

21 世纪高等学校计算机教育实用规划教材

# DB2数据库实践基础教程

赵慧敏 主 编  
杨鑫华 邓 武 副主编



清华大学出版社

21 世纪高等学校计算机教育实用规划教材

# DB2数据库实践基础教程

赵慧敏 主编  
杨鑫华 邓 武 副主编

清华大学出版社  
北京



## 内 容 简 介

本书是《数据库原理与 DB2 应用教程》(杨鑫华、丁传华、赵慧敏编著,北京,清华大学出版社,2007 年)的配套教材,着重于对实践操作的介绍,包括 DB2 实验环境设置,数据库和表空间的创建方法,数据库对象,如表、视图、索引、触发器等创建方法,移动与操纵数据的方法,数据查询、数据库恢复的方法,权限管理的方法等。

本书可作为计算机、软件工程、电子商务、信息管理等本科生数据库实验课程的教材;也可作为参加 IBM DB2 专业认证考试的辅导教材,以及有意于从事数据库应用开发人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

## 图书在版编目(CIP)数据

DB2 数据库实践基础教程/赵慧敏主编. —北京:清华大学出版社,2009.10

(21 世纪高等学校计算机教育实用规划教材)

ISBN 978-7-302-21221-8

I. D… II. 赵… III. 关系数据库—数据库管理系统,DB2—高等学校—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2009)第 174320 号

责任编辑:闫红梅 赵晓宁

责任校对:李建庄

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185×260 印 张:7.75 字 数:186 千字

版 次:2009 年 10 月第 1 版 印 次:2009 年 10 月第 1 次印刷

印 数:1~3000

定 价:16.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:034438-01

# 目 录

---

实验 1 DB2 实验环境设置 .....	1
实验 2 创建数据库/表空间 .....	13
实验 3 创建数据库对象 .....	34
实验 4 移动与操纵数据 .....	55
实验 5 数据操纵——查询 .....	81
实验 6 数据库恢复 .....	90
实验 7 管理权限 .....	101
参考文献 .....	114





# 出版说明

---

随着我国高等教育规模的扩大以及产业结构调整的不断深入,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和帮助下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设仍然把重点放在公共基础课和

专业基础课的教材建设上；特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本，合理配套。基础课和专业基础课教材要配套，同一门课程可以有多本具有不同内容特点的教材。处理好教材统一性与多样化，基本教材与辅助教材，教学参考书，文字教材与软件教材的关系，实现教材系列资源配套。

(5) 依靠专家，择优选。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主编。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会

联系人：丁岭 [dingl@tup.tsinghua.edu.cn](mailto:dingl@tup.tsinghua.edu.cn)



# 前 言

目前在国内外软件人才市场上,对具有大型数据库管理、开发经验和技能的人才需求非常大,用人单位尤其注重其数据库应用的实践能力。DB2 通用数据库是 IBM 公司的旗舰产品,也是关系数据库的代表,其性能高、产品架构全面合理。目前市场上关于 DB2 的资料大多是英文的,少有的中文资料内容安排上也与教学需求不符。

针对这一现状,我们在数据库实践课程教学中,摸索出对学生有利的教学安排和课程设置,编写了本书。按照课程的进度和学生的接受能力,本着从易到难、从整体到局部、从具体到抽象的原则安排学生的实践教学,使学生在学习了理论知识后,能在实践中充分地消化和吸收理论知识,使其能更加满足实用型人才培养的目标。

本书是《数据库原理与 DB2 应用教程》(杨鑫华、丁传华、赵慧敏编著,北京,清华大学出版社,2007 年)的配套教材,全书共分为 7 个部分。实验 1 介绍 DB2 实验环境设置;实验 2 介绍数据库和表空间的创建方法;实验 3 介绍数据库对象,如表、视图、索引、触发器等创建方法;实验 4 介绍移动与操纵数据的方法;实验 5 介绍数据查询;实验 6 介绍数据库恢复的方法;实验 7 介绍权限管理的方法。本书中所使用到的脚本文件请到清华大学出版社网站([www.tup.tsinghua.edu.cn](http://www.tup.tsinghua.edu.cn))下载。

本书由大连交通大学赵慧敏任主编,杨鑫华、邓武任副主编,由多年从事数据库课程教学经验的一线教师编写。其中,实验 1 和实验 6 由李雪梅编写;实验 2 由赵慧敏编写;实验 3 由牛一捷编写;实验 4 由李媛媛编写;实验 5 由邓武编写;实验 7 由杨鑫华编写。

在教材编写过程中,得到了 IBM 大中华区大学合作部及程毓佳女士的大力支持和帮助,在此表示诚挚的谢意。此外,在编写过程中,编者参阅了大量的著作、教材,谨对这些著作、教材的编著者表示衷心感谢。

计算机技术日新月异,由于我们水平有限,虽经过努力,但其中错误在所难免。对教材中出现疏漏、不足之处,恳请读者朋友们批评指正,并与我们联系,联系信箱:hm\_zhao1977@126.com。

编 者

2009 年 8 月

# 实验 1

## DB2 实验环境设置

### 实验任务

- 建立拥有系统管理权限(SYSADM)的新用户(user)和组(adml);
- 建立新的 DB2 实例(inst1);
- 增加一些新的系统变量,并且更新 DBM CFG 以反映刚创建的新系统管理用户组。

### 实验内容

#### 1. 安装 DB2 9.1 数据库管理系统

本实验使用的 DB2 9.1 版本要求浏览器版本在 IE 5.0 以上,安装界面如图 1-1 所示,单击“安装新产品”按钮即可安装。

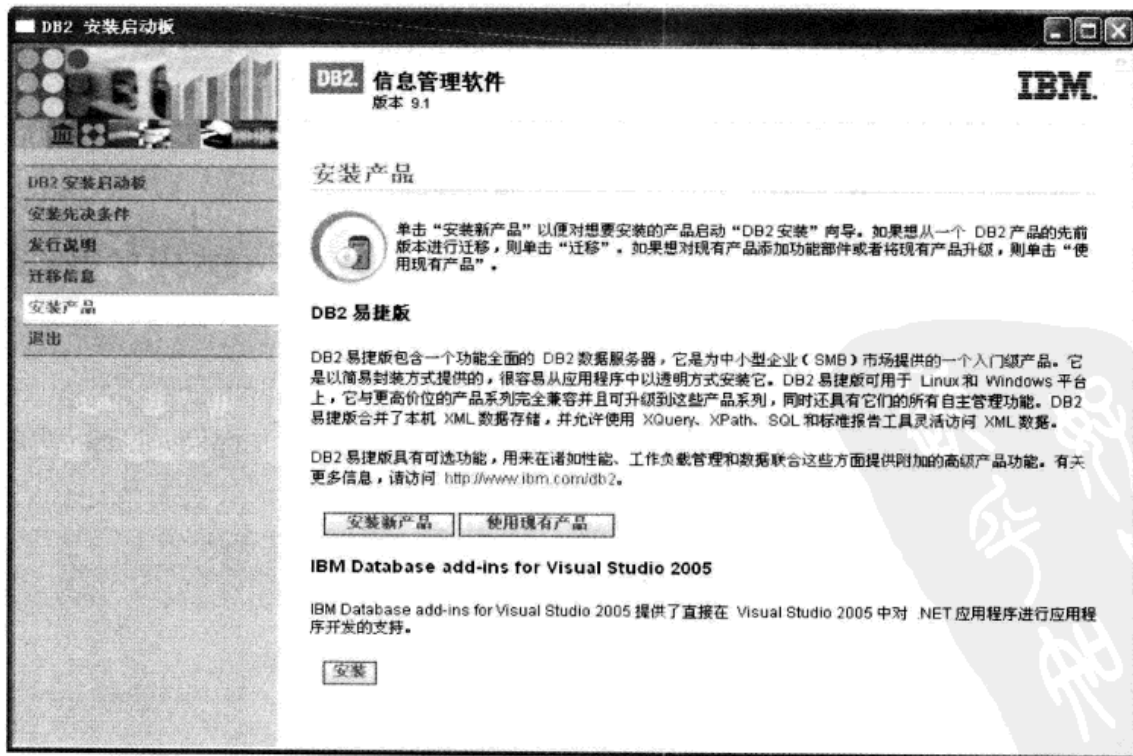


图 1-1 “DB2 安装启动板”窗口



## 2. 建立新用户

在 DB2 数据库管理系统中, DB2 实例是一个逻辑的数据库服务器环境。DB2 数据库就将创建在数据库服务器的 DB2 实例中。在一台计算机上可以同时创建多个实例, 每个实例都可以单独作为一个 DB2 服务器对外提供服务。例如, 在同一台计算机上, 可以运行一个生产环境实例和一个测试环境实例。

每一个实例都有一个管理员组。该管理员组必须定义在一个称为数据库管理器配置文件的实例配置文件中, 如何创建用户 ID 和用户组, 依赖于具体的操作系统环境。因此, 首先要在操作系统中建立新的用户和用户组用以管理数据库的实例。

这里以 Windows 操作系统为例。Windows 是一个支持多用户、多任务的操作系统, 不同的用户在访问计算机时, 将会有不同的权限。同时, 对用户权限的设置也是基于用户和进程而言的。在 Windows 中, 用户被分成许多组, 组和组之间都有不同的权限, 并且一个组的用户和用户之间也可以有不同的权限。下面介绍 3 个常见的用户组。

### 1) Users

普通用户组, 这个组的用户无法进行有意或无意的改动。因此, 用户可以运行经过验证的应用程序, 但不能运行大多数旧版应用程序。Users 组是最安全的组, 因为分配给该组的默认权限不允许成员修改操作系统的设置或用户资料。Users 组提供了一个最安全的程序运行环境。在经过 NTFS 格式化的卷上, 默认安全设置旨在禁止该组的成员危及操作系统和已安装程序的完整性。用户不能修改系统注册表设置、操作系统文件或程序文件。Users 可以创建本地组, 但只能修改自己创建的本地组。Users 可以关闭工作站, 但不能关闭服务器。

### 2) Power Users

高级用户组, Power Users 可以执行除了为 Administrators 组保留的任务外的其他任何操作系统任务。分配给 Power Users 组的默认权限允许 Power Users 组的成员修改整个计算机的设置。但 Power Users 不具有将自己添加到 Administrators 组的权限。在权限设置中, 这个组的权限是仅次于 Administrators 的。

### 3) Administrators

管理员组, 默认情况下, Administrators 组中的用户对计算机/域有不受限制的完全访问权。分配给该组的默认权限允许对整个系统进行完全控制。一般来说, 应该把系统管理员或者与其有着同样权限的用户设置为该组的成员。

本实验中要创建新的用户组 adm1, 并创建用户 user 使其隶属于用户组 adm1。右击“我的电脑”图标, 在弹出的菜单中选择“管理”选项, 在“本地用户和组”中根据提示按以下要求创建新用户组及用户, 如图 1-2~图 1-4 所示。

用户名: user  
全名: user Instance SysAdmin  
描述: System Administrator for INST1 Instance  
密码: user  
用户组: Administrators

组名: adm1  
描述: inst1 Instance SysAdmin Group  
成员: db2admin; user

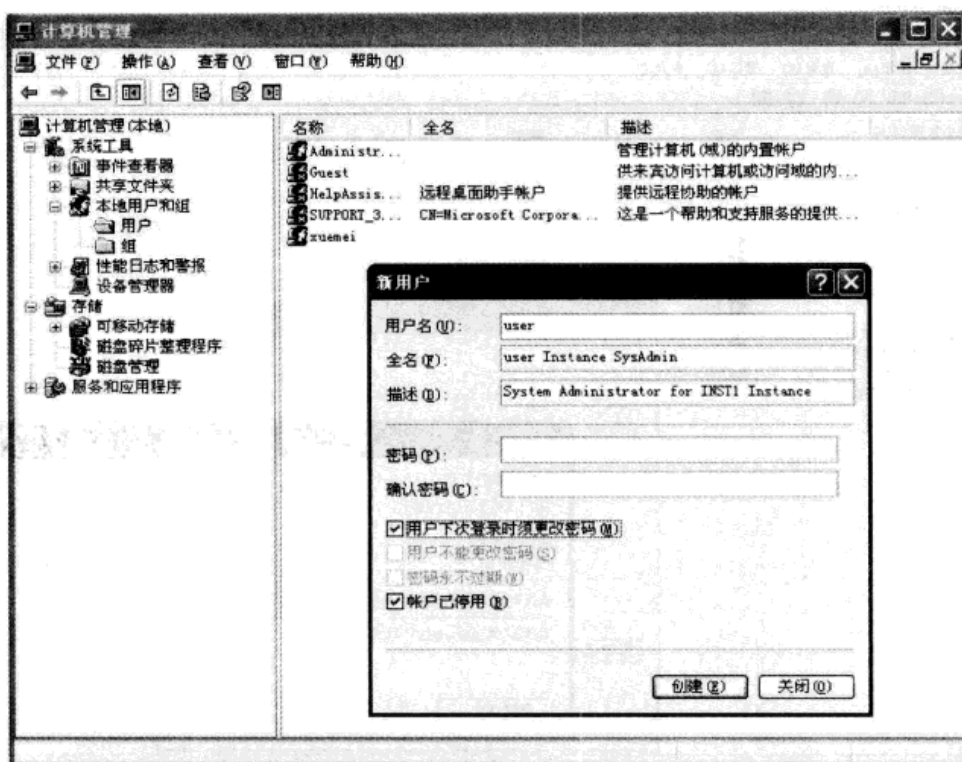


图 1-2 建立新用户 user



图 1-3 添加新用户到 administrators 组中

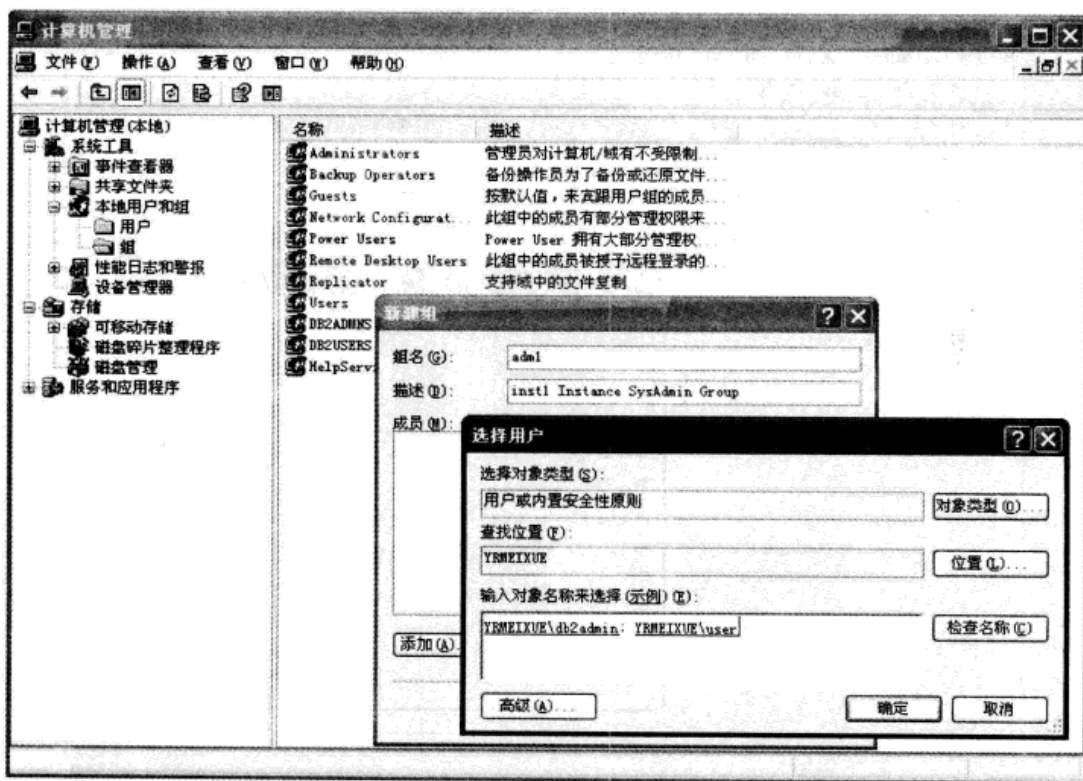


图 1-4 创建新用户组

退出当前用户登录的系统,然后以用户 user 的身份登录到操作系统。在以后的实验中,当要求登录到系统中时,除特别说明外都是要求以 user 身份登录。

### 3. 创建实例

DB2 安装程序在安装时会创建一个默认的实例,这是推荐使用的创建实例的方法。当然,实例也可以在安装过程结束后,根据需要被创建或删除。

(1) 在 DB2 的 Windows 版本安装完成后,系统会自动建立一个默认名为 DB2 的实例。要验证该 DB2 实例确实存在,可以打开 DB2 命令窗口并输入 db2ilist 命令。

选择“开始”→“程序”→IBM DB2→“DB2COPY1(缺省值)”→“命令行工具”→“命令窗口”选项,如图 1-5 所示。输入 db2ilist,db2ilist 命令显示 DB2 是该主机的一个实例。

(2) 出于实验环境的需要,要建立另一个称为 inst1 的新实例。在命令窗口中输入如下命令:

```
db2icrt inst1
```

(3) 为得到当前实例列表,再次输入 db2ilist 命令,将会发现 inst1 实例已成功创建,如图 1-6 所示。

(4) 如果需要删除一个实例则用: db2idrop 实例名。

(5) 输入 exit 退出 DB2 命令窗口。

### 4. 设置环境

#### 1) 设置环境变量

(1) 检查当前 DB2 环境变量,并且设置刚刚建立的 inst1 实例为系统默认实例。



图 1-5 DB2 命令窗口路径

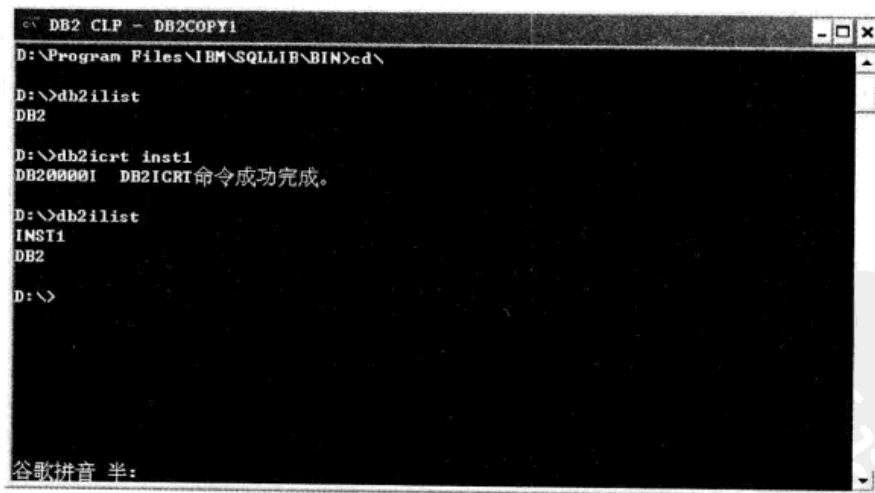


图 1-6 创建实例 inst1

打开控制面板的系统属性窗口(或右击“我的电脑”，选择“属性”选项)，选择“高级”标签页，然后单击“环境变量”按钮；弹出的“环境变量”对话框被分为上下两部分，上半部分所定义的用户变量只影响当前用户，而系统变量则影响每一个用户。查看系统变量中 DB2INSTANCE 所对应的值，默认情况下为 DB2，现将其改为 inst1。双击系统变量列表框中的 DB2INSTANCE 变量，在“编辑系统变量”对话框的变量值一栏中输入 inst1，然后单击



“确定”按钮,如图 1-7 所示。



图 1-7 设置 DB2INSTANCE 环境变量

双击系统变量列表中的 PATH 选项,可见 PATH 系统变量值中包含路径 X:\SQLLIB\bin,因此 X:\SQLLIB\bin 下的可执行文件可直接在命令提示符下启动。

(2) 启动 DB2DAS-DB2DAS00 和 DB2-DB2COPY1-INST1 服务。

右击“我的电脑”图标,在弹出的快捷菜单中选择“管理”选项,打开管理窗口在左侧目录树中选择“服务和应用程序”中的“服务”选项,则右侧服务列表中列出系统中的服务及其状态信息。

右击 DB2DAS-DB2DAS00 选项,在弹出的菜单中选择“属性”选项,在 DB2DAS-DB2DAS00 属性对话框中(如图 1-8 所示),将“常规”选项卡中的“启动类型”设为自动;然后单击“启动”按钮,启动 DB2DAS-DB2DAS00 服务(也可以通过在“命令窗口”中输入 db2admin start 和 db2admin stop 命令来启动和关闭该服务);单击“登录”标签,将“登录身份”设置为“本地系统账户”。

用同样的方法设置 DB2-DB2COPY1-INST1 服务的启动类型为自动,并检查该服务是否已启动,如果尚未启动则单击“启动”按钮启动该服务(也可以在 DB2 控制中心中右击该实例名选择“启动”来启动这个服务)。

## 2) 检查 DB2 环境变量

选择“开始”→“程序”→“附件”→命令提示符;输入 set;more。

检查 DB2 环境变量是否设置正确,假如不对重复上面必要的步骤予以纠正。

## 3) 检查 DB2 注册表变量

在命令提示符窗口中输入 db2set -all 命令来检查 DB2 注册表变量,DB2ADMIN SERVER 的值为 DB2DAS00,DB2SYSTEM 的值与主机名相同,如图 1-9 所示。



