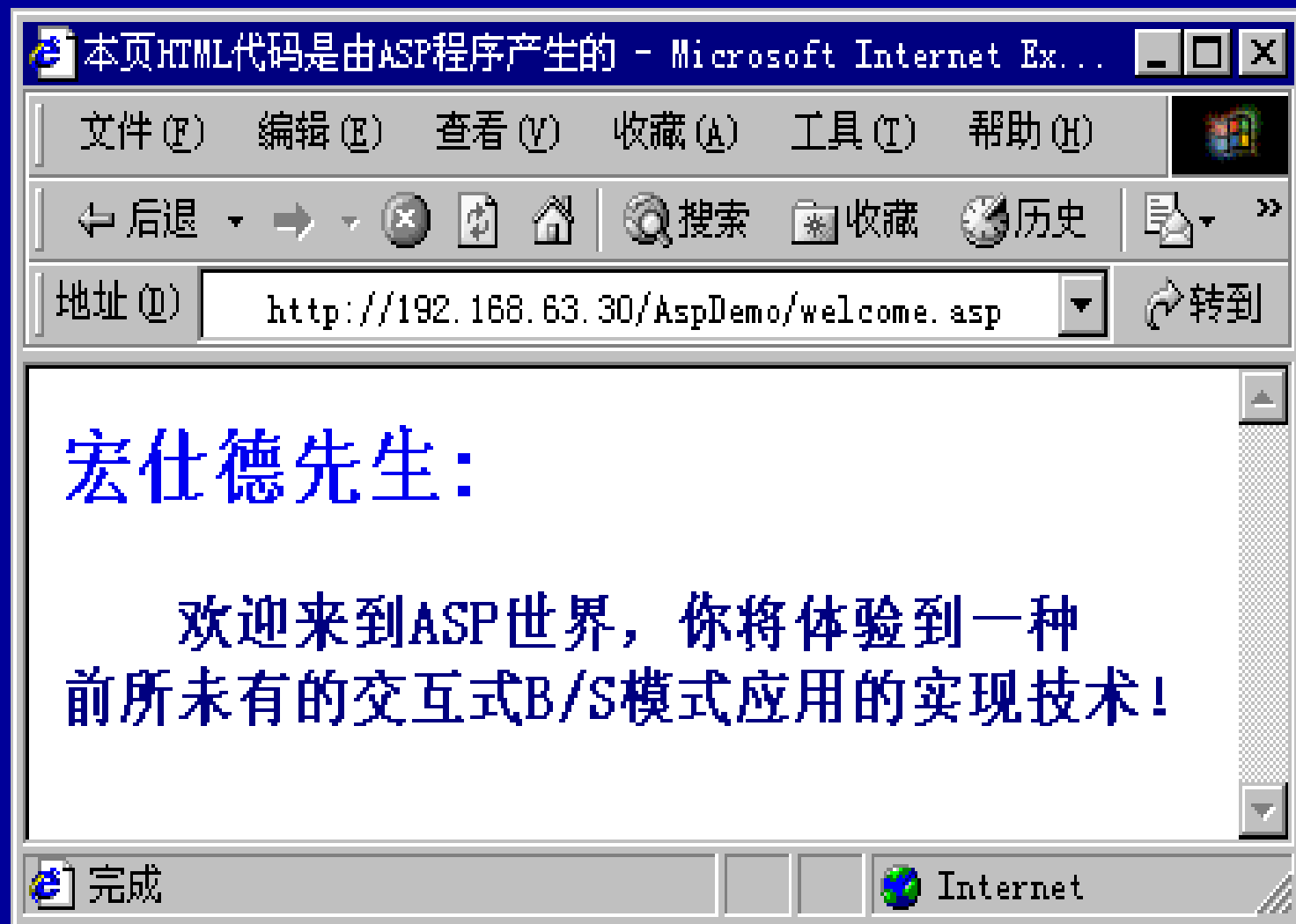
ASP 处理程序把执行结果传回给 WebServer。

WebServer 把结果传回到前端浏览器。

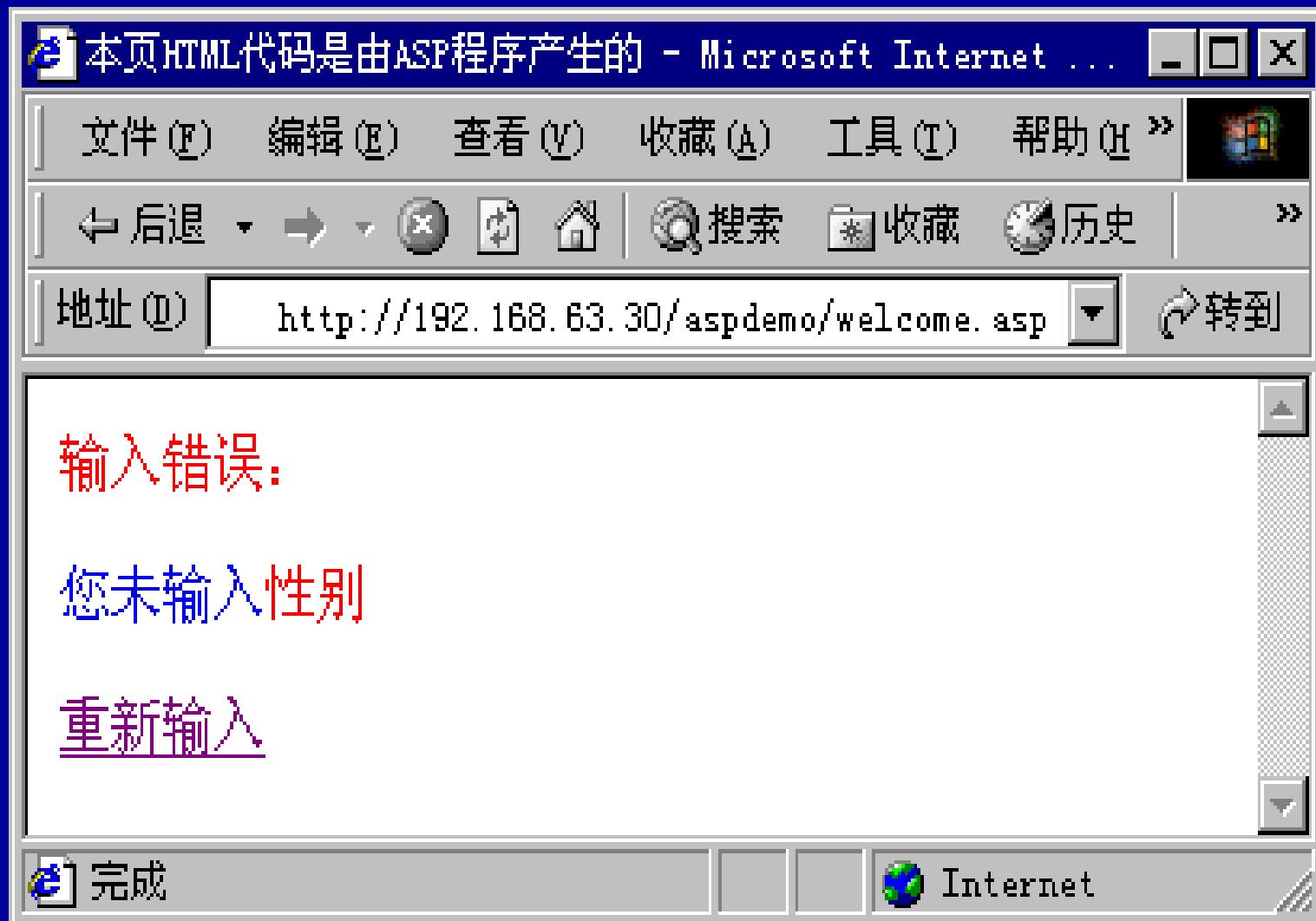
```
1 <html>
2 <head>
3   <title> 本页 HTML 代码是由 ASP 程序产生的 </title>
4 </head>
5 <body>
6 <%
7   Dim nErrFlag,sErrMsg,sName,nSex,sSex
8   NErrFlag = 0
9   nSex=1
10  sSex="先生"
11  If Request.form("name").Count= 0 Then nErrFlag=1
12  If Request.form("sex").Count = 0 Then nErrFlag=nErrFlag+2
13  If nErrFlag=0 Then
14    sName = Request.form("Name")
15    nSex=Request.form("sex")
16    If nSex= 0 Then  sSex= "女士"
17    Response.Write ("<p><b><font size= '5' color='$0000ff">")
18    Response.Write (sName + sSex + ":<br>")
19    Response.Write ("<font size='4' color= '$000008'><br>")
20    Response.Write ("  欢迎来到 ASP 世界,你将体验到一种<br>")
21    Response.Write ("前所未有的交互式 B/S 模式应用的实现技术!")
22    Response.Write ("</font></b></p>")
23  Else
24    sErrMsg="<font color='$ff0000'>输入错误: <br><br>" +
25      "<font color='$0000ff'>您未输入<font color='$ff0000'>"
26    If nErrFlag mod 2 =1 then sErrMsg = sErrMsg + "姓名"
27    If nErrFlag /2 >=1 then sErrMsg = sErrMsg + "性别"
28    Response.Write(sErrMsg)
29    Response.Write("<br><br><a href='\"welcome.htm\"'> ")
30    Response.Write(" 重新输入 </a>")
31  End If
32 %>
</body>
```