图 1-8 启动 DB2DAS-DB2DAS00 服务

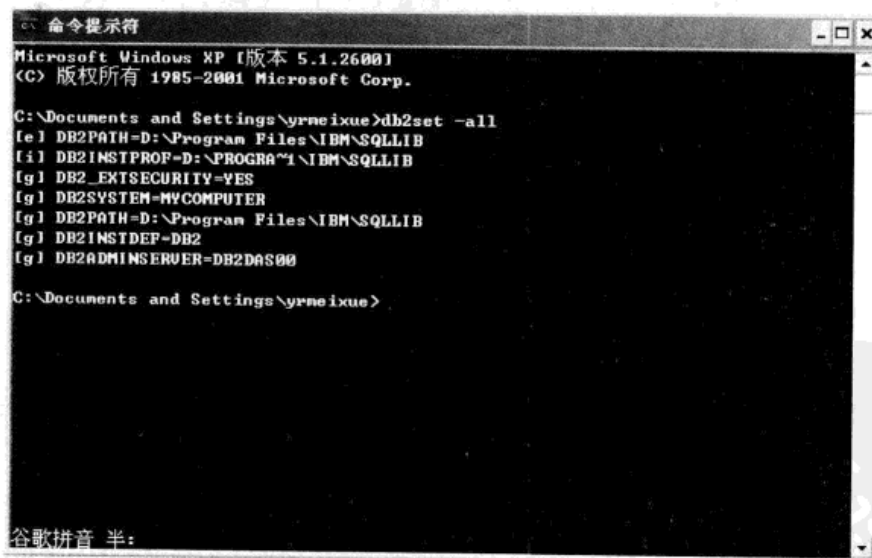


图 1-9 检查 DB2 注册表变量

**注意：**使用带-all 选项的 db2set 命令可以查看系统中设置的所有配置文件注册表变量。

#### 4) 设置 DB2COMM 变量

为了对实例 inst1 和全局注册表变量提供 TCP/IP 协议的支持,输入如下命令设置 DB2COMM 变量:

```
db2set DB2COMM=TCPIP -i inst1
```

```
db2set DB2COMM=TCPIP -g
```

**注意：**设置当前实例的一个参数时输入格式为 `db2set parameter=value`

为特定实例设置参数值时输入格式为 `db2set parameter=value -i instance_name`

设置一个全局级的注册表变量时输入格式为 `db2set parameter=value -g`

查看能够被设置的所有配置文件注册表变量,可输入 `db2set -lr`

再次查看注册表变量。输入 `db2set -all`,注意在输出列表中实例级的注册表变量旁边标以[i],全局级的注册表变量旁边标以[g],如图 1-10 所示。

```

C:\>db2set db2comm=tcPIP -i inst1
C:\>db2set db2comm=tcPIP -g
C:\>db2set -all
[i] DB2PATH=D:\Program Files\IBM\SQLLIB
[i] DB2INSTPROF=D:\PROGRAM~1\IBM\SQLLIB
[i] DB2COMM=tcPIP
[g] DB2_EXTSECURITY=YES
[g] DB2SYSTEM=MYCOMPUTER
[g] DB2PATH=D:\Program Files\IBM\SQLLIB
[g] DB2INSTDEF=DB2
[g] DB2COMM=tcPIP
[g] DB2ADMINSERVER=DB2DAS00
C:\>

```

图 1-10 设置 DB2COMM 变量值

#### 5) 完成

重启操作系统,并以用户 user 登录(用户密码按照先前所设都为 user)。

#### 5. 使用 DB2 控制中心来查看和设置数据库管理器配置参数

选择“开始”→“程序”→IBM DB2→DB2COPY1(默认值)→“一般管理工具”→“控制中心”选项,如图 1-11 所示。

(1) 在控制中心窗口中选择“所有系统”左边的“+”号,将看到系统中的主机名称,单击本机名称左边的“+”号即列出本机上的实例。

(2) 选择“实例”左边的“+”号,如果“实例”下没有列出 INST1,则右击“实例”项,在弹出菜单中选择“添加”选项,在“添加实例”窗口中的“远程实例名”栏输入 INST1,单击“应用”按钮,再单击“取消”按钮。

(3) 查看数据库管理器配置文件。为了查看或更改数据库管理器配置文件,右击控制中心里的 INST1 实例,在弹出菜单中选择“配置参数”项,如图 1-12 所示。

(4) 根据“DBM 配置”窗口,记录下列值。

① “管理”栏下的 AUTHENTICATION 参数。

单击 AUTHENTICATION 参数,选择该参数所对应的值,并单击值右边的省略号按钮,在弹出的“更改 DBM 配置参数-AUTHENTICATION”对话框中列出 5 个值:服务器、

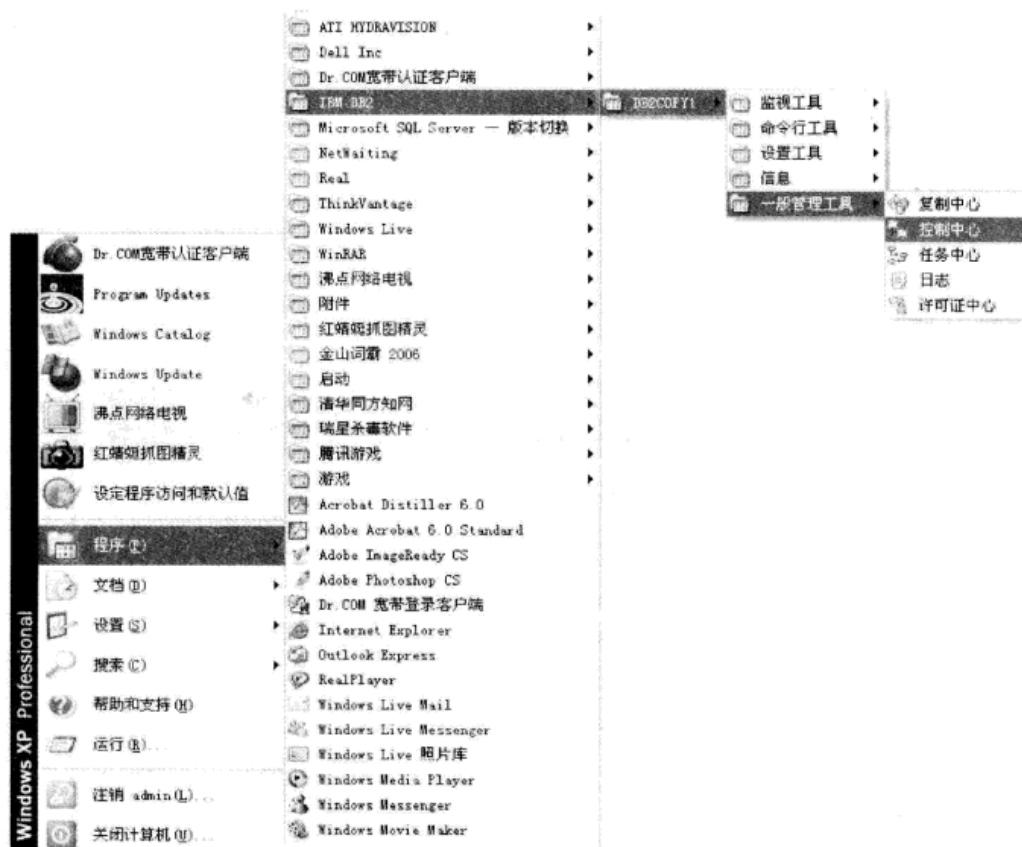


图 1-11 控制中心路径

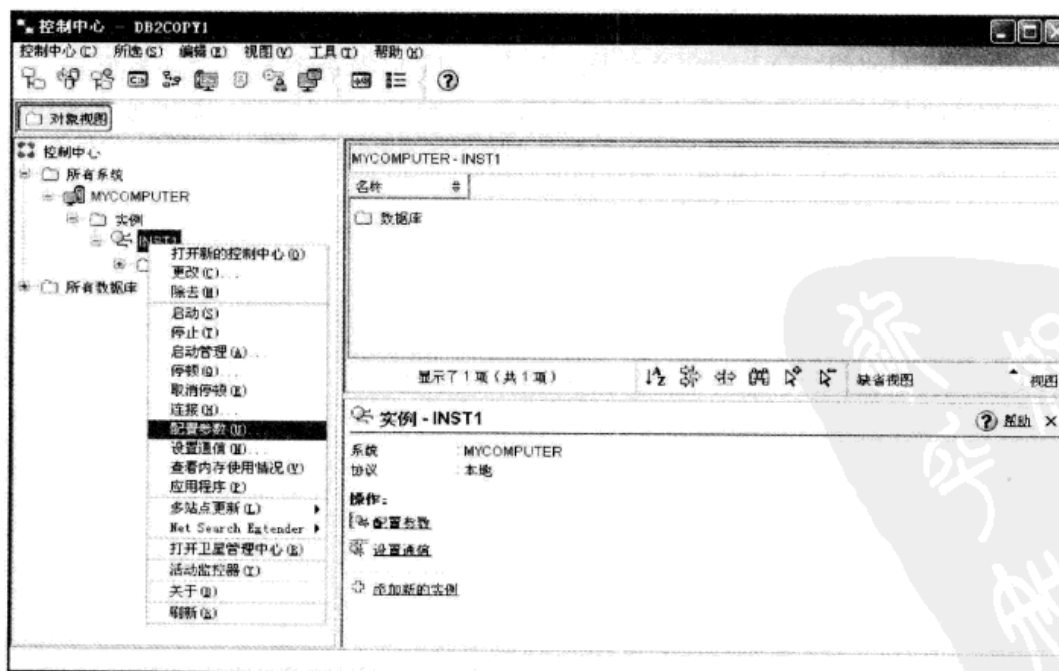


图 1-12 查看 INST1 实例配置参数



客户机、服务器加密、Kerberos 和 Kerberos 加密,如图 1-13 所示。其中“服务器”选项指验证过程将在服务器上进行,这是 DB2 默认的身份鉴别方式。单击“取消”按钮关闭“更改 DBM 配置参数-AUTHENTICATION”对话框。

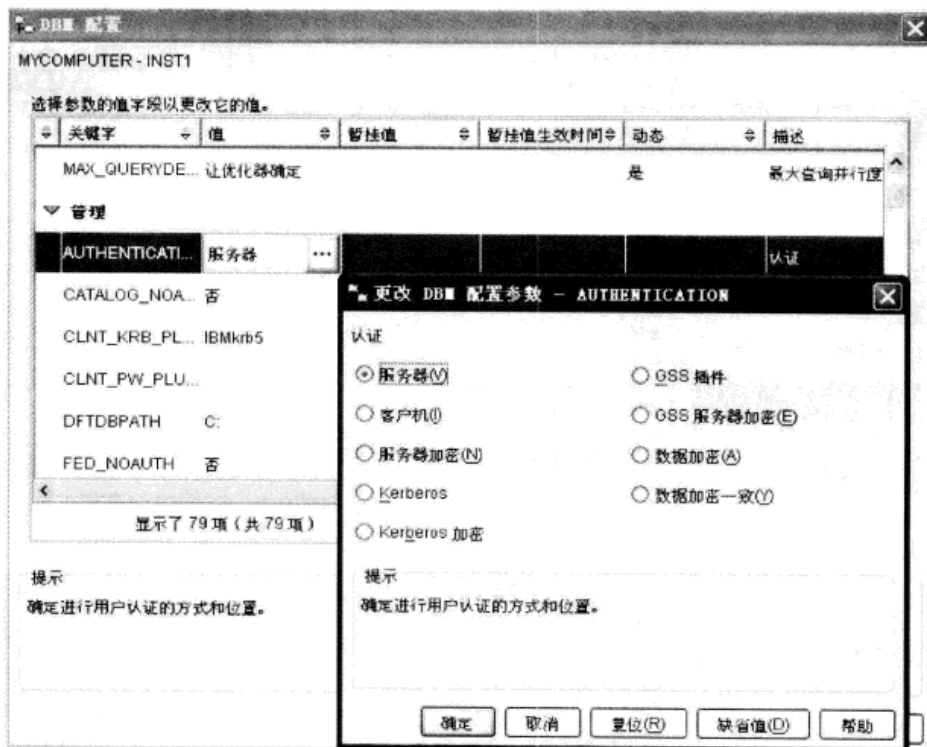


图 1-13 更改 DBM 配置参数

**注意：**AUTHENTICATION 参数指定了 DB2 UDB 服务器端的验证类型。

在服务器端,验证类型被定义在数据库管理器配置文件中。该文件用于配置特定的 DB2 实例,其包含的参数值将会影响到该实例中所有的数据库。因此,如果为某个实例指定了验证类型,那么实例中的所有数据库都会使用这种验证类型,DB2 系统也会基于这种验证类型对所有连接到该实例中的数据库的用户进行验证。可以在服务器端指定的验证类型包括:

- **SERVER。**该验证类型强迫所有的验证过程都在服务器端进行,因此,在对实例或数据库进行连接时,必须提供用户名和密码。DB2 系统在将密码传递给服务器的过程中不对密码进行加密。
- **SERVER\_ENCRYPT。**该验证类型与 SERVER 相似,所有的验证过程都在服务器端进行。但是,所有的密码在传递给服务器之前都将在客户端进行加密。在密码传到服务器后,密码将被解密并进行验证。但是,如果在连接的同时要求更改现有密码,DB2 系统在将新密码传递给服务器之前不对新密码进行加密。
- **CLIENT。**该验证类型允许在客户端代表服务器对用户进行验证,但不能保证验证过程肯定能够在客户端进行。
- **KERBEROS。**该验证类型既可以设置在客户端,也可以设置在服务器端,但是要求 DB2 系统所驻留的操作系统必须支持 Kerberos 安全协议。Kerberos 安全协议是一

种第三方的验证服务,它使用传统的加密手段来创建一个共享密钥,这个密钥将成为用户的凭证。无论是请求本地服务还是网络服务,系统都将用它来验证用户的身份。该密钥消除了将用户和密码以明文方式在网络中传输所带来的隐患。用 Kerberos 安全协议允许对一个远程的 DB2 服务器进行单点登录。

- KRB\_SERVER\_ENCRYPT。该验证类型指定了 DB2 服务器可以接受的验证类型为 KERBEROS 或者 SERVER\_ENCRYPT。如果客户端的验证类型为 KERBEROS,则验证将通过 Kerberos 安全系统进行;如果客户端的验证类型不是 KERBEROS,则系统默认为验证类型是 SERVER\_ENCRYPT。

② “管理”栏下的 SYSADM\_GROUP 参数。

该参数为实例定义具有系统管理 SYSADM 权限的组名。默认情况下 SYSADM\_GROUP 并没有值,这意味着 Windows 的管理员用户将会继承 DB2 的 SYSADM 权限。如果想更改这种默认的设置,可以先建立一个用户组,该组拥有一些特定成员,然后将该组名设置为 SYSADM\_GROUP 参数值,只有该组的成员才拥有 SYSADM 权限。

③ “环境”栏下的 NODETYPE 参数。

该参数值为“带有本地和远程客户机的数据库分区服务器”,注意该参数是由 DB2 设置的,且不能被更改。

(5) 更新 SYSADM\_GROUP 参数,设置它的值为 adm1。

在“DBM 配置”对话框中选择 SYSADM\_GROUP 的值,然后单击该值右边的省略号按钮,在“系统管理权限组”下的文本框中输入 adm1,单击“确定”按钮,单击“DBM 配置”对话框中“确定”按钮,再单击弹出的“DB2 消息”对话框中的“关闭”按钮,如图 1-14 所示。

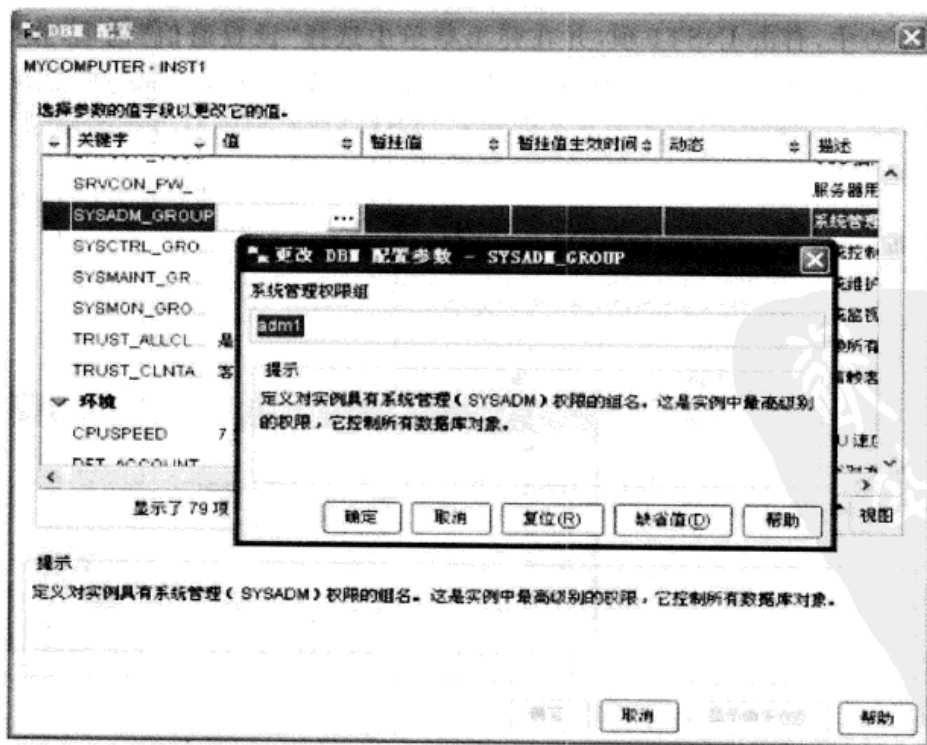


图 1-14 修改 SYSADM-GROUP 参数

为了使设置生效,重新启动 INST1 实例。右击 INST1 项,并在弹出菜单中选择“停止”选项,在“确认停止”对话框中单击“确定”按钮;右击 INST1 项,并在弹出菜单中选择“启动”选项。如果系统显示错误的消息,请阅读相关的错误信息,在下一步修正该错误。

(6) 如果收到一个 SQL5043N 错误信息,这是因为未能成功启动对一个或多个通信协议的支持,尽管核心数据库管理器功能启动成功,如图 1-15 所示。

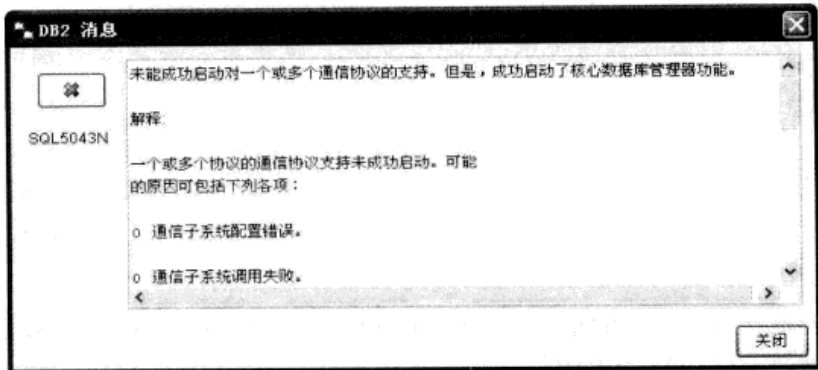


图 1-15 启动错误信息

在实验 4 中,将 DB2COMM 参数值设为 TCP/IP。当 INST1 实例启动时,DB2 会试着启动该通信协议,但是却未能找到一个服务端口来使用。因此在数据库管理器配置文件中,需要设置参数 SVCENAME,指定一个端口号或一个服务文件入口名,为前来访问的 TCP/IP 客户端服务。

(7) 更新 TCP/IP 服务名(SVCENAME 参数),设置它的值为 55000。

在控制中心中,右击 INST1 项,并在弹出菜单中选择“设置通信”选项,在“设置通信-INST1”对话框中选 TCP/IP 复选框,并且单击该复选框右边的“属性”按钮,在“配置 TCP/IP-INST1”对话框中的服务名文本框输入 INST1(或其他名),再在端口号文本框中输入 55000,连续单击“确定”按钮关闭这些配置窗口,如图 1-16 所示。

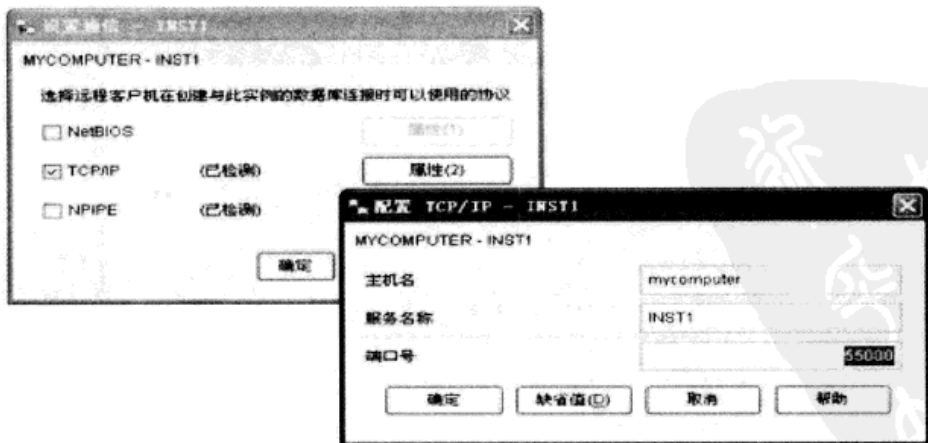


图 1-16 设置通信

重新启动实例使配置生效,在 DB2 消息窗口中将提示“DB2START 处理成功”。

## 实验 2

## 创建数据库/表空间

### 实验任务

- 建立数据库；
- 建立表空间；
- 执行脚本文件来建立多个表空间；
- 访问包含表空间信息的 SYSCAT 视图；
- 列出表空间相关信息；
- 列出容器相关信息。

### 实验内容

#### 1. 创建数据库

数据库的创建可以采用两种方法。一种是在命令编辑器中,使用 DB2 命令来创建,另一种是通过控制中心来进行。下面分别介绍这两种方法。

##### 1) 查看 create database 指令语法并学习使用 create database 创建数据库

创建数据库可以通过 DB2 命令 create database 来完成。通过在线帮助工具可以查看 create database 指令语法。

(1) 选择“命令行工具”→“命令编辑器”选项,或在其他窗口中单击相应的图标来启动命令编辑器,如图 2-1 所示,单击工具栏中的第 4 个图标即可启动命令编辑器。

(2) 在命令编辑器中,在“目标”文本框中输入“? create database”。

(3) 选择“所选”→“执行”选项(或使用 Ctrl+Enter 组合键,或单击工具栏中的绿色三角图标),结果如图 2-2 所示。

(4) 从 create database 命令语法可以看出,该命令可以指定:数据库名、数据库位置(控制文件、设定文件、事务日志文件以及默认的表空间要建立在什么位置)、数据库别名、国别区码、排序集、默认数据块大小,该语句也可为编目表空间、临时表空间、用户表空间等指定特定值。

此外,也可以通过“创建数据库向导”创建一个新数据库(本实验采用该方法)。

##### 2) 通过“创建数据库向导”创建数据库

(1) 在“控制中心”中,右击实例 inst1 下的“数据库”项,在弹出菜单中选择“创建数据库”→“标准”选项,如图 2-3 所示。



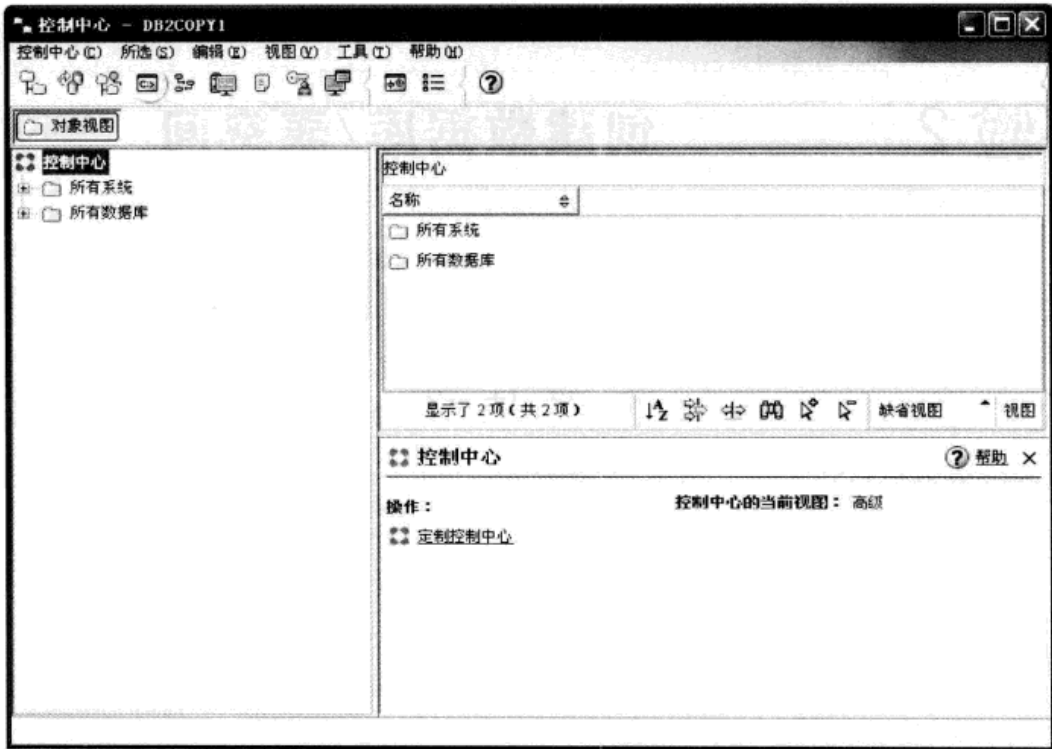


图 2-1 控制中心中启动命令编辑器的方法

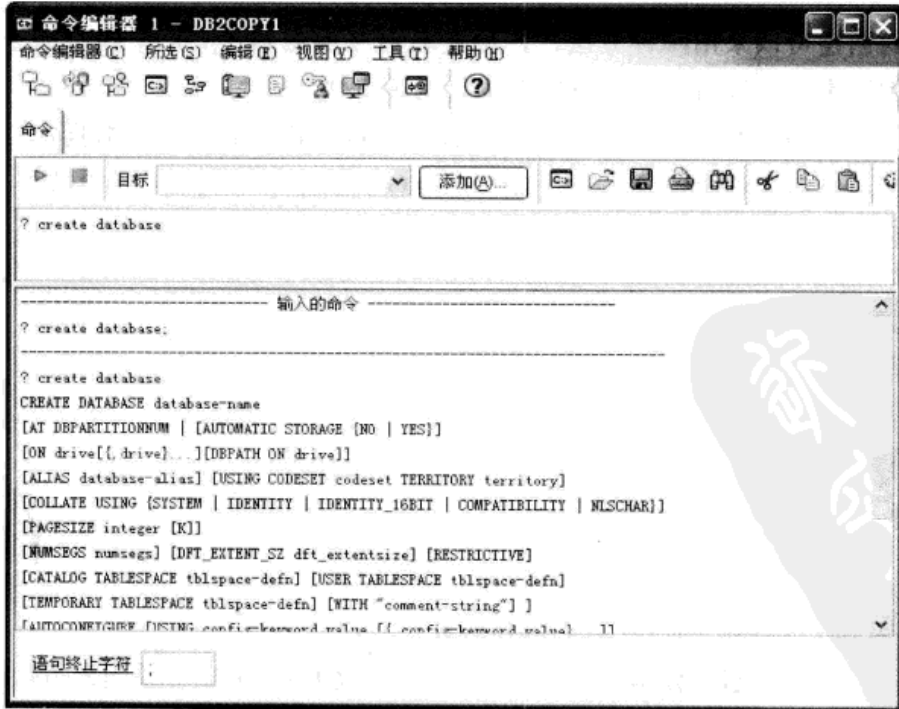


图 2-2 “? create database”的执行结果

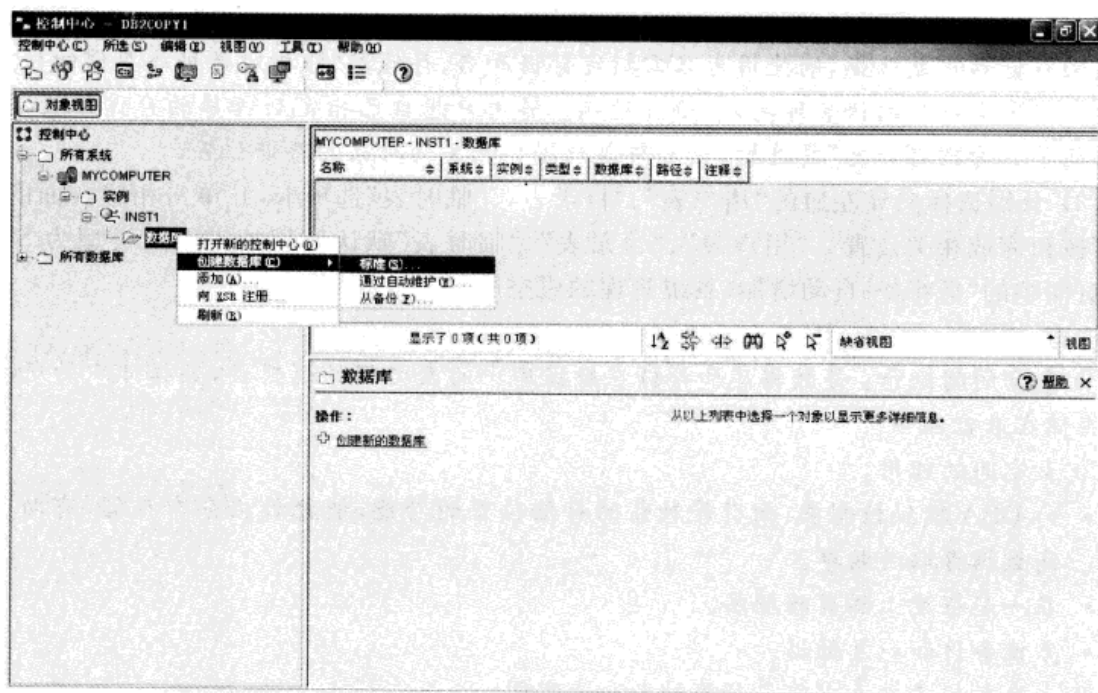


图 2-3 在控制中心中创建数据库

(2) 在向导的“名称”页面中,数据库名文本框中输入 MUSICDB,选中“我想手工管理存储引擎”复选框,如图 2-4 所示,其他值均按默认设置,单击“下一步”按钮。

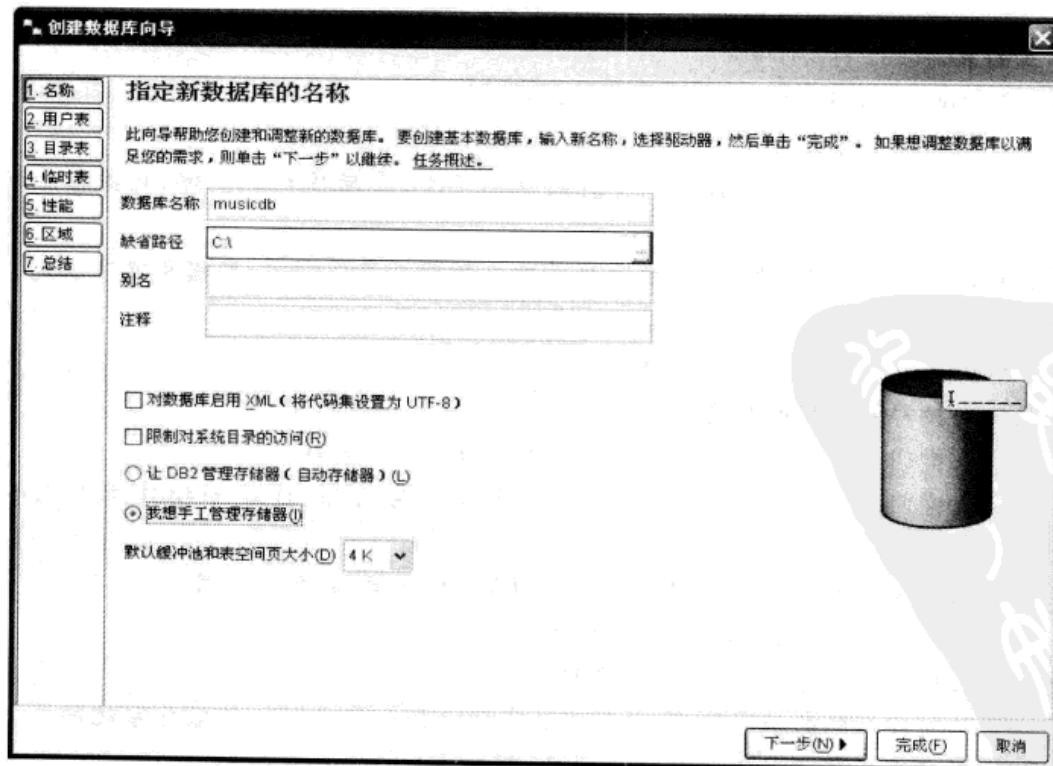


图 2-4 填写数据库名称

**注意：**如果不选中“我想手工管理存储器”复选框，则系统默认选中“让 DB2 管理存储器(自动存储器)”复选框，则使用者只需指定数据库名、存储器(即指定数据库存储的位置)、区域(即指定数据库的语言环境)和语言选项；若用户想自己指定数据库的存储位置，则选中“我想手工管理存储器”复选框，指定与本数据库相关的对象的存储位置。

(3) 分别选择向导左边的“用户表”、“目录表”、“临时表”选项卡，并单击相应页面的“下一步”按钮完成相关设置。“用户表”、“目录表”、“临时表”默认使用的表空间类型为 SMS，即各页面中的“低维护-自动增加(系统管理的表空间)(L)”选项。

**注意：**

① 表空间的概念：是数据库和存储在数据库中的表之间的逻辑层，数据库中的所有对象都存储在表空间中。

② 表空间的作用。

- 为 DBA 提供控制表、索引等对象的存储位置的功能，能把数据分布存储，有助于提高数据存取的效率。
- 在一定程度上隔离物理层。
- 方便备份和恢复数据。

③ 一个数据库中至少需要具备的 3 个表空间。

- 系统编目表空间：每个数据库只有一个，命名为 SYSCATSPACE，用于保存数据库的所有系统表。
- 系统临时表空间：每个数据库至少有一个，命名为 TEMPSPACE1，存放 DB2 在执行 SQL 时，因为排序、连接等操作所产生的临时表。
- 用户表空间：每个数据库中至少有一个，命名为 USERSPACE1，用于存放用户所建立的表。

即上文所说的“目录表”、“临时表”、“用户表”。

④ 表空间按管理方式分类。

- 系统管理表空间 SMS：DB2 默认的表空间类型，由操作系统管理，数据容器只能是目录。
- 数据库管理表空间 DMS：由数据库管理和控制的存储空间，数据容器可以是文件或设备。

一般说来，DMS 的性能要优于 SMS。

⑤ 表空间按存储数据的种类分类。

- 常规表空间：用于存放系统表或用户建立的表格和索引。
- 临时表空间：用于存放在执行 SQL 指令中因为排序、连接等操作所产生的暂时数据。
- 大型表空间：用于存放 LONG VARCHAR 与 LOB 类型的数据。

⑥ DB2 的对象层次结构图如图 2-5 所示。

图 2-5 中的每条虚线框表示一个表空间。

⑦ 容器是一块实际的磁盘空间，是数据的实际存储位置。可以是操作系统的文件系统目录、文件或是没有文件系统的存储设备(裸磁盘)。

⑧ 容器和表空间的对应关系图如图 2-6 所示。

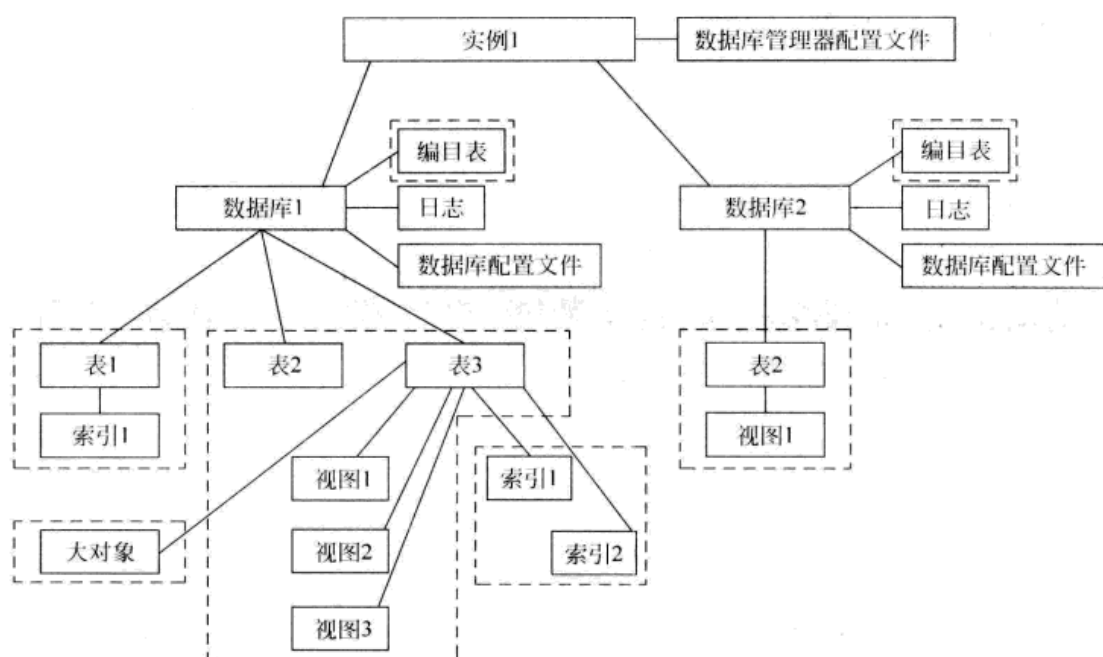


图 2-5 DB2 的对象层次结构图

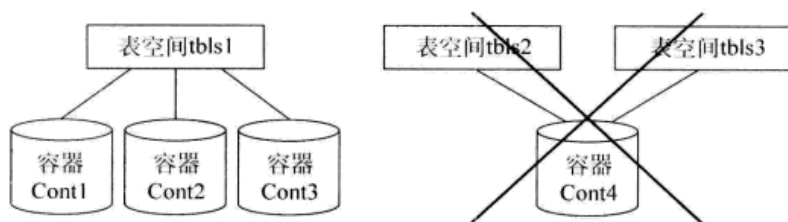


图 2-6 容器和表空间对应关系图

可以看出，一个表空间可以分放在多个容器里，但是一个容器只能属于一个表空间。

#### ⑨ 容器分类如下。

- 目录容器：SMS 表空间可以唯一使用的容器类型。SMS 表空间中可以定义多个容器，每一个容器都可以映射到不同的物理磁盘上，这样可以平衡 I/O 负载。DB2 将会平衡写入到多个容器的数据量。
- 设备容器：因为建立了容器的设备不能用在另一个表空间里面，所以指定容器大小的时候，尽量考虑利用率，没被使用的空间都会浪费。
- 文件容器：文件容器是一个由 DMS 表空间使用的预分配大小的文件。在操作上，文件和设备没有任何区别。定义容器名时，可以使用绝对或相对的文件名。如果目录不存在，会由 DB2 自动创建。同样的，如果文件不存在，也会由 DB2 创建并初始化成指定的大小。当创建表空间时，必须存在指定数量的空间。空间需求的数量，可以以页、KB、MB 或 GB 为单位指定。

#### ⑩ 表空间、容器、表之间的关系总结。

- 一个表空间只能给一个数据库使用。
- 一个数据库中可以有多个表空间。

- 一个表可以使用多个表空间。
- 一个表空间也可以由多个表使用。
- 一个表空间可以位于多个容器上。
- 一个容器只能属于一个表空间。

① 实际应用中,为了提高系统性能,通常会把表空间分配到几个容器上。若需要这样做,则单击“添加”按钮即可,如图 2-7 和图 2-8 所示。为了方便,可直接单击“下一步”按钮。

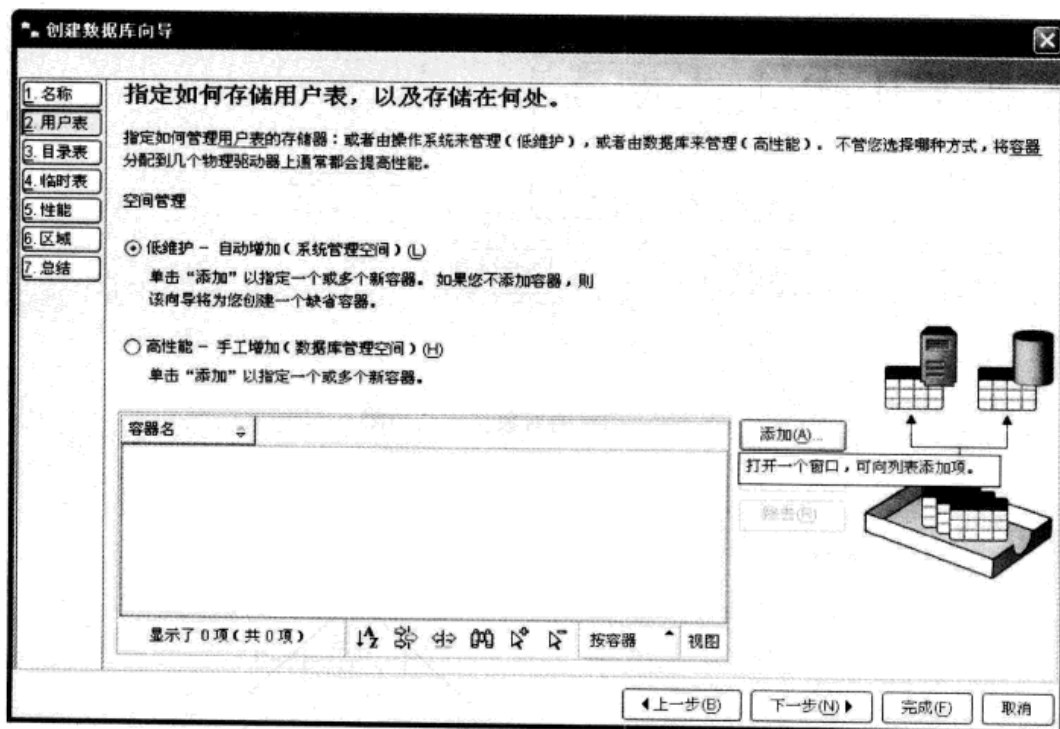


图 2-7 添加容器



图 2-8 指定容器位置

(4) 在向导的“性能”页面中,因为 MUSICDB 的表空间均按照系统默认设置,所以此时不能修改数据块大小或预取大小,如图 2-9 所示(如果为用户表、目录表、临时表设置了特定的容器,则可以通过“性能”页面设置数据块或预取大小)。单击“下一步”按钮开始数据库“区域”的设置。