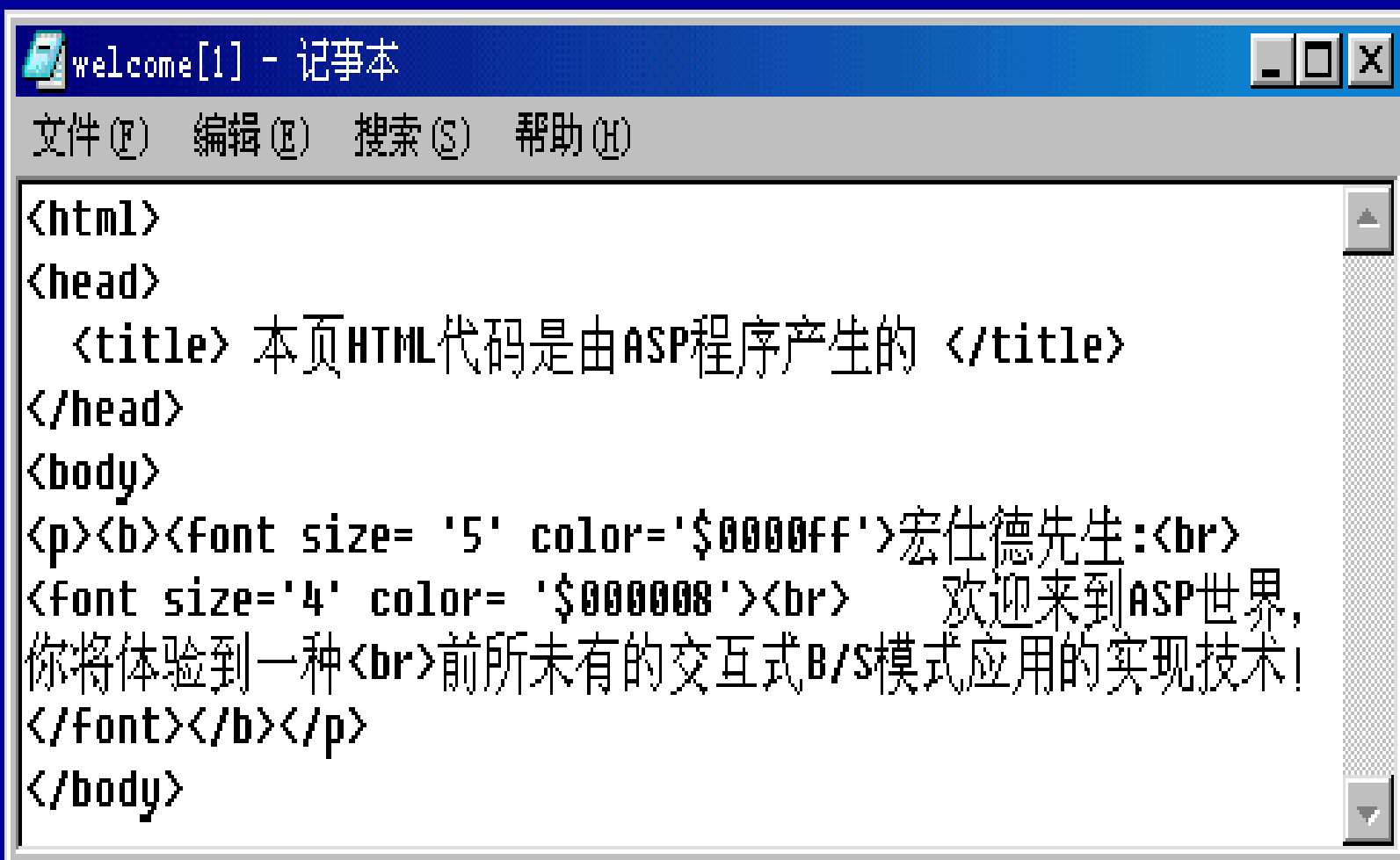
# welcome.asp页面效果 (1)