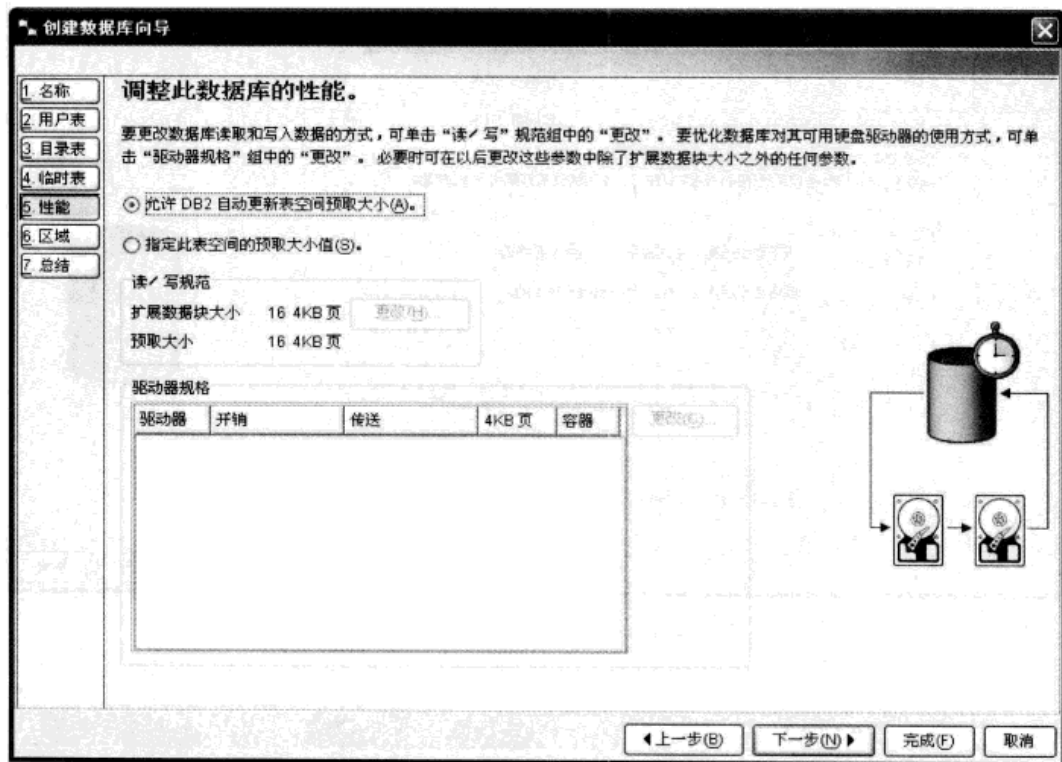


图 2-9 调整数据库性能页面

(5) 向导的“区域”页面可以设置数据库的语言环境和整理顺序,系统已自动设置完成这些选项,如图 2-10 所示,直接单击“下一步”按钮。

(6) 在向导“总结”页面中,会显示前几步所设的创建数据库的参数。单击“显示命令”按钮即可查看创建该数据库的全部命令。单击“上一步”按钮则回到前面步骤修改创建参数。单击“总结”页面中的“完成”按钮,则生成数据库,如图 2-11 所示。

然后,系统就会弹出“DB2 消息”对话框提示数据库已经创建完毕,这时,所创建的 MUSICDB 数据库会出现在控制中心对象树中“数据库”节点下。

### 3) 查看数据库信息

系统数据库目录包含当前实例下所有数据库的条目信息,可以在命令编辑器中输入命令 `list db directory` 查看系统数据库目录中 MUSICDB 的信息,如图 2-12 所示。

#### 注意:

① 数据库建立时如果未设置别名,系统会自动设置一个与数据库名相同的别名。MUSICDB 的别名为 MUSICDB。

② 目录条目类型的“间接”是指该数据库为本地数据库(运行在同一主机),而“远程”指数据库运行在另一个远程系统中。本实验中 MUSICDB 的目录条目类型为“间接”。



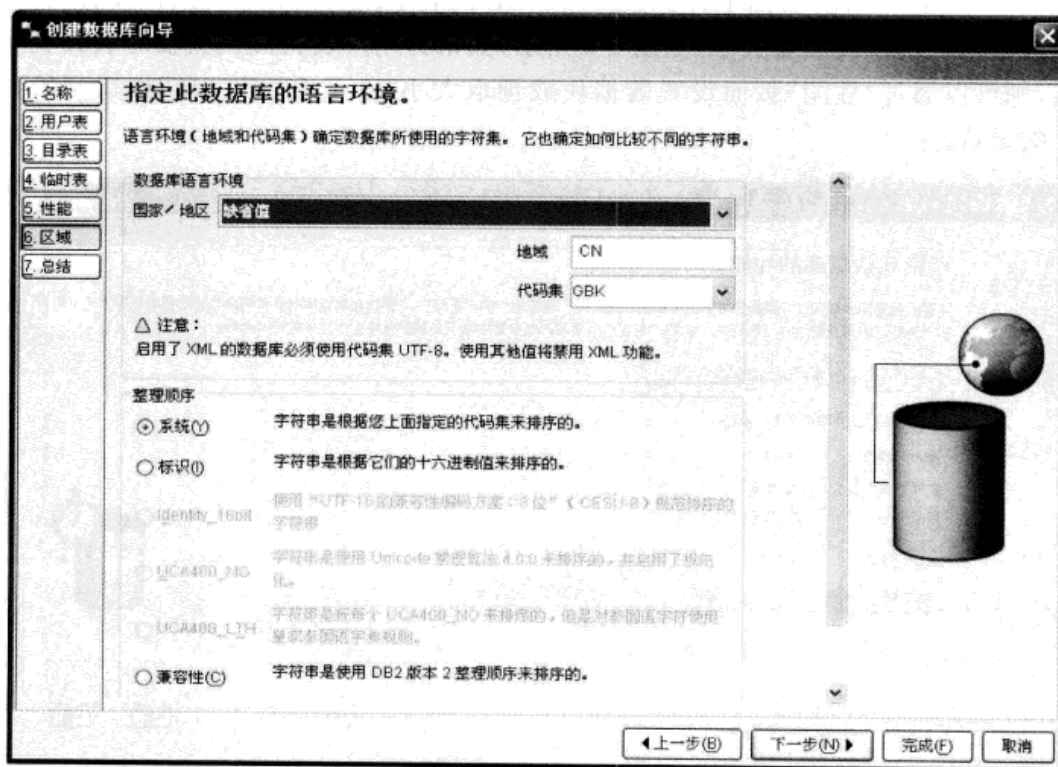


图 2-10 数据库的语言环境设置页面

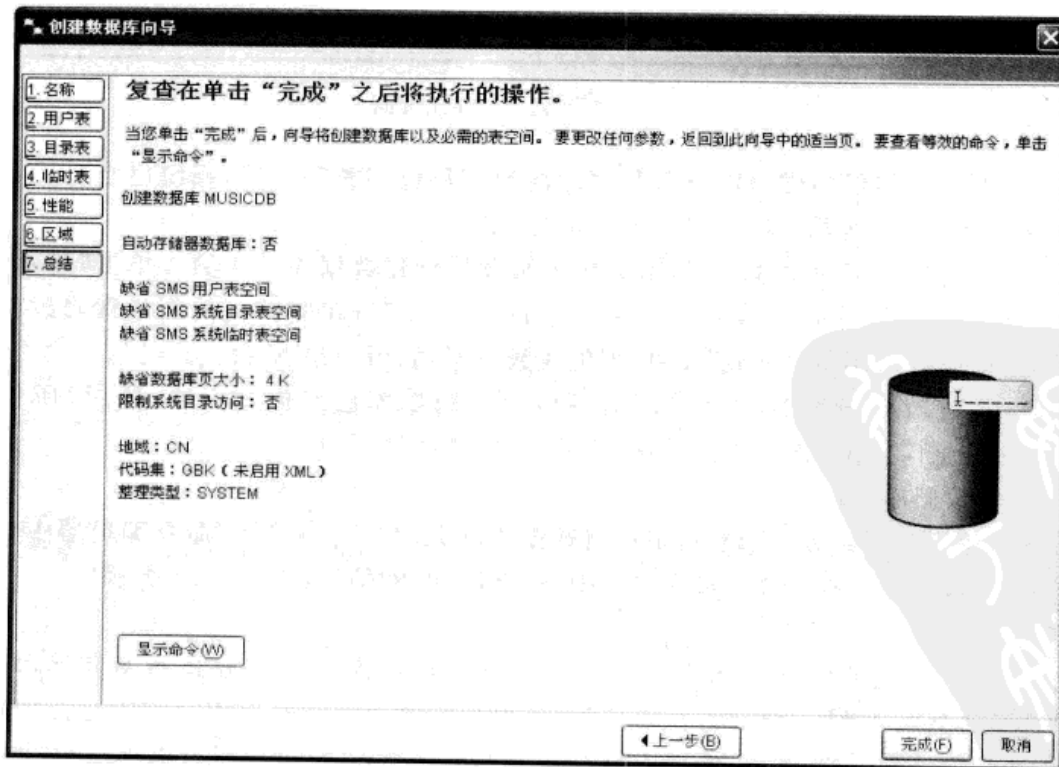


图 2-11 总结页面

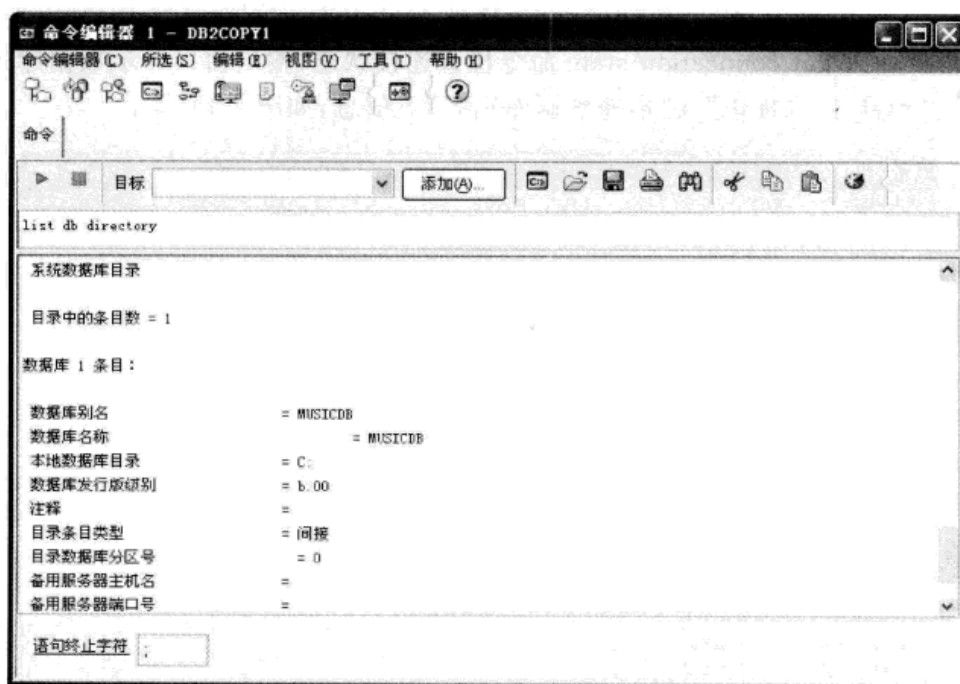


图 2-12 系统数据库目录中 MUSICDB 的信息

#### 4) 查看数据库的连接状态

(1) 在“命令编辑器”中输入 get connection state 命令来检查现在的连接状态。执行该命令后,输出窗口显示连接状态为“可连接而未连接”,如图 2-13 所示。

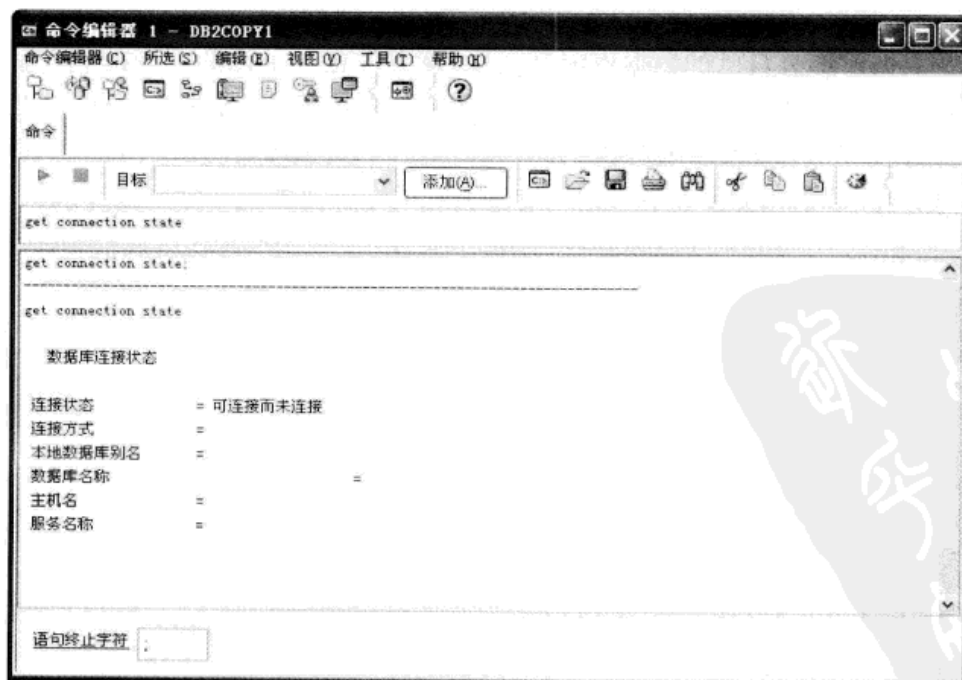


图 2-13 查看数据库的连接状态

(2) 输入 `connect to MUSICDB` 命令连接到 MUSICDB 数据库。

(3) 再次输入 `get connection state` 命令检查连接状态,输出窗口显示数据库连接信息,列出相应的“SQL 授权标识”及“本地数据库别名”等信息,如图 2-14 所示。

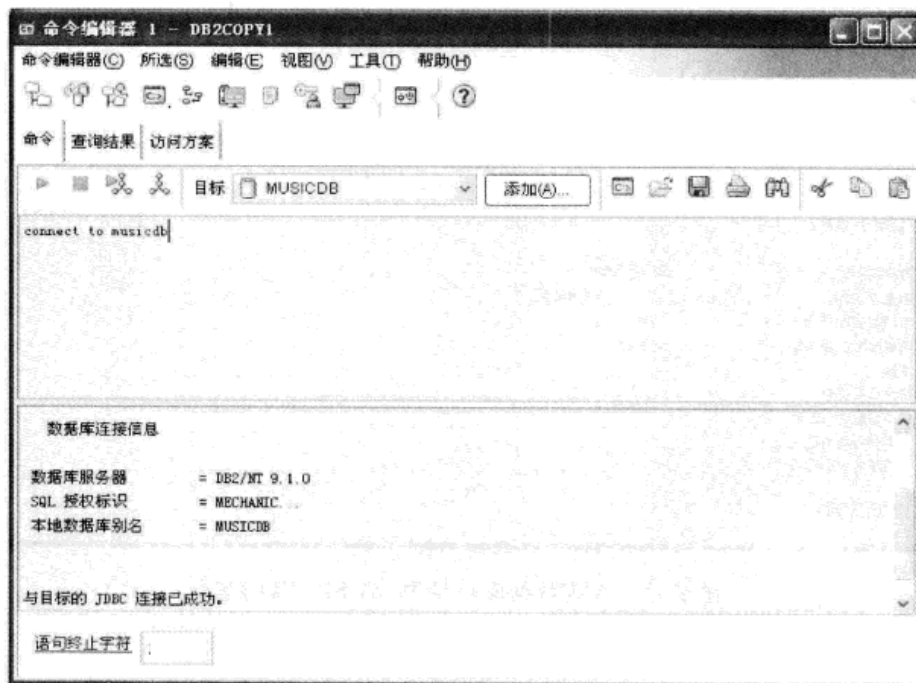


图 2-14 新的数据库连接信息

#### 5) 修改数据库配置文件

每一个数据库都有自己的数据库配置文件,该文件包含了该数据库的相关信息和配置参数。要查看 MUSICDB 数据库的配置文件,可在控制中心的对象树中右击数据库名 MUSICDB,并在弹出菜单中选择“配置参数”选项。借助弹出的“数据库配置—MUSICDB”窗口,可以查看或更改相应的参数值,如图 2-15 所示。

#### 6) 查看表空间及表空间容器等信息

(1) 前文提到,数据库创建时,三个表空间也同时被创建,在命令编辑器中输入命令 `list tablespaces`,可查看这些表空间的相关信息,如图 2-16 所示。结果如下。

- SYSCATSPACE——ID 0 系统编目表空间。
- TEMPSPACE1——ID 1 系统临时表空间,用于存储系统临时数据。
- USERSPACE1——ID 2 用户表空间。

(2) 表空间容器信息可以通过命令 `list tablespace containers` 获取。

要查看 MUSICDB 中标识为 0 的表空间容器信息,可输入命令:

```
list tablespace containers for 0.
```

结果如图 2-17 所示。

该容器类型为路径/目录,此目录为 `X:\inst1\NODE0000\SQL00001\SQLT0000.0` (`inst1` 为实例名,X 为 MUSICDB 创建时所在的盘符)。