# welcome.asp页面效果 (2)



## 从客户端浏览器中看到的 welcome.asp源文件内容



```
welcome[1] - 记事本
文件(F) 编辑(E) 搜索(S) 帮助(H)
<html>
<head>
  <title> 本页HTML代码是由ASP程序产生的 </title>
</head>
<body>
<p><b><font size= '5' color='$0000ff'>宏仕德先生:<br>
<font size='4' color= '$000008'><br>  欢迎来到ASP世界,
你将体验到一种<br>前所未有的交互式B/S模式应用的实现技术!
</font></b></p>
</body>
```

## 观点:

此实例表明 WebServer 不但能接受客户端浏览器的输入，还能根据用户的输入信息生成不同（动态）的 HTML 代码返回给浏览器。显然，ASP 和 HTML 的结合使 B/S 模式应用的交互性、流程可控制性大大增强。

## 3.5 浏览器/服务器模式技术实现

3.5.1 ASP工作原理

3.5.2 用ASP实现与前端浏览器的交互

 3.5.3 通过ASP实现对数据库的访问

3.5.4 用ASP实现B/S模式应用实例

### 3.5.3 通过 ASP 实现对数据库的访问

用ASP访问Web数据库时，必须使用ADO组件。ADO是ASP内置的ActiveX服务器组件，通过在Web服务器上建立并设置ODBC和OLE DB可连接多种数据库(如SyBase、Oracle、SQL Server、Access、VFP等)。

#### ASP使用ADO组件访问远程数据库步骤:

- Step1 在WebServer上定义ODBC数据源DSN。
- Step2 创建数据库连接并打开数据库。
- Step3 执行指定的SQL命令并暂存结果。
- Step4 关闭数据库连接，释放有关资源。

## 通过 ASP 实现对数据库的访问例

以本章第四节（3.4.4）中员工信息查询应用为例（那里曾以C/S模式实现），下面我们将用ASP以B/S模式实现对数据库的访问。

这里假定：

- ①在WebServer上已创建了ODBC的数据源（DSN）orasvr。
- ②建立了数据库表employeeinfo，并对表添加了数据记录。
- ③程序文件adodemo.asp存放在WebServer的AspDemo目录下。