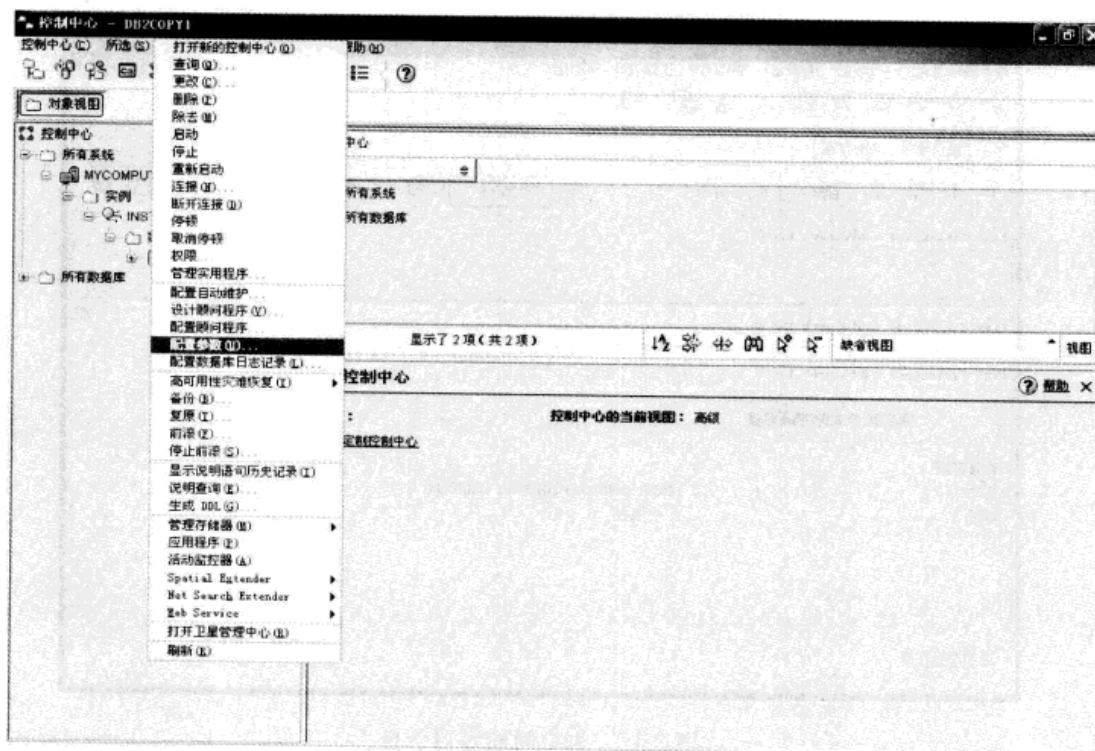


图 2-15 查看数据库配置参数

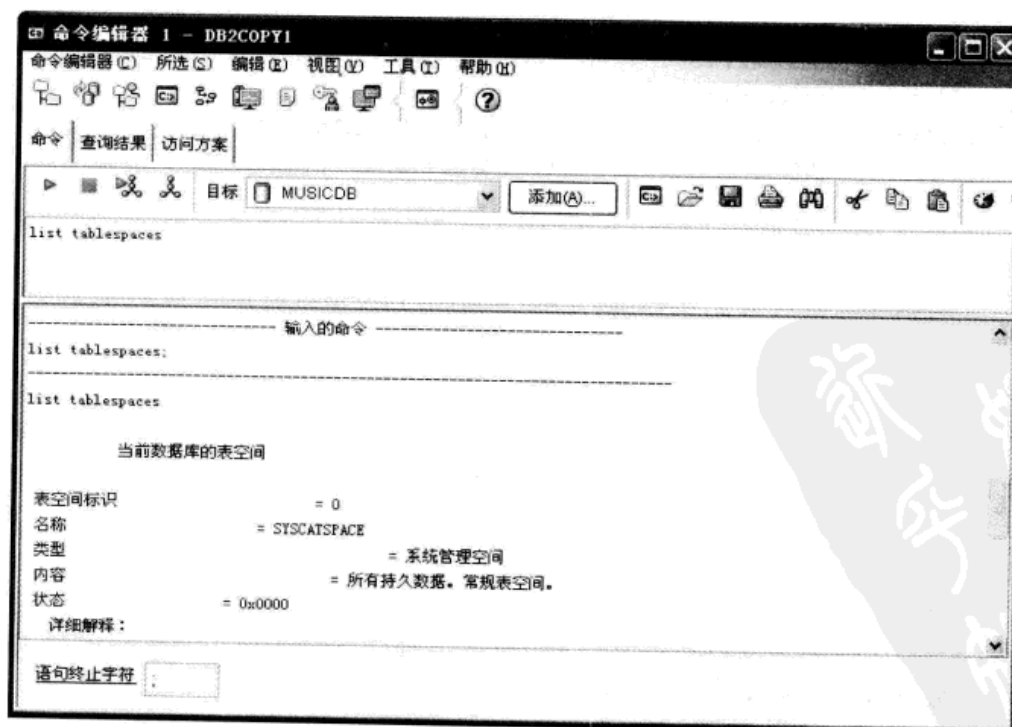


图 2-16 查看表空间信息

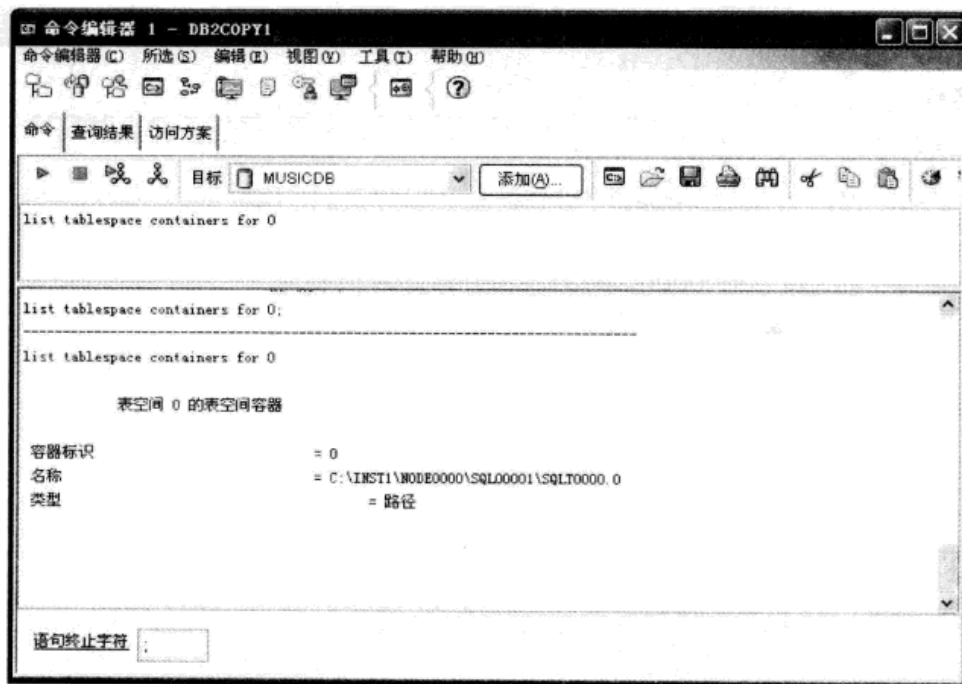


图 2-17 表空间的容器信息

(3) 要查看临时表空间(ID=1)、默认用户表空间(ID=2)和大型表空间(ID=3)的容器信息,可用下列语句实现:

```
list tablespace containers for 1
list tablespace containers for 2
list tablespace containers for 3
临时表空间的默认路径为 X: \inst1\NODE0000\SQL00001\SQLT0001.0
用户表空间的默认路径为 X: \inst1\NODE0000\SQL00001\SQLT0002.0
大型表空间的默认路径为 X: \inst1\NODE0000\SQL00002\SYSTOOLSPACE
```

#### 7) 查看系统目录表和视图

在命令编辑器中输入命令 `list tables for system` 列出系统目录表和视图,这些表和视图存储在 SYSCATSPACE 表空间中。

SYSCAT 模式拥有所有建立在系统目录表上的视图,可通过该模式的这些视图来访问系统目录表。例如在命令编辑器中输入如下命令查看所有模式为“SYSIBM”(即系统目录表)的表名和表标识号:

```
select tabname,tableid from syscat.tables where tabschema = 'SYSIBM' and type = 'T'
```

## 2. 建立表空间

数据库 MUSICDB 需要额外的几个表空间,先通过“控制中心”创建其中一个表空间。其余表空间通过脚本命令文件来创建。

### 1) 通过向导建立表空间 dms01

(1) 在控制中心左边的对象树上,右击数据库 MUSICDB 下的“表空间”项,在弹出菜单中选择“创建”选项,如图 2-18 所示。

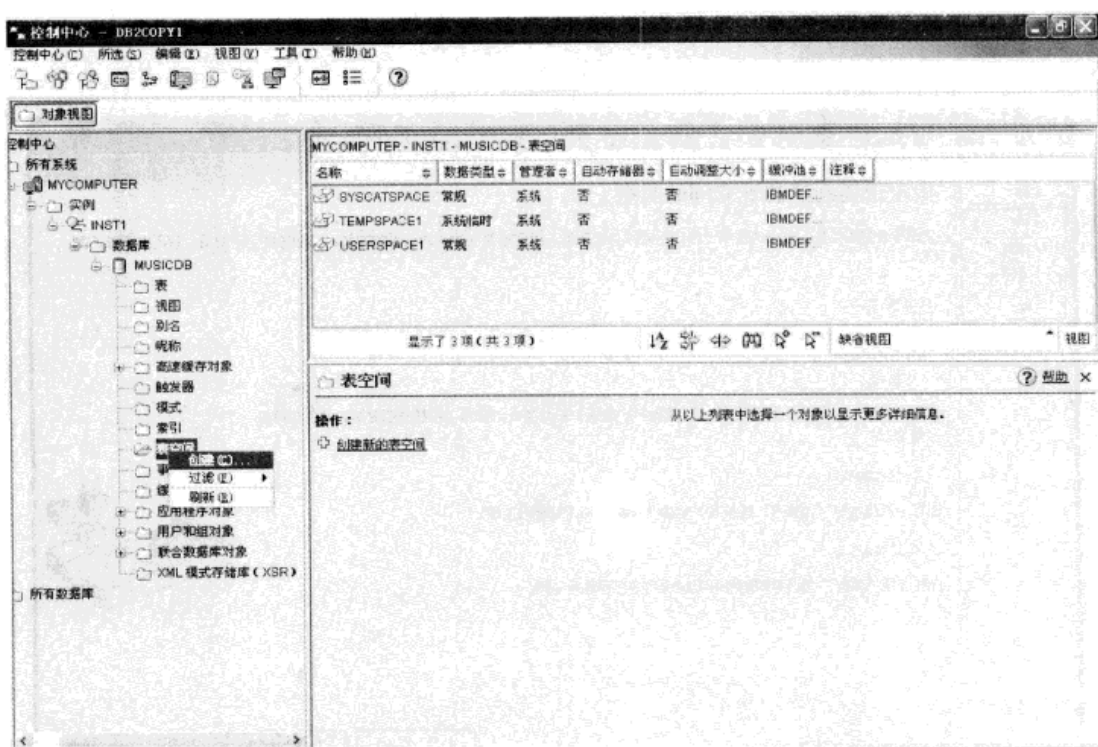


图 2-18 在命令中心中创建表空间

(2) 在“创建表空间向导”的第一页中,指定新表空间的名称为 dms01,并单击“下一步”按钮,如图 2-19 所示。

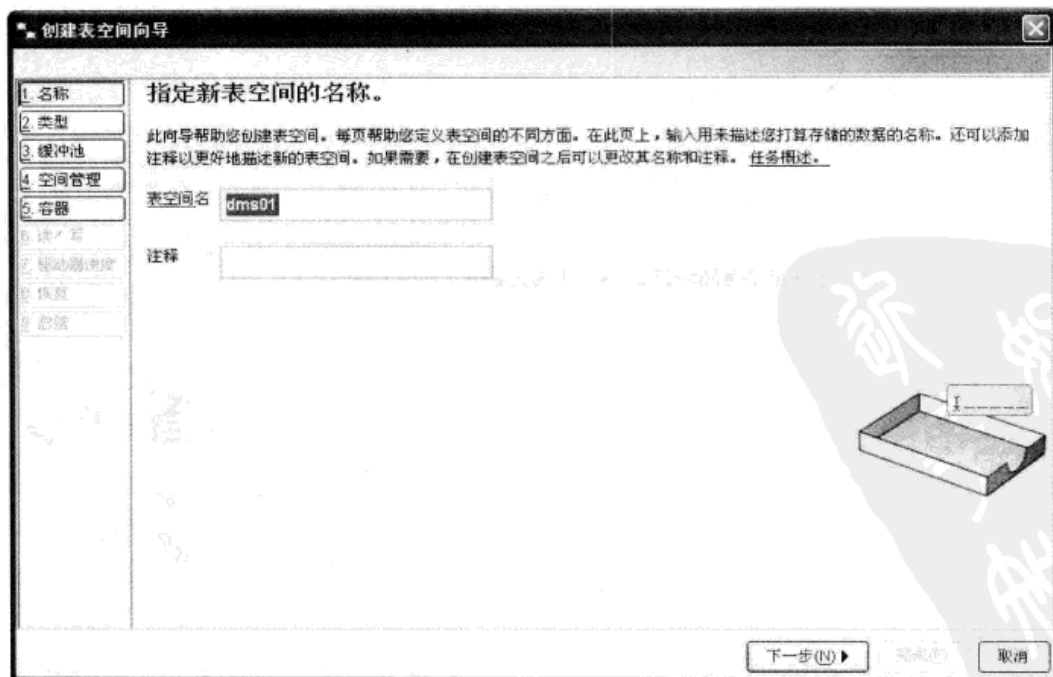


图 2-19 指定表空间名称页面

(3) 在向导的“类型”页面,选择表空间的类型为常规,并单击“下一步”按钮,如图 2-20 所示。

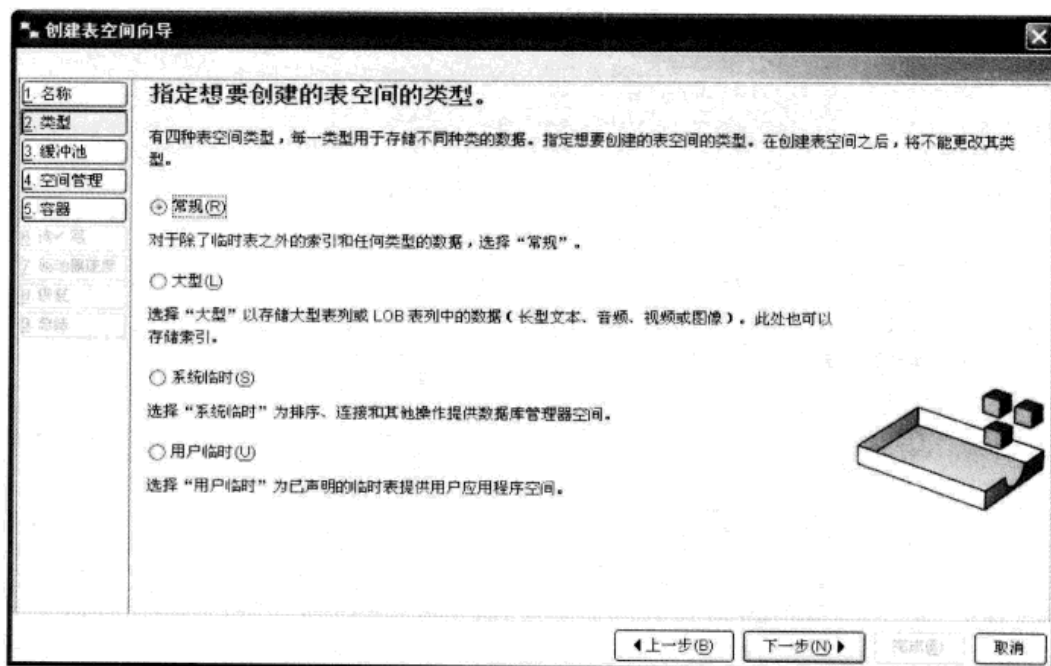


图 2-20 指定表空间存储类型页面

(4) 在“缓冲池”页面,选择默认的缓冲池 IBMDEFAULTBP(如果不想使用默认的缓冲池,可以单击“创建”按钮新建),并单击“下一步”按钮,如图 2-21 所示。

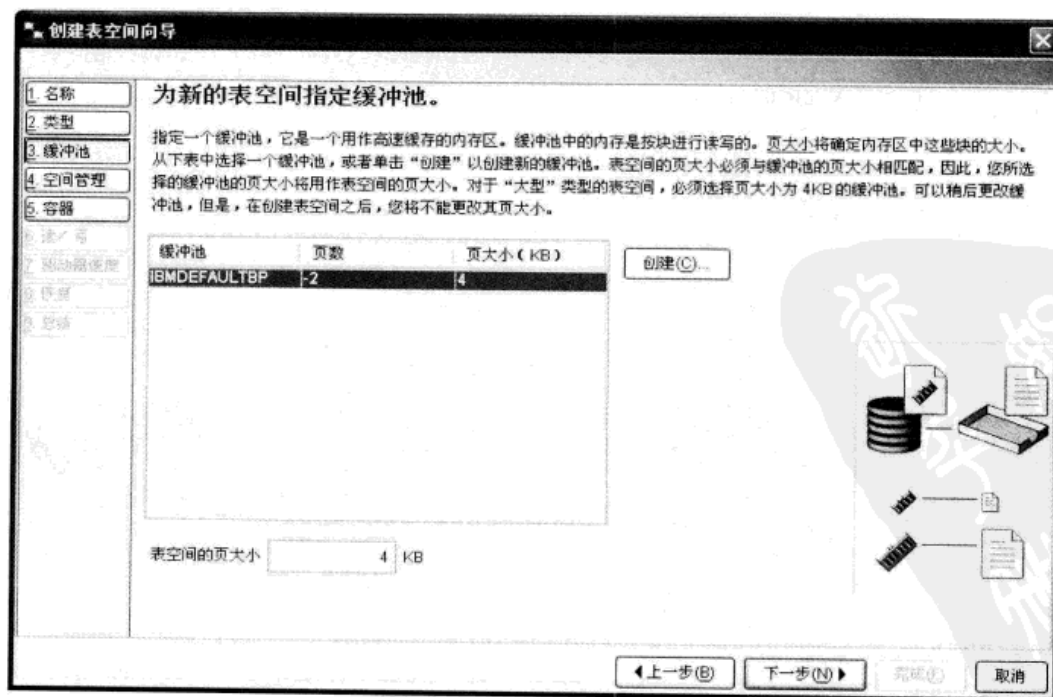


图 2-21 指定缓冲池页面



(5) 在“空间管理”页面,选择该表空间为“数据库管理表空间”,并单击“下一步”按钮,如图 2-22 所示。

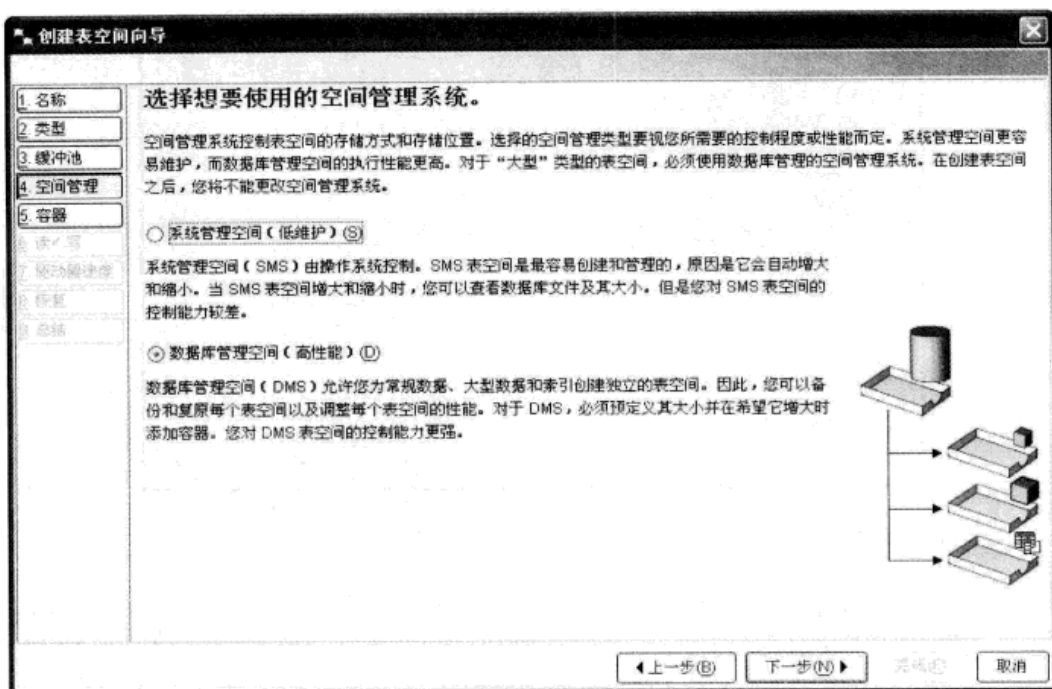


图 2-22 指定表空间管理类型页面

(6) 在“容器”页面中,单击“添加”按钮,弹出“定义容器”对话框。选择容器所在的目录(为了提高性能,可以为 dms01 表空间定义多个容器),然后单击“确定”按钮。继续单击“下一步”按钮,如图 2-23 和图 2-24 所示。

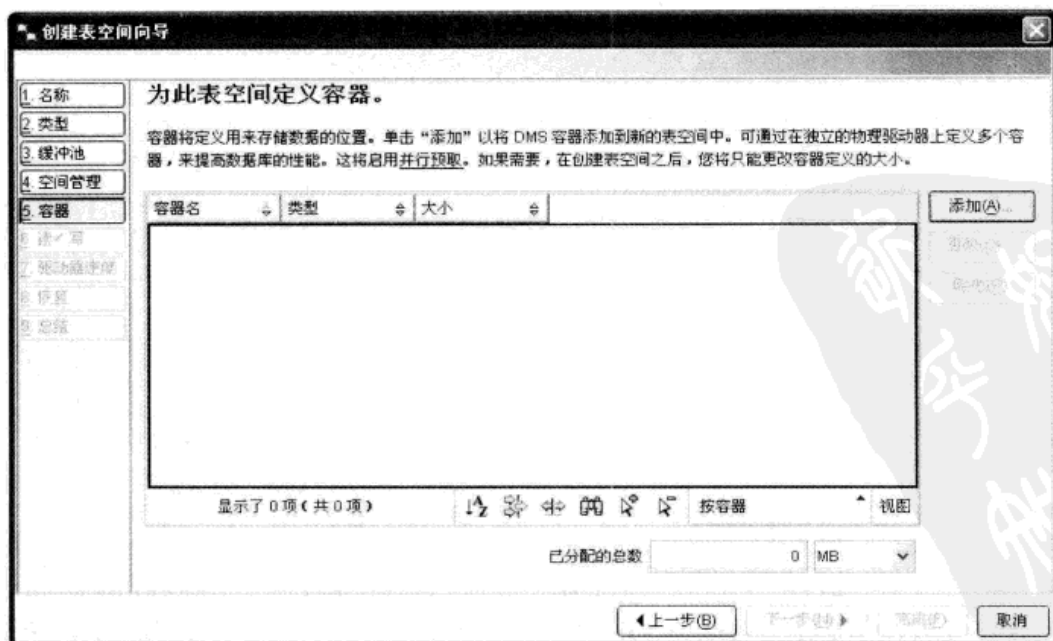


图 2-23 为表空间定义容器页面

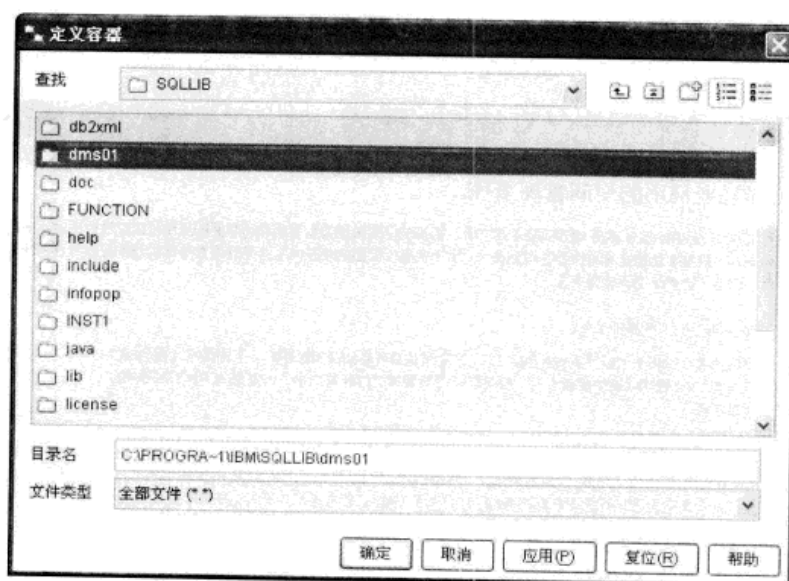


图 2-24 指定容器名称

(7) 在向导的“读/写”页面,根据表的大小确定表空间的扩展数据块大小和预取大小。继续单击“下一步”按钮,如图 2-25 所示。

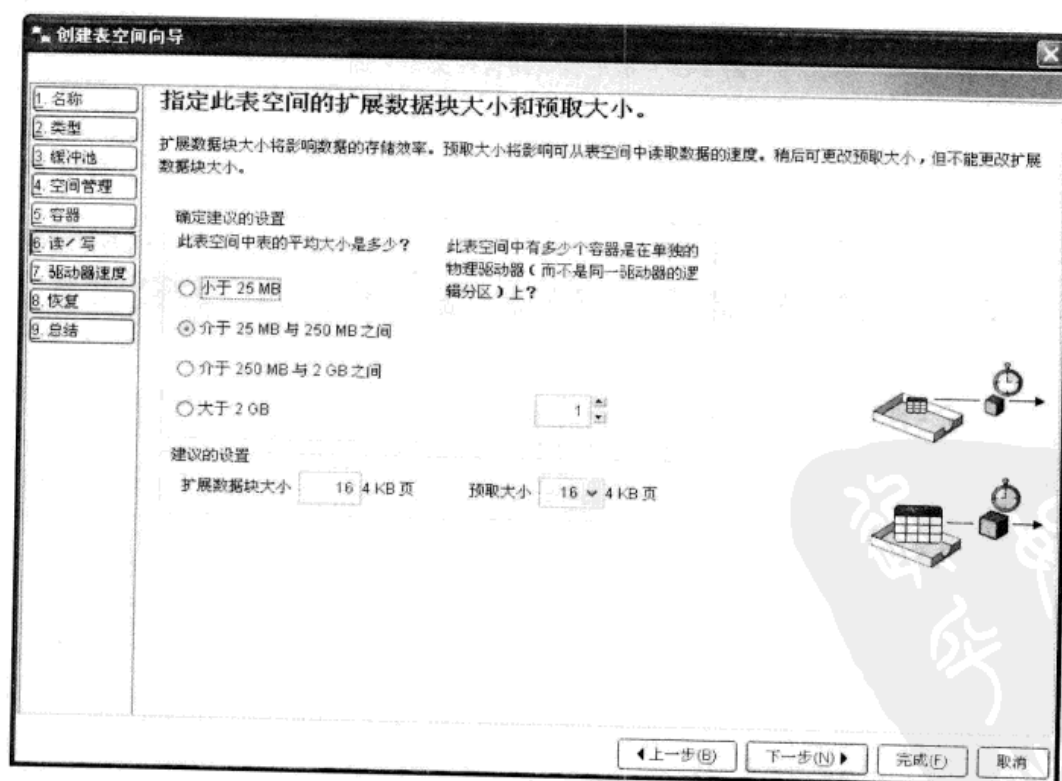


图 2-25 读/写页面

(8) 在“驱动器”页面,选择驱动器类型。继续单击“下一步”按钮,如图 2-26 所示。

(9) 在恢复选项中,可以为新表空间指定已废弃的表的恢复选项。

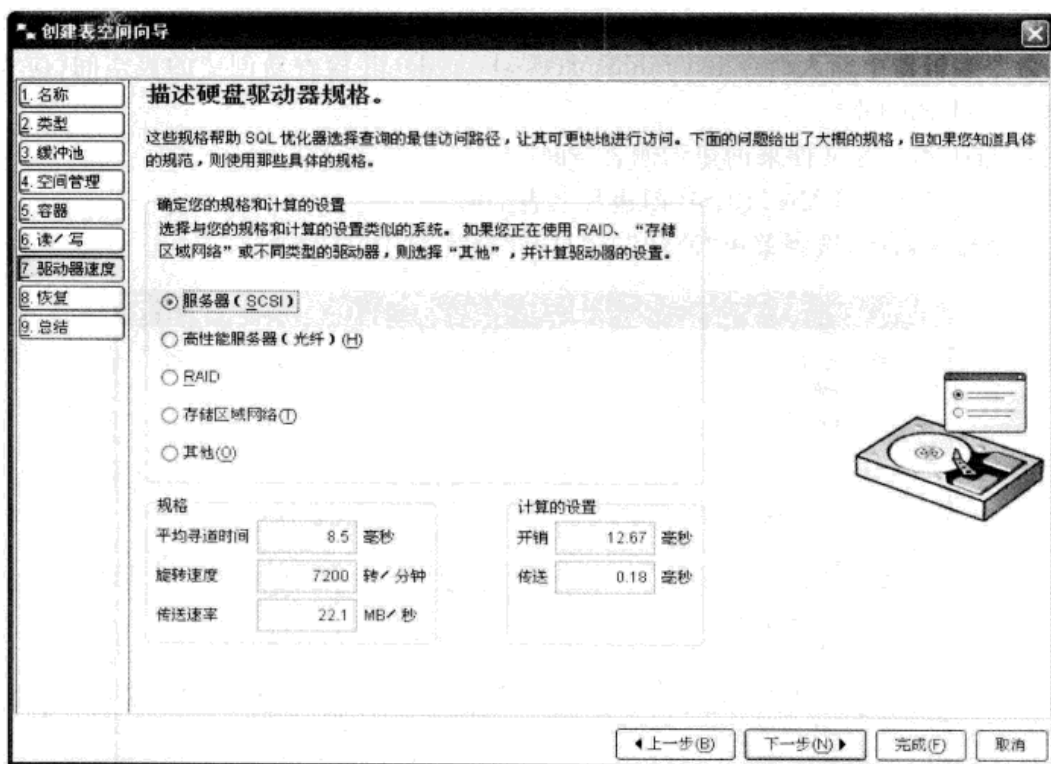


图 2-26 驱动器页面

(10) 设置完该表空间的所有选项后,单击“总结”页面的“显示 SQL”按钮查看所做的设定,如图 2-27 所示。

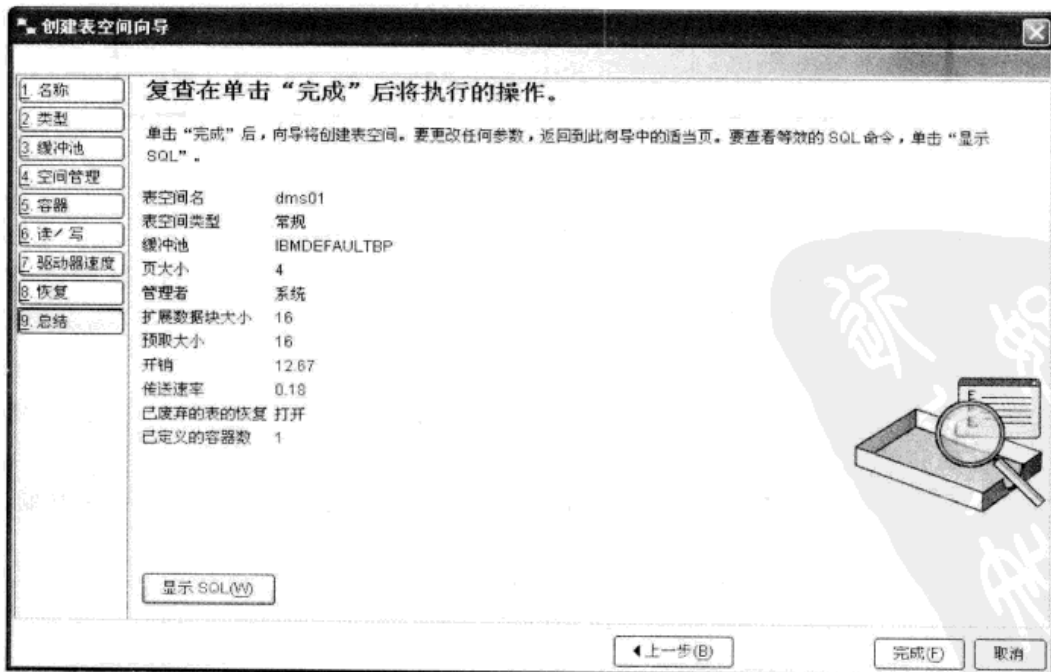


图 2-27 总结页面

(11) 单击向导的完成按钮。

在命令编辑器中输入命令 `list tablespaces show detail` 检查新创建的表空间(应先执行 `connect to MUSICDB`)。

2) 执行脚本命令文件来创建其他表空间

通过执行脚本文件 CRTBLSP 创建其余表空间。

在命令编辑器中,选择菜单“所选”→“打开”命令,如图 2-28 所示。

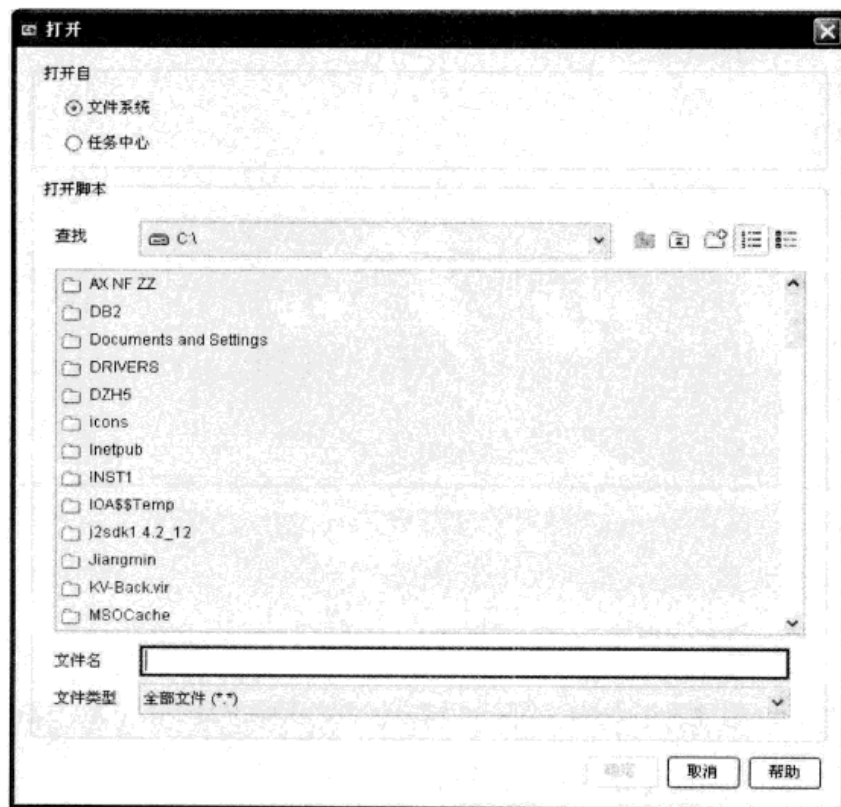


图 2-28 查找脚本

找到脚本所在的目录及 CRTBLSP 脚本,单击“确定”按钮,如图 2-29 所示。

该脚本将创建 dms02, dms03, dms04, dms05, dms06 和 sms01 表空间,执行该脚本,如图 2-30 所示。

3) 查看表空间信息

(1) 在控制中心对象树中,右击 MUSICDB 下的表空间,并在弹出菜单中选择“刷新”选项,可以查看到新的表空间信息。

在控制中心中并不能查看到表空间的标识号,但可通过在命令编辑器中输入 `list tablespaces` 看到,如图 2-31 所示。

表空间及其关联的标识号如下:

```
SYSCATSPACE—ID 0
TEMPSPACE1—ID 1
USERSPACE1—ID 2
DMS01—ID 3
```

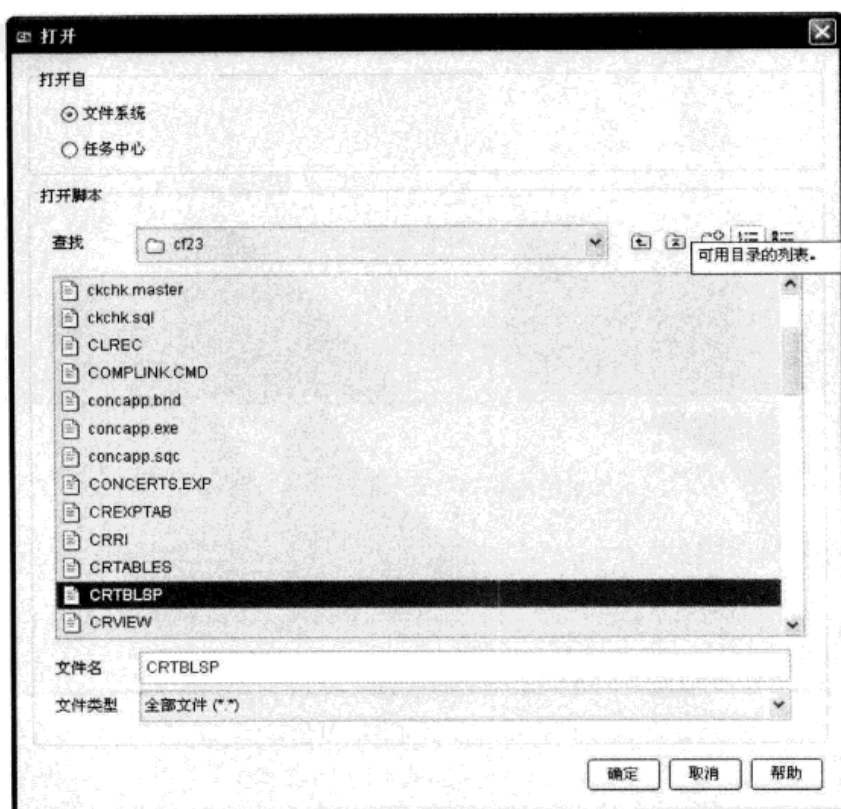


图 2-29 cf23 下的 CRTBLSP 脚本

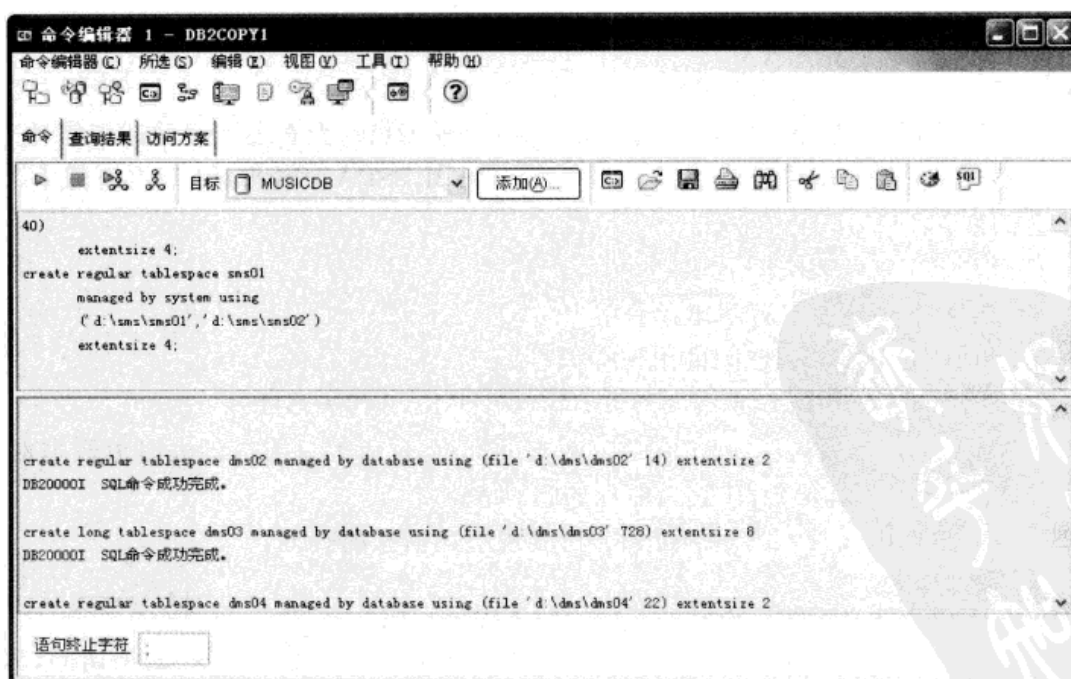


图 2-30 执行 CRTBLSP 脚本

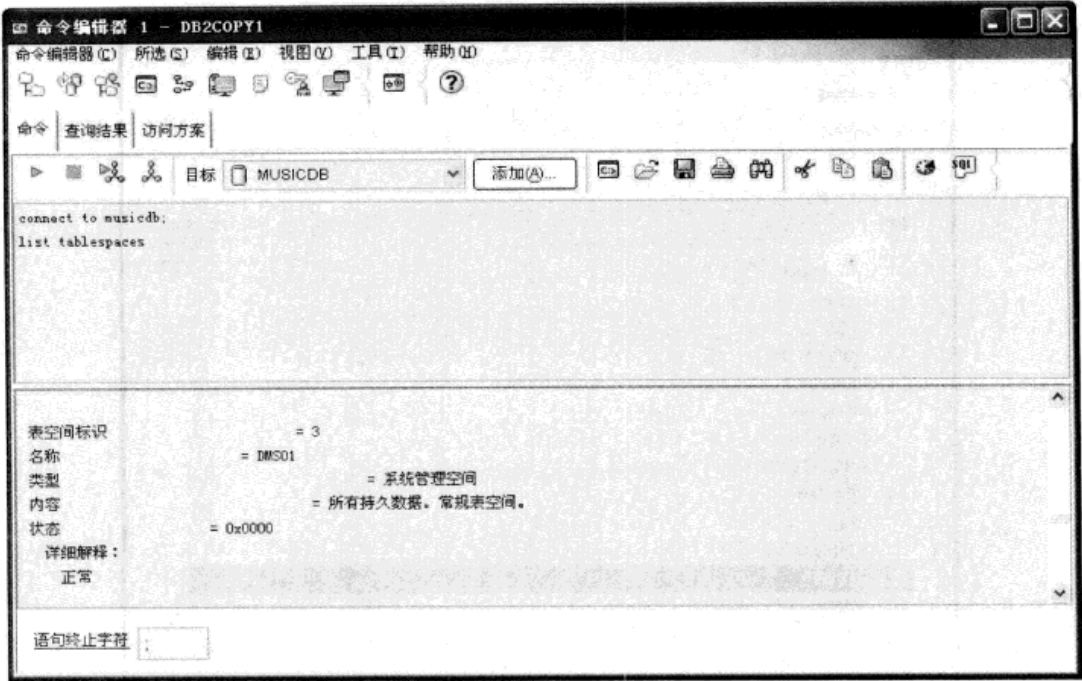


图 2-31 表空间的标识信息

- DMS02—ID 4
- DMS03—ID 5
- DMS04—ID 6
- DMS05—ID 7
- DMS06—ID 8
- SMS01—ID 9

在命令编辑器中输入命令：list tablespaces show detail,查看表空间的详细信息。

(2) 表空间的信息也可通过视图 SYSCAT. TABLESPACES 来访问,该视图中包含表空间信息的列如下：

- TBSPACE——表空间名；
- DEFINER——表空间创建者的用户名；
- TBSPACEID——表空间的内部标识号；
- TBSPACETYPE——表空间类型,D 代表 DMS,S 代表 SMS；
- DATATYPE——表空间可存储的数据的类型,L 代表只能存储大对象数据对象,A 代表所有永久对象,T 代表只能存储临时表。

在命令编辑器中输入下列 SQL 语句,可实现上述信息的查询：

```
select tbpace, definer, tbpaceid, tbpacetype, datatype from syscat.tablespaces
```

(3) 特定的表所相关的表空间信息可以通过 SYSCAT. TABLES 视图来访问,该视图中包含表空间信息的列如下：

- TBSPACEID——该表所在的主表空间内部标识号；
- TBSPACE——该表所在的主表空间名称；

- INDEX\_TBSPACE——包含该表索引的表空间；
- LONG\_TBSPACE——包含该表大对象数据的表空间。

为列出表 SYSIBM.SYSTABLES 所在表空间的信息,可在命令编辑器输入如下 SQL 语句:

```
select tabname, tbspaceid, tbspace, index_tbspace, long_tbspace  
from syscat.tables  
where tabname = 'SYSTABLES'
```

在命令编辑器中输入命令 connect reset 断开与 MUSICDB 的连接。





### 实验任务

- 建立表；
- 建立索引；
- 建立视图；
- 建立别名；
- artists 表与 stock 表之间添加参照完整性约束；
- 为 stock 表添加检查约束；
- 为 reorder 表添加触发器；
- 访问这些对象相关的系统编目信息。

### 实验内容

#### 1. 创建表

##### 1) 利用向导创建表 artists

(1) 在控制中心刷新表空间,并确保在开始创建表之前,能看到所有的表空间。

(2) 在控制中心的 MUSICDB 数据库对象列表中,右击“表”项,并在弹出菜单中选择“创建”选项,则弹出“创建表向导”对话框。

(3) 在“创建表向导”对话框中,如图 3-1 所示,在“表名”文本框中输入 artists,然后单击“下一步”按钮定义列。

- 在列定义窗口中,单击“添加”按钮,在“添加列”对话框(如图 3-2 所示)中,输入列名 artno,在“数据类型”下拉列表中,选择 SMALLINT,注意这一列不允许为空。单击“应用”按钮,继续定义其他列。
- 定义列 name,选择数据类型为 VARCHAR,长度为 50,允许为空。
- 定义 classification 列,数据类型为 CHARACTER,长度为 1,不允许为空。
- 定义 bio 列,数据类型 CLOB,LOB 单位选择 KB,长度为 100。LOB 选项的“记录”和“压缩”均选上,并选择“可空”。
- 定义 picture 列,数据类型为 BLOB,LOB 单位为 KB,长度为 500。选择 LOB 选项的“压缩”选项,选择“可空”。

(4) 在继续下面操作前,要确保上述定义的列具有下列的性质,如表 3-1 所示。如果没有,则用“更改”按钮进行更改(此处定义错误,可能会使后续结果不正确),如图 3-3 所示。