## 数据库表 employeeinfo

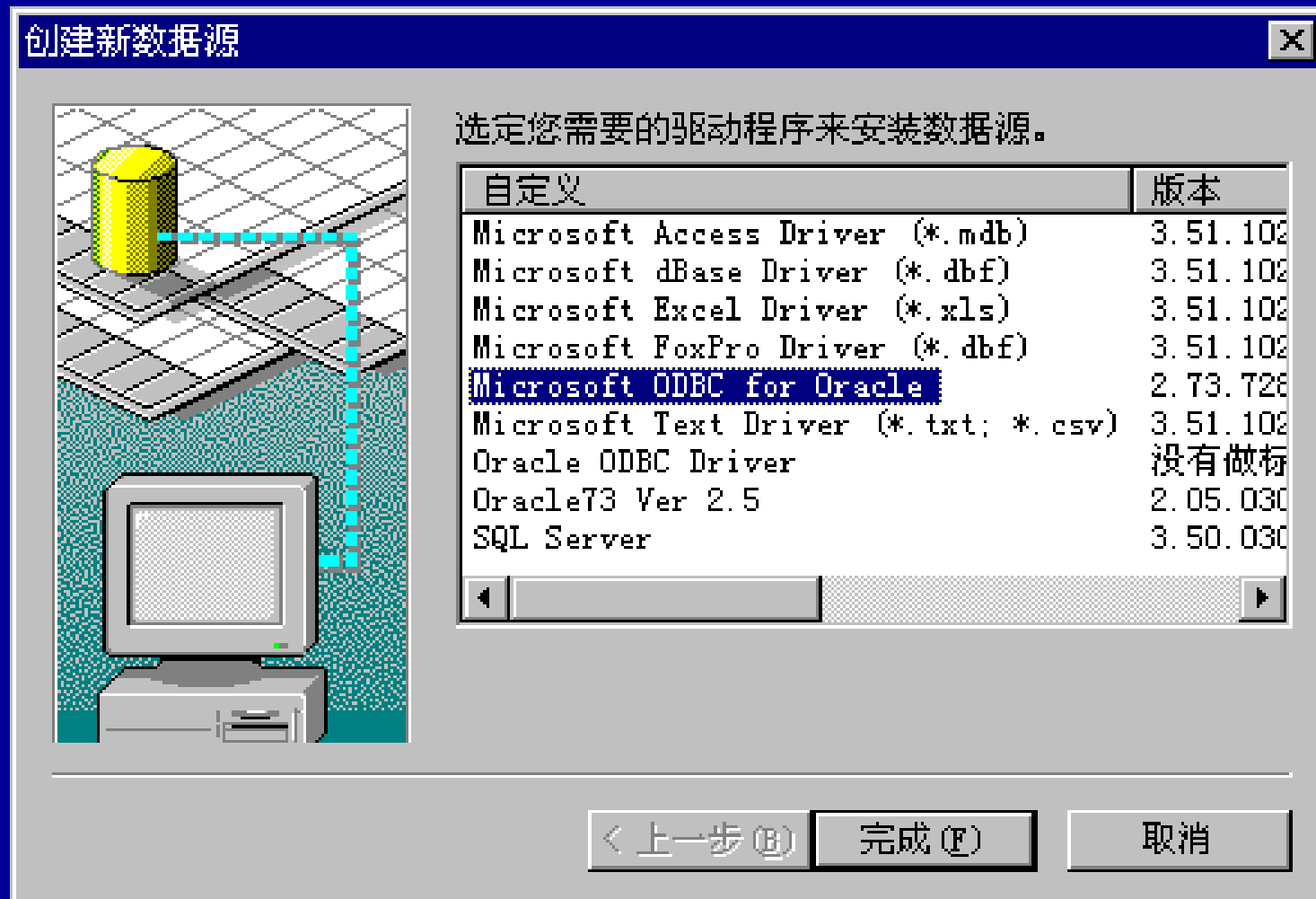
Emno	Empname	Birthdate	Sex	Salary
277052	王伟怀	19-JUL-69	M	1117.29
279809	周 梦	08-SEP-70	F	1028.58
456789	廖卫东	18-DEC-79	M	856.76
576868	谢 敏	22-SEP-76	F	962.73
807956	徐瑞忠	24-MAY-62	M	1168.36
809139	黄建伟	11-JAN-67	M	1038.91



# 创建ODBC的数据源 (1)



## 创建ODBC的数据源 (2)



# 创建ODBC的数据源（3）

## 一设置Microsoft ODBC for Oracle 数据源



Microsoft ODBC for Oracle 安装

数据源名称 (N):	Orasvr	确认
描述 (D):	Oracle Server for stu	取消
用户名称 (U):	stu	帮助
服务器 (S):	myorasvr	选择 >>

# adodemo.asp执行结果

<http://192.168.63.30/aspdemo/adodemo.asp>:

用ASP访问ORACLE数据库示例 - Microsoft Intern...

文件 (F) 编辑 (E) 查看 (V) 收藏 (A) 工具 (T) >>

后退 前进 刷新 主页 搜索 收藏 >>

地址 (D) 192.168.63.30/aspdemo/adodemo.asp 转到

月薪高于1000元的男性员工清单

EMPLOYEEENO	EMPNAME	BIRTHDATE	SEX	SALARY
277052	王伟怀	69-7-19	M	1117.29
807956	徐瑞忠	62-5-24	M	1168.36
809139	黄建伟	67-1-11	M	1038.91
891212	孙 凯	71-5-15	M	1069.94
892207	叶卓辛	64-8-12	M	1235.88
893448	江文济	65-7-28	M	1043.18

完成 Internet

# adodemo.asp 说明

用“ADODB.Connection”定义数据库连接实例 Conn,并用该实例打开数据库

创建游标实例,指定所使用的数据库连接和要执行的SQL语句

ASP将RecordSet中的各列的列名取出并生成表的第一行;

ASP对RecordSet中每一行依次取出其各列的值,生成整个表的具体内容

关闭游标和连接并释放其占用的资源

```
1 <%  
2 Dim conn, sql, r, f  
3 Set conn= Server.CreateObject("ADODB.Connection")  
4 Conn.ConnectionString="DSN=orasvr;UID=stu;PWD=stu"  
5 Conn.open  
6 Sql="select * from employeeinfo "  
7         &"where salary>1000 and sex='M'"  
8 Set r = Server.CreateObject("ADODB.Recordset")  
9 r.open sql,conn  
10 Response.write "<html>"  
11 Response.write "<head>"  
12 Response.write "<title>用 ASP 访问 ORACLE 数据库示例"  
13 Response.write "</title> </head>"  
14 Response.write "<body>"  
15 Response.write "月薪高于 1000 元的男性员工清单"  
16 Response.write "<table align='left' border='1'">  
17 Response.write "<tr>"  
18     For each f in r.fields  
19         response.write "<td>" + f.name + "</td>"  
20     Next  
21 Response.write "</tr>"  
22 While not r.eof  
23     response.write "<tr>"  
24     for each f in r.fields  
25         response.write "<td>" & f.value & "</td>"  
26     Next  
27     response.write "</tr>"  
28     r.movenext  
29 wend  
30 Response.write "</table></body>"  
31 r.close  
32 Set r= nothing  
33 Conn.close  
34 Set conn = nothing  
35 %>
```