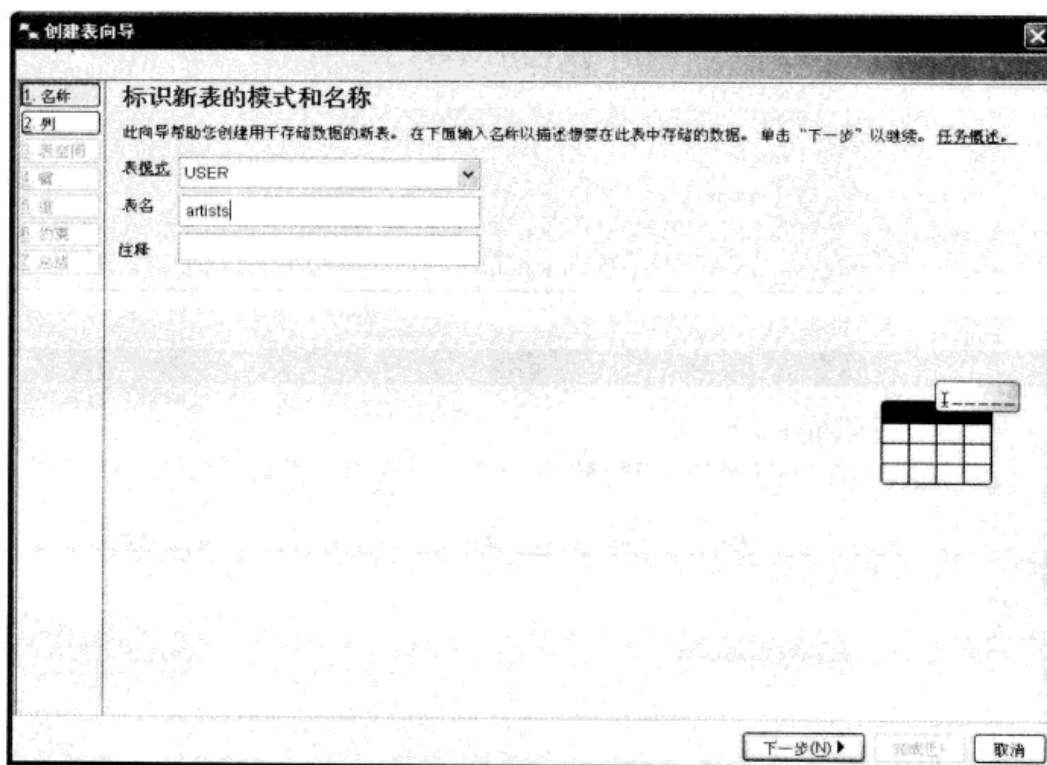


图 3-1 “创建表向导”对话框

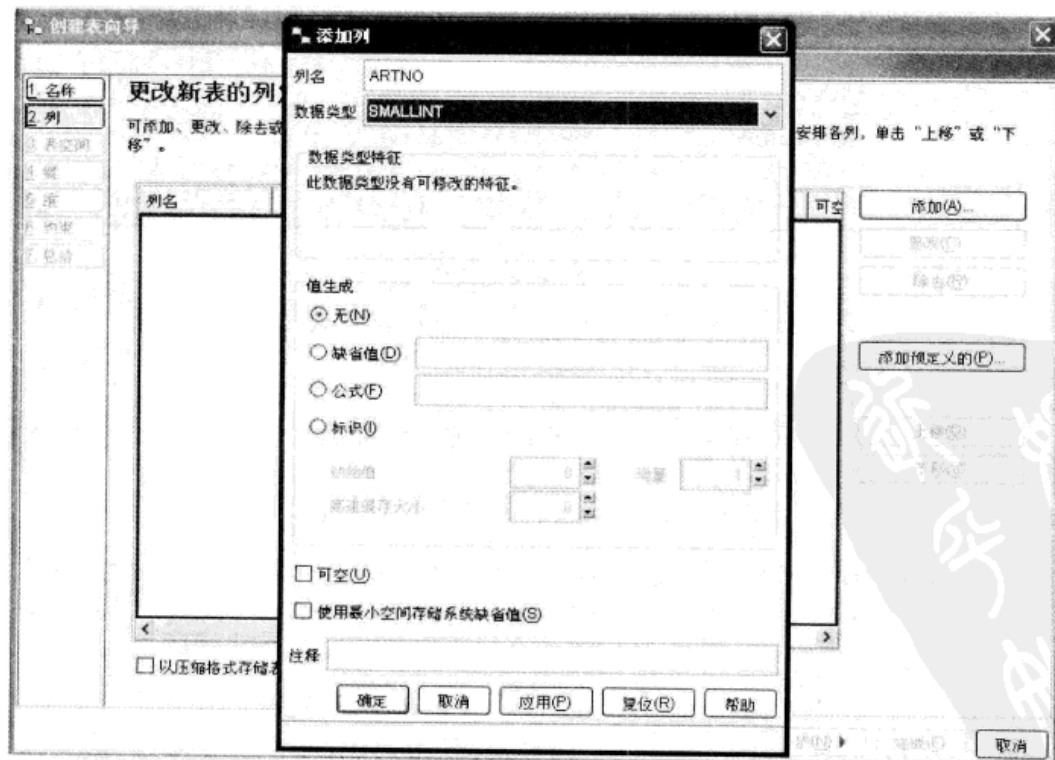


图 3-2 添加列定义

表 3-1 artists 表结构

Column name	Data type	Length	Nullable	LOB option
artno	smallint		No	
name	varchar	50	Yes	
classification	character	1	No	
bio	clob	100KBytes	Yes	Logged, Compact
picture	blob	500KBytes	Yes	Compact

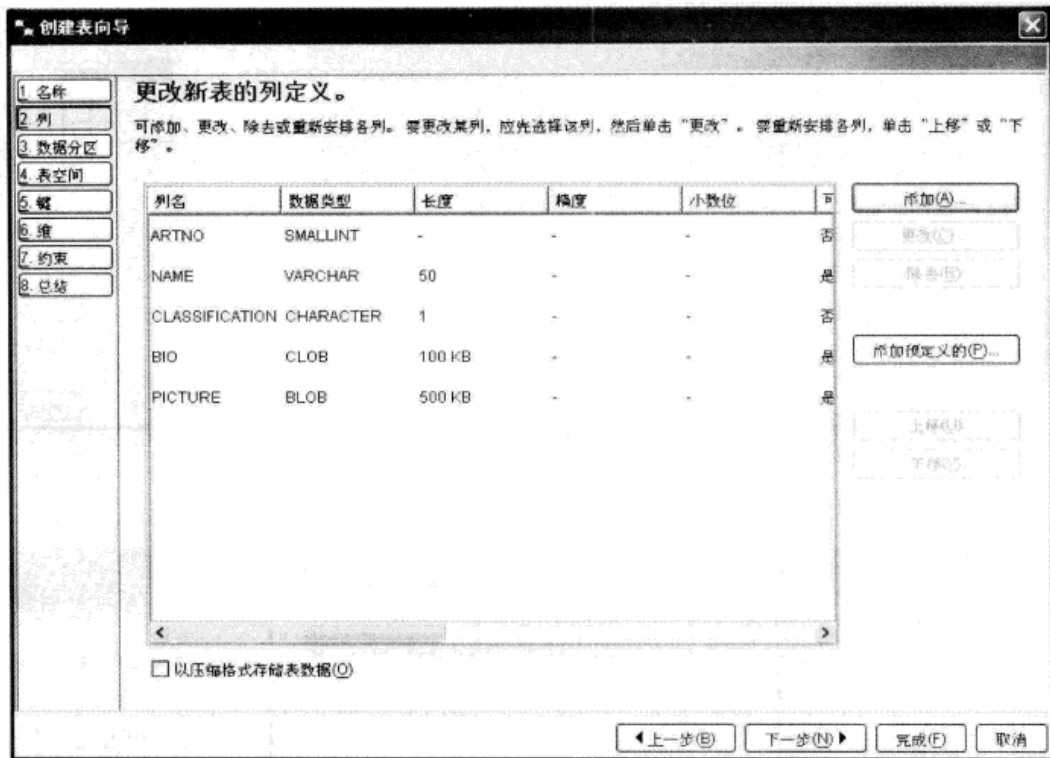


图 3-3 检查列定义结果表

(5) 单击“下一步”按钮进入“数据分区”对话框,跳过不做设置。

**注意:** 通过表分区能够将属于一个表的数据放入一个或多个数据分区中,而每个数据分区可以存储在不同的表空间中。这样做的好处是:提高了性能,并且更容易添加和除去表数据。

(6) 单击“下一步”按钮进入“表空间”对话框,为 artists 表定义表空间,如图 3-4 所示。

在表空间处,选择 DMS01 表空间;选择“使用单独的索引空间”,并选择 DMS02 表空间;选择“使用独立的长空间”,并选择 DMS03 表空间。然后单击“下一步”按钮,为 artists 表定义键。

(7) 定义键值对话框,如图 3-5 所示,单击“添加主键”按钮,在可用列中,选择 ARTNO 列,并按下“>”按钮,将 ARTNO 加入到“选择的列”中(只有非空列才被选择,所以主键必须被定义为非空)。

(8) 单击“下一步”按钮进入“维”对话框,跳过不做设置。

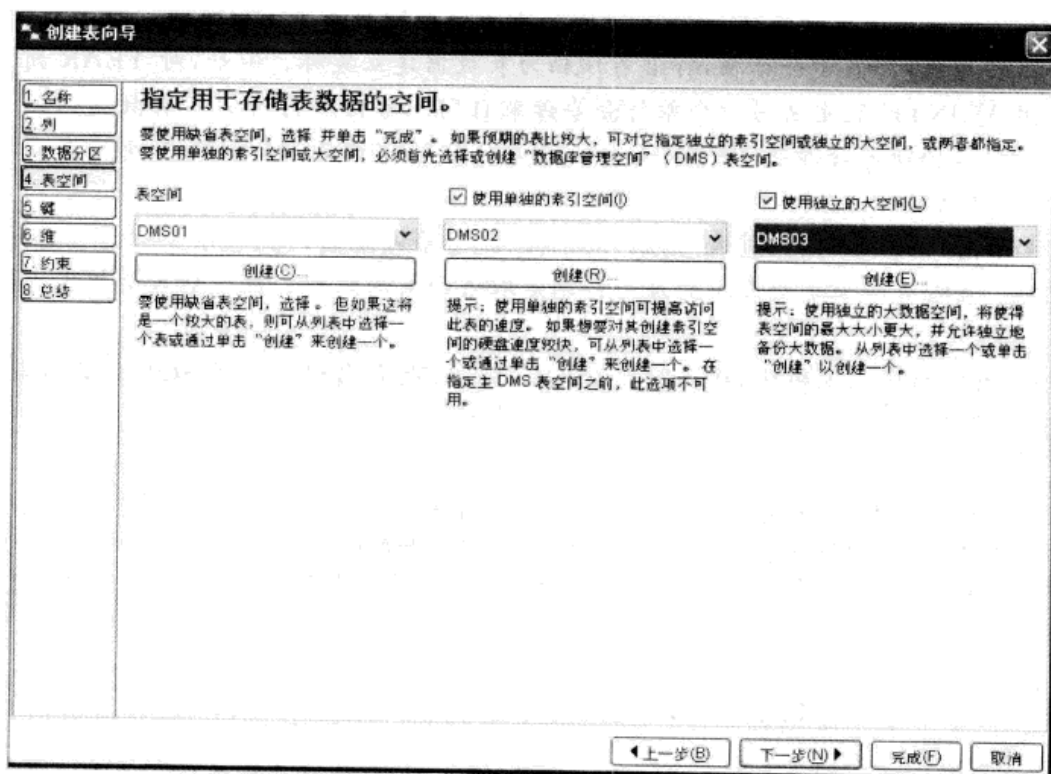


图 3-4 指定存储表数据的空间

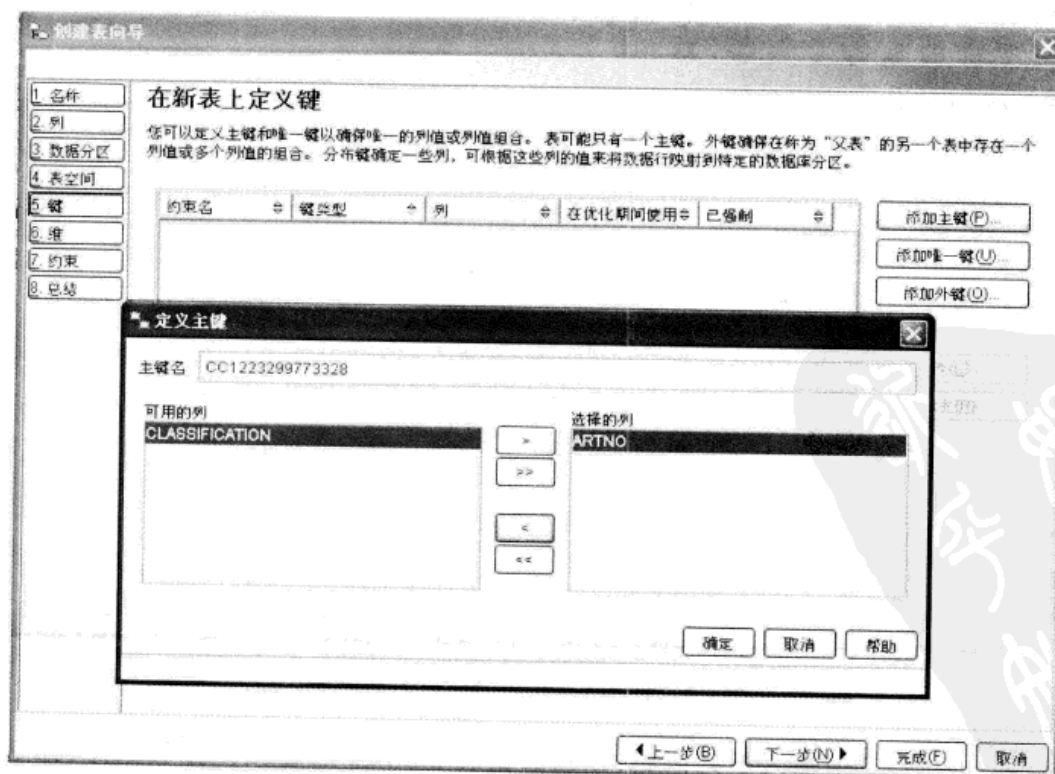


图 3-5 为表定义键

**注意：**设置数据集群窗口，对表定义一个或多个维即启用了多维集群。每个维（它可包含一个或多个表列）表示一个轴，将沿着该轴为表数据建立集群。例如，对 YEAR 列定义一个维并对 MONTH 列定义另一个维将会导致来自同一年份和月份的所有数据列组合存在于磁盘上。这会使包含涉及 YEAR 和/或 MONTH 的谓词的查询性能有所提高。在创建表之后，不可改变维。

(9) 单击“下一步”按钮进入“约束”对话框，跳过不做设置。

**注意：**表检查约束是对表中的每一行制定取值约束的规则。每当插入数据或更新数据时，就检查它们。

(10) 在“总结”对话框中，单击“显示 SQL”按钮，则会出现如下的 SQL 语句。其中，约束名会有所不同。

```
CONNECT TO MUSICDB;
CREATE TABLE USER. ARTISTS ( ARTNO SMALLINT NOT NULL, NAME VARCHAR (50), CLASSIFICATION
CHARACTER (1) NOT NULL, BIO CLOB (100 K ) LOGGED COMPACT, PICTURE BLOB (500 K ) NOT LOGGED
COMPACT, CONSTRAINT CC1218533820171 PRIMARY KEY ( ARTNO ) ) IN DMS01 INDEX IN DMS02 LONG
IN DMS03;
CONNECT RESET;
```

(11) 单击“完成”按钮创建该表。

## 2) 创建表

利用 Create Table 语句创建表 albums, 如图 3-6 所示。

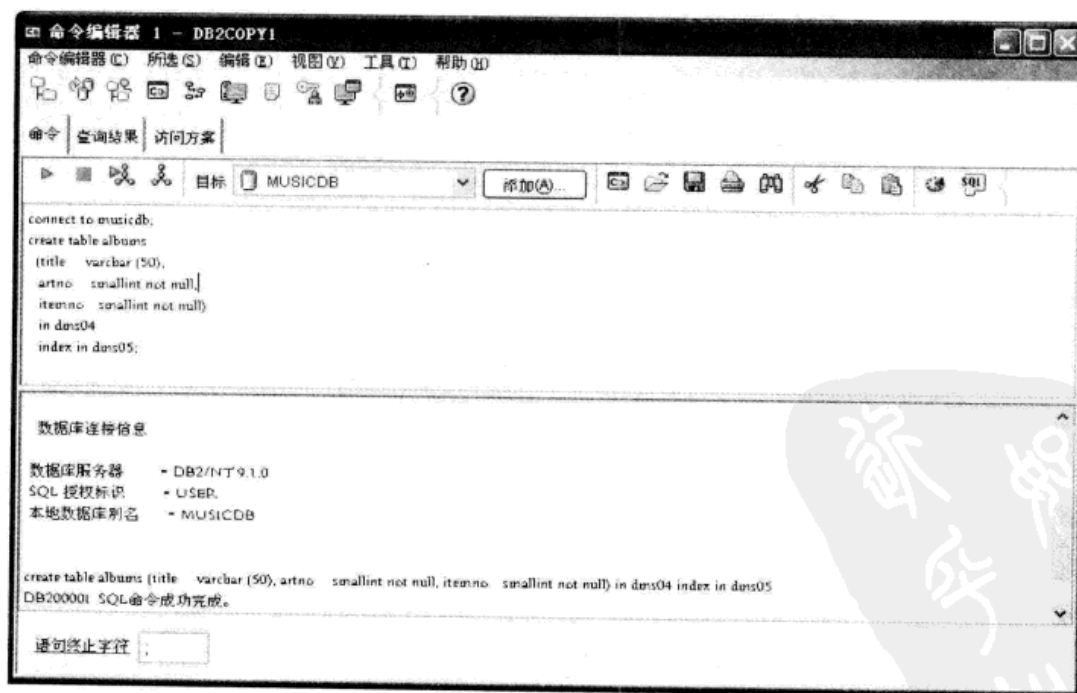


图 3-6 使用 Create Table 语句创建表

在“命令编辑器”中输入下列语句，创建数据表 albums 需要先连接数据库，即 connect to musicdb)。

```
connect to musicdb;
create table albums
(title      varchar (50),
artno      smallint not null,
itemno     smallint not null)
in dms04
index in dms05;
```

在控制中心左边的对象树中,右击 MUSICDB 下的“表”,在弹出菜单中选择“刷新”选项,再在右边的内容面板中找到 artists 表和 albums 表,查看这两个表。如果表不符合要求,则将其删除,然后再返回上面步骤重新建立。

### 3) 导入脚本文件创建其余表

在“命令编辑器”窗口中,单击“所选”下的“打开”选项,在“查找”栏中选定 cf23 文件夹所在的盘符,之后导入 CRTABLES 的脚本文件创建其余表。在执行之前,先查看脚本内容,注意表 STOCK、CONCERTS、REORDER 将分别把表数据和索引放在哪些表空间。按 Ctrl+Enter 组合键执行脚本文件。

### 4) 查看所建表的信息

在控制中心左边的对象树中,右击 MUSICDB 下的“表”,在弹出菜单中选择“刷新”选项,然后再在右边内容面板中查看脚本所创建的表,确保这些表都建立得正确。

要查看表的详细信息,可以在内容面板中右击所要查看的表,在弹出菜单中选择“改变”。使用此方法查看 albums 表,如图 3-7 所示。

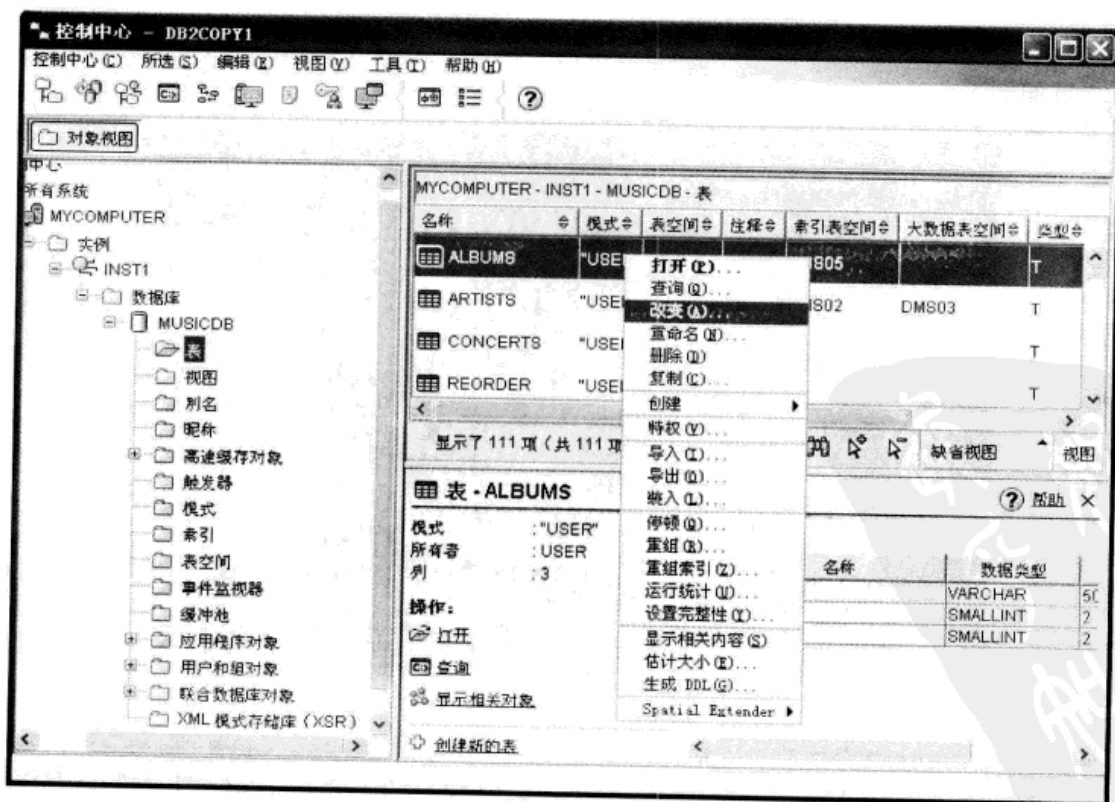


图 3-7 查看表信息

## 5) 导入脚本 GRANTS 完成权限授予操作

通过在命令中心中导入执行 GRANTS 脚本,将 USER 模式下所有表的 select 权限授予 public,并在脚本执行完毕后检查权限授予是否成功。

GRANTS 脚本内容如下:

connect to musicdb; 连接数据库,将 USER 模式下所有表的 select 权限授予 public。

```
grant select on table artists to public;
grant select on table albums to public;
grant select on table stock to public;
grant select on table concerts to public;
grant select on table reorder to public;
```