## 3.5 浏览器/服务器模式技术实现

3.5.1 ASP工作原理

3.5.2 用ASP实现与前端浏览器的交互

3.5.3 通过ASP实现对数据库的访问

 3.5.4 用ASP实现B/S模式应用实例

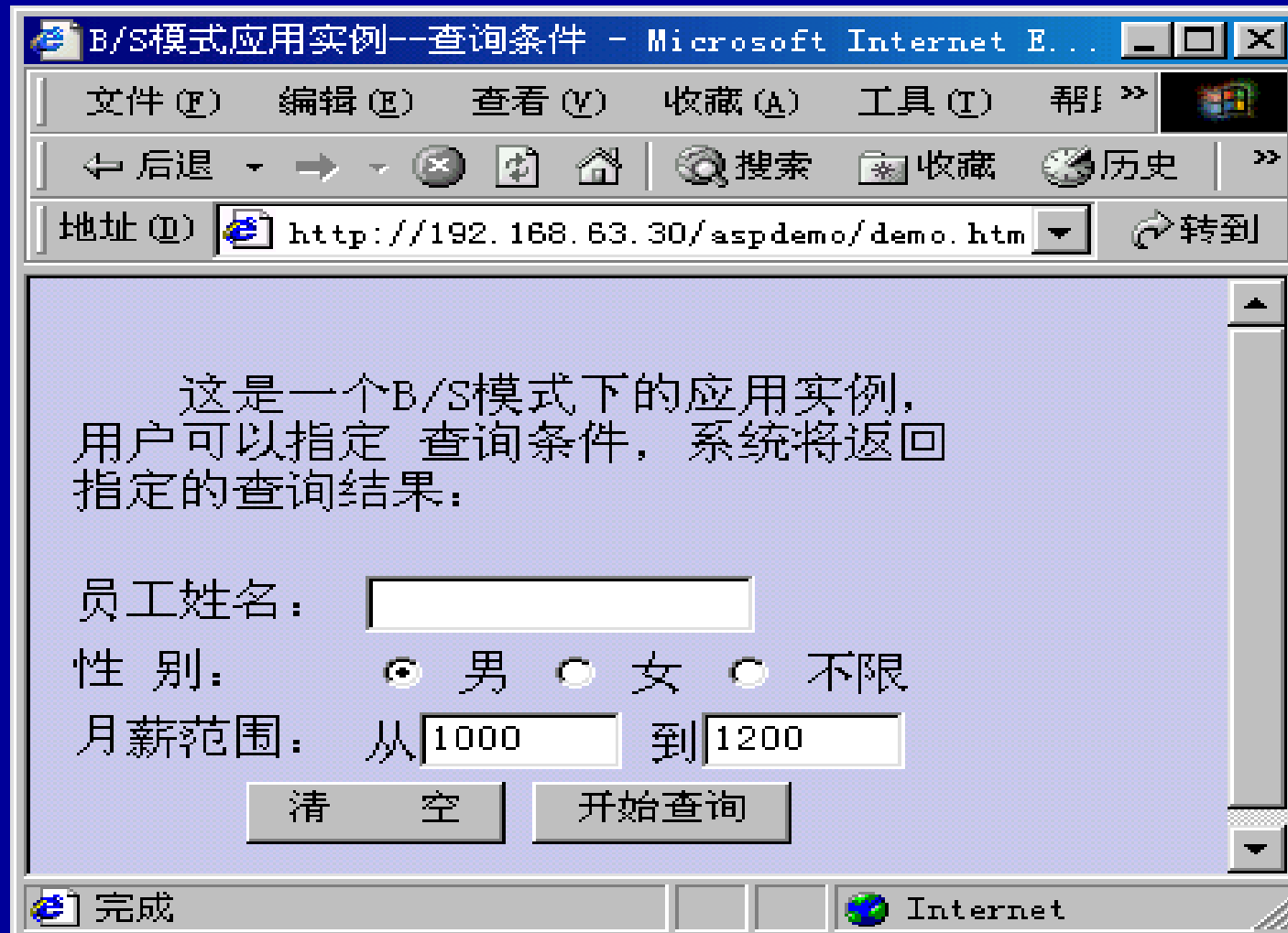
## 3.5.4 用ASP实现B/S模式应用实例

我们将前面所讲到的：“①用ASP实现与前端浏览器的交互；②通过ASP实现对数据库的访问。”两方面实现技术结合起来，在B/S模式下实现对员工信息的定制查询。用户可以指定查询条件，系统将返回指定的查询结果。

该应用的实现由两个文件（Demo.asp和Demo.htm）完成，其中涉及到的技术点均已在前文中涉及。

# Demo.htm页面执行效果

<http://192.168.63.30/aspdemo/Demo.htm>:





# Demo.asp执行效果



The screenshot shows a Microsoft Internet Explorer browser window. The title bar reads "B/S模式应用实例--查询结果 - Microsoft Internet E...". The address bar contains the URL "http://192.168.63.30/aspdemo/demo.asp". The main content area displays a table with the following data:

EMPLOYEEENO	EMPNAME	BIRTHDATE	SEX	SALARY
277052	王伟怀	69-7-19	M	1117.29
807956	徐瑞忠	62-5-24	M	1168.36
809139	黄建伟	67-1-11	M	1038.91
891212	孙 凯	71-5-15	M	1069.94
893448	江文济	65-7-28	M	1043.18

The status bar at the bottom shows "完成" (Completed) and "Internet".

## 其它:

在实际开发基于ASP的主页时，可把专门的网页编辑器与ASP的专业开发工具(如Visual InterDev等)结合应用:

利用专业网页编辑器所见即所得地编写主页丰富多彩的界面部分，用Visual InterDev编写服务器端的脚本语句，这种组合可以快速开发Web数据库的应用。

## 第3章小结

- (一) 什么是企业计算模式？
- (二) 三种企业计算模式：
  - (1) 单主机计算模式
  - (2) 客户/服务器 (C/S) 计算模式
  - (3) 浏览器/服务器 (B/S) 计算模式
- (三) B/S计算模式革命性地改变了计算机应用系统的面貌。
- (四) C/S模式和B/S模式的技术实现。