注意查看该脚本的具体内容,以及相应的 SQL 语句的写法。

## 6) 查看模式 USER 所有表的列信息

表的列信息可以通过 SYSCAT.COLUMNS 视图查询到。在 DB2 命令行处理器输入如下指令,执行 X:\cf23 下的 tabchk.sql 脚本,并将输出结果重定向到 tabchk.file 中。

具体执行过程如下:

打开“开始”→“程序”→IBM DB2→“命令行工具”→“命令窗口”,输入:

X:

Cd \cf23(数据文件所在目录)

db2 connect to musicdb

db2 -tf tabchk.sql > tabchk.file(将输出结果重定向到 tabchk.file 中)

more tabchk.file

结果如图 3-8 所示。

```

C:\Program Files\IBM\SQLLIB\BIN>c:
C:\Program Files\IBM\SQLLIB\BIN>cd \cf23
C:\cf23>db2 connect to musicdb

      数据库连接信息

数据库服务器      = DB2/NT 9.1.0
SQL 授权标识      = USER
本地数据库别名    = MUSICDB

C:\cf23>db2 -tf tabchk.sql>tabchk.file
C:\cf23>more tabchk.file

TABNAME                                COLNAME
                                     TYPENAME
-----                                -

```

图 3-8 查看表的列信息

检查输出文件 tabchk.file 与 tabchk.master 内容上的差别(须确保两文件的内容没有任何差别)。首先在命令提示符下输入指令:



```
fc /w tabchk.file tabchk.master | more
```

如果输出消息显示“FC: 找不到相异处”, tabchk. file 与 tabchk. master 相一致。如果输出显示两个文件某些行不相同, 则应仔细检查相应错误, 删除建立不当的表, 并返回到上一步重新建立, 如图 3-9 所示。

```
C:\cf23>
C:\cf23>fc /w tabchk.file tabchk.master | more
正在比较文件 tabchk.file 和 TABCHK.MASTER
FC: 找不到相异处
```

图 3-9 判断表建立的正确性

## 7) 查看模式 USER 表空间的信息

(1) 某个表所相关的表空间信息可以通过 SYSCAT. TABLES 视图来获得。输入一条 SQL 语句来从 SYSCAT. TABLES 获取模式 USER 下所有表的表名(TABNAME)、存储该表数据的表空间(TBSPACE)、存储该表索引的表空间(INDEX\_TBSPACE)信息, 并将结果集按表名排序。完整的 SQL 语句为:

```
select tabname, tbpace, index_tbpace from syscat.tables where tabschema = 'USER' order
by tabname
```

X: \cf23 下的脚本文件 tbschk. sql 包含该 SQL 语句, 在 DB2 命令窗口中进入 x: \cf23 目录下, 输入:

```
db2 -tf tbschk.sql > tbschk.file
more tbschk.file
```

(2) 检查输出文件 tbschk. file 与 tbschk. master 内容上的差别。

```
fc /w tbschk.file tbschk.master | more
```

如果输出消息显示“FC: 找不到相异处”, tbschk. file 与 tbschk. master 相一致。如果输出显示两个文件某些行不相同, 则应仔细检查相应错误, 删除建立不当的表, 并返回到上一步重新建立。

## 2. 创建索引

1) 在 stock 表的 itemno 列上建立名为 item 的索引

在控制中心的对象列表中, 右击 MUSICDB 下的“索引”项, 并在弹出菜单中选择“创建”选项; 然后选择“索引”选项, 在“创建索引”对话框的“目标表”选项卡中, 按下列要求填入相关信息。其中, 表模式为 USER, 表名为 stock。

单击“下一步”按钮, 进入“名称”页面, 填写相关信息, 索引模式为 USER, 索引名为 item;

单击“下一步”按钮, 进入“列”页面, 在“可用的列”列表中选择 itemno 并单击“>”按钮将该列加入到“选择的列”中。然后单击“下一步”按钮, 进入“选项”页面, 在“指定索引选项”采用默认设置, 然后单击“完成”按钮, 如图 3-10 所示。

**注意:** 在图 3-10 中, 如果选中“强制唯一性”复选框, 表示 ITEMNO 中的数据将保持唯一性。

“包含列”是 DB2 提供的一个特性, 用来在索引中存储额外的数据列, 从而加快数据访问, 尤其是在只扫描索引的时候。在这里没有包含列。

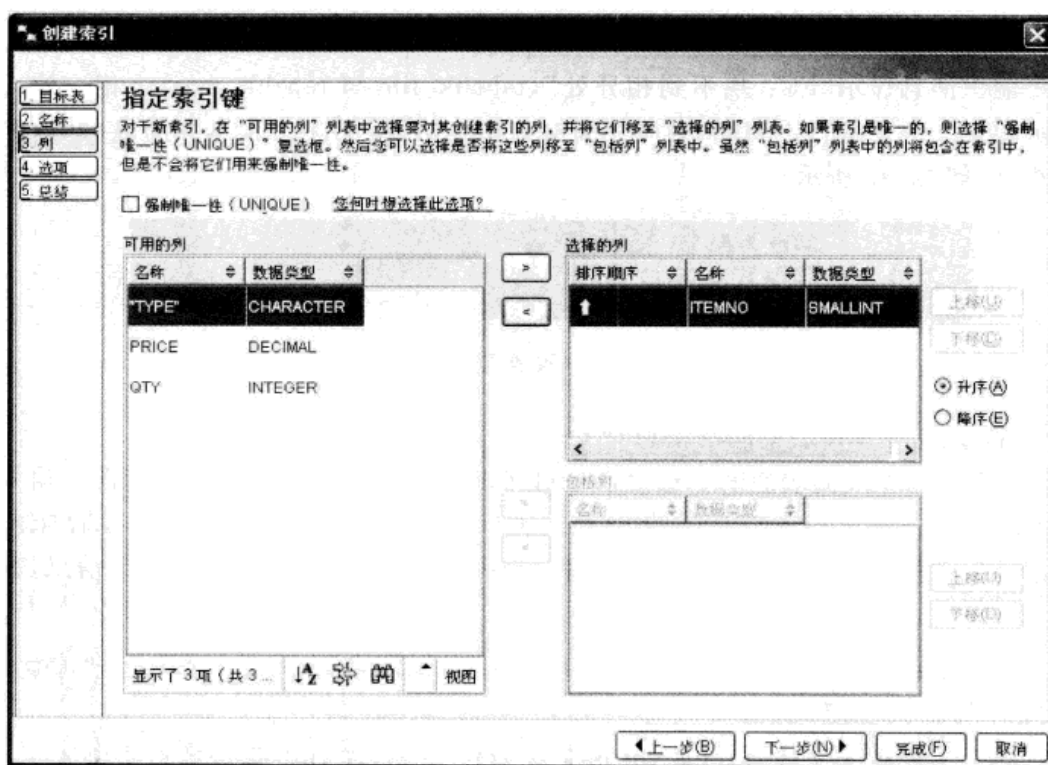


图 3-10 指定索引键

## 2) 为 albums 表的 itemno 列建立唯一索引

其中，表模式为 USER，表名为 albums；索引模式为 USER，索引名为 itemno。

在“可用的列”中选择 ITEMNO 并单击“>”按钮将该列加入到“选择的列”中，并且选中“强制唯一性 (UNIQUE)”复选框，然后单击“确定”按钮。

有关索引的信息可以通过 SYSCAT. INDEXES 视图获取。

### 注意：

如果把数据库比作一本书，那么表的索引就是这本书的目录，可见建立索引是加快查询操作速度的有效手段。在定义基本表时，通常还要定义索引（一个或多个），以提供多种存取路径。主要的索引类型有聚簇索引和唯一索引。

聚簇是根据码值找到数据的物理存储位置，从而达到快速检索数据的目的。聚簇索引的顺序就是数据的物理存储顺序，叶节点就是数据节点。非聚簇索引的顺序与数据物理排列顺序无关，叶节点仍然是索引节点，只不过有一个指针指向对应的数据块。一个表最多只能有一个聚簇索引。但是建立聚簇索引的空间耗费比较大，对以后的增加删除的影响比较大，因此，对于经常变更的列不宜建立聚簇索引。

唯一索引表示此索引的每一个索引值，只对应唯一的数据记录，即不允许表中不同的行在索引列上取相同值。

## 3. 建立视图

### 1) 在控制中心中建立一名为 music 的视图

在控制中心的对象列表中，右击 MUSICDB 下的“视图”项，在弹出菜单中选择“创建”选项。

选择视图的模式为 USER 视图名一栏中输入 music。

单击 SQL 语句框右边的“清除”按钮,并输入:

```
as select title, classification, name from user.albums, user.artists where user.artists.artno  
= user.albums.artno;
```

单击“确定”按钮,如图 3-11 所示。



图 3-11 创建视图

## 2) 利用 Create View 语句创建视图 inventory

在“命令编辑器”中,输入下列语句,建立视图 inventory。

```
connect to musicdb;  
create view inventory (type, itemno, totcost, totqty)  
as select type, itemno, sum (price * qty), sum(qty)  
from stock group by type, itemno;
```

### 注意:

视图是关系数据库系统提供给用户从多种角度观察数据库中数据的重要机制。

视图是从一个或几个基本表(或视图)导出的表,与基本表不同,是一个虚表。也就是说,数据库中只存放视图的定义,而不存放视图对应的数据,这些数据仍然存放在原来的基本表中。基本表中的数据发生变化,从视图中查询出的数据也就随之变化了。从这个意义上讲,视图就像是一个窗口,透过它可以看到数据库中自己感兴趣的数据及变化。

3) 查看视图信息

(1) 在控制中心的对象树中,右击 MUSICDB 下的“视图”项,并在弹出菜单中选择“刷新”选项,然后可见在右边的内容面板中列出了新建立的视图,如图 3-12 所示。

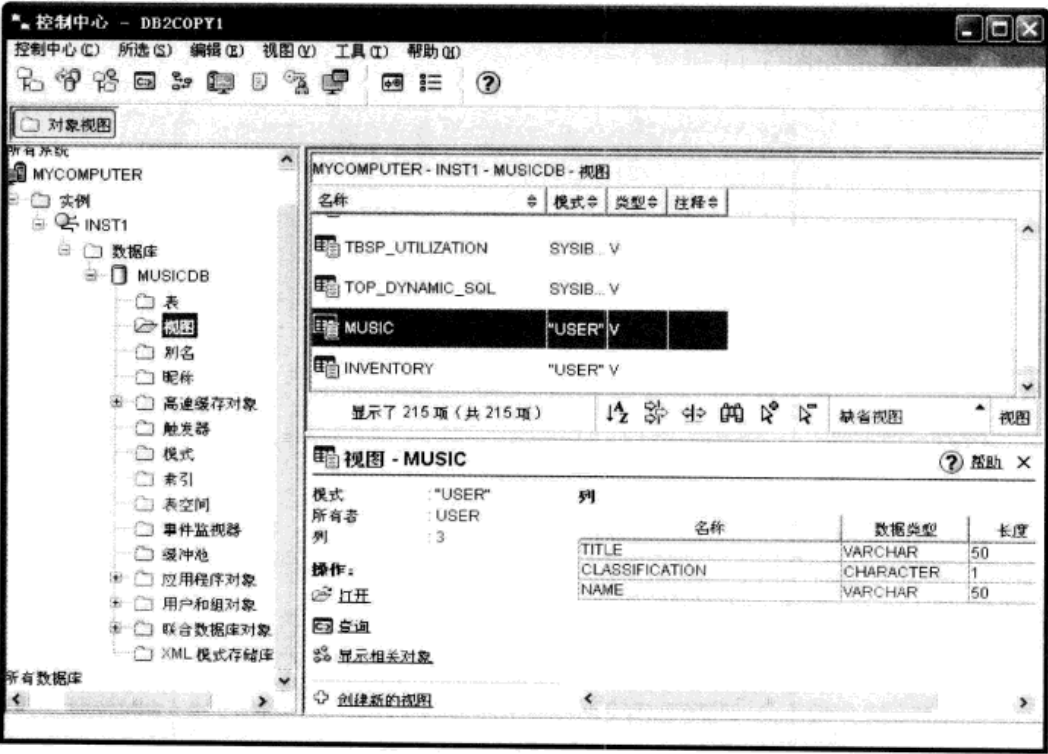


图 3-12 视图结果查询

(2) 视图的相关信息可以通过 SYSCAT.VIEWS 和 SYSCAT.TABLES 来获取。List tables 同样可以列出数据库中的视图(类型为 V)。

在“命令编辑器”中输入如下命令,查看新建立的视图是否列出:

```
connect to MUSICDB;  
list tables for user;
```

或

```
connect to MUSICDB;  
select viewschema, viewname from syscat.views where definer = user;
```

4. 建立别名

1) 在控制中心为 artists 表建立别名 singers

在控制中心为 artists 表建立别名 singers,为 reorder 表建立别名 emptystock。在控制中心的对象列表中,右击 MUSICDB 下的“别名”项,在弹出菜单中选择“创建”选项,如图 3-13 所示。

2) 通过 SYSCAT.TABLES 视图查看别名的相关信息

别名的相关信息可以通过视图 SYSCAT.TABLES 来获取,此外 list tables 命令也能获取当前连接的用户下的别名列表。

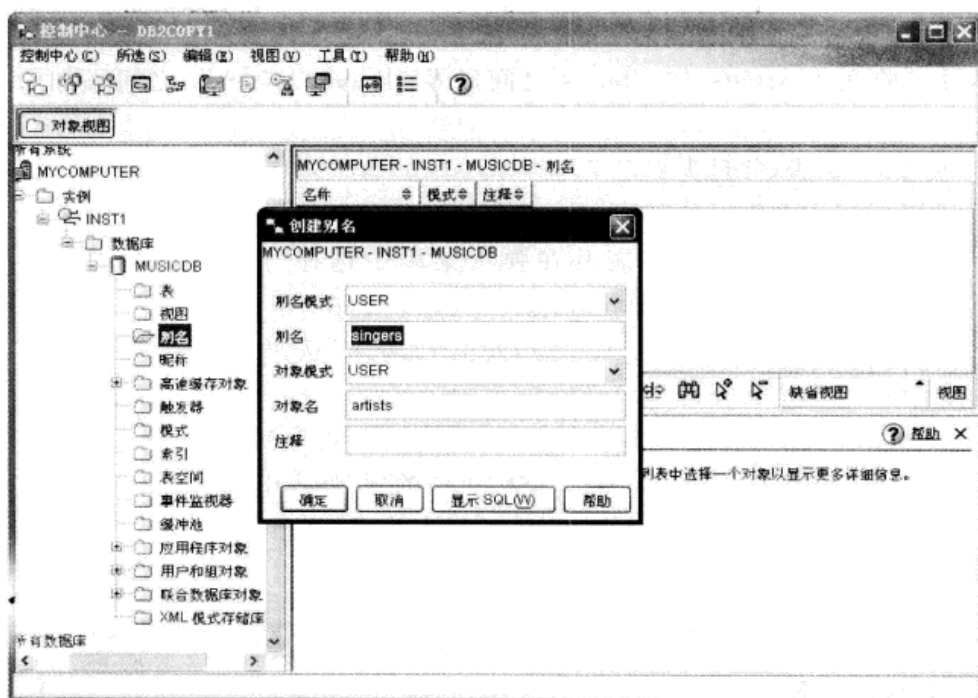


图 3-13 建立别名

输入如下命令列出别名列表：

```
connect to MUSICDB;
list tables for user; (如图 3-14 所示)
```

或

```
select tabname, type from syscat.tables where tabschema = user;
```

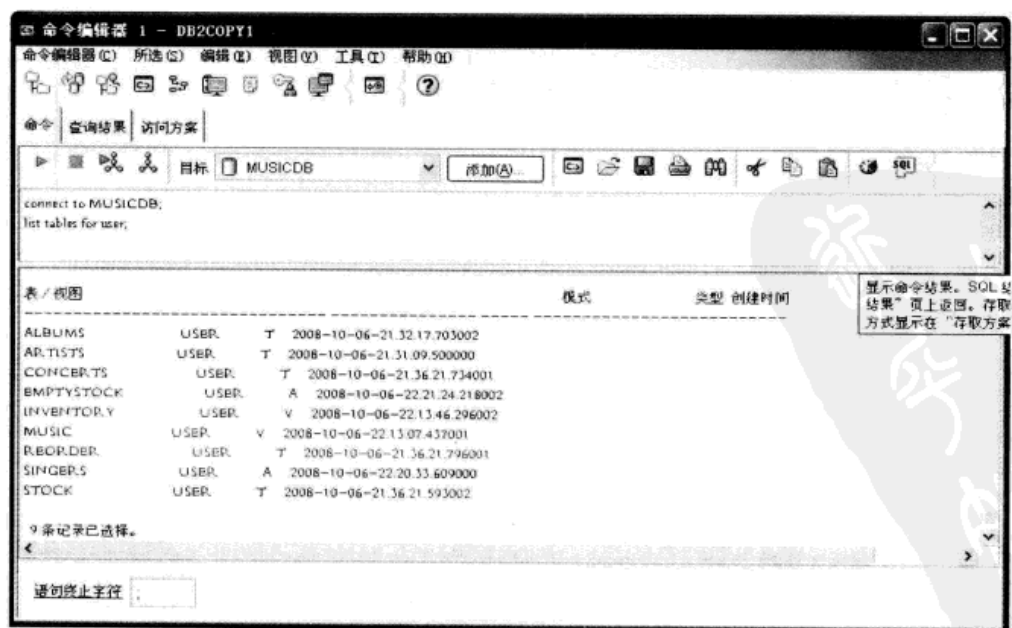


图 3-14 别名结果显示

### 5. 添加参照完整性约束条件

本部分实验在表 artists 与 albums 之间以及 albums 与 stock 之间添加参照完整性约束。

1) 修改 albums 表, 并且为它定义与表 artists 的参照完整性约束

(1) 为 albums 表在 itemno 列上建立主键。

在控制中心中, 右击 albums 表并在弹出菜单中选择“改变”选项, 再在“改变表—albums”对话框中选择“键”标签, 单击“添加主键”按钮, 在“定义主键”对话框中的“可用的列”列表中选择 itemno, 然后单击“>”按钮, 并单击“确定”按钮。

(2) 为 albums 表定义外键。

单击“添加外键”按钮, 在“添加外键”对话框中, 表模式为 USER, 父表名为 artists, 注意这时主键框内显示该表主键为 artno, 再选择“可用的列”列表中 artno, 单击“>”按钮使该列成为外键, 选择“删除时”一栏为 CASCADE, 约束名为 fkartno(即为外键名), 单击“确定”按钮, 如图 3-15 和图 3-16 所示。



图 3-15 添加外键

2) 使用脚本 CRR1 修改 STOCK 表并为其定义参照完整性约束

在命令编辑器脚本模式下执行 CRR1 脚本, 该脚本修改 STOCK 表并为其定义参照完整性约束(在运行脚本前, 请仔细查看相关 SQL 语句的书写方法)。



图 3-16 添加主外键后的结果显示

CRR1 脚本内容：

```
connect to musicdb;
alter table stock
foreign key fkitemno (itemno) references albums on delete restrict;
```

首先连接数据库,为 STOCK 添加外键,itemno 列设置为外键,父表是 albums,删除规则设置为 restrict,约束名为 fkitemno。

3) 通过 SYSCAT. REFERENCES 视图查看参照完整性约束的相关信息

在命令编辑器中输入如下 SQL 语句获取 USER 模式下所有参照完整性约束的名称、所在表名、参照表的模式、参照表名、删除规则、外键所在列名、主键所在列名：

```
select constname, tabname, reftabschema, reftabname, deleterule, fk_colnames, pk_colnames
from syscat.references
where tabschema = 'USER';
```

完整性约束相关信息,如图 3-17 所示。

4) 通过 SYSCAT. TABLES 视图查看参照完整性约束的相关信息

在命令中心中输入如下 SQL 语句,其中 CHILDREN 为参照该表的所有表的个数, PARENTS 为该表所要参照的表的个数：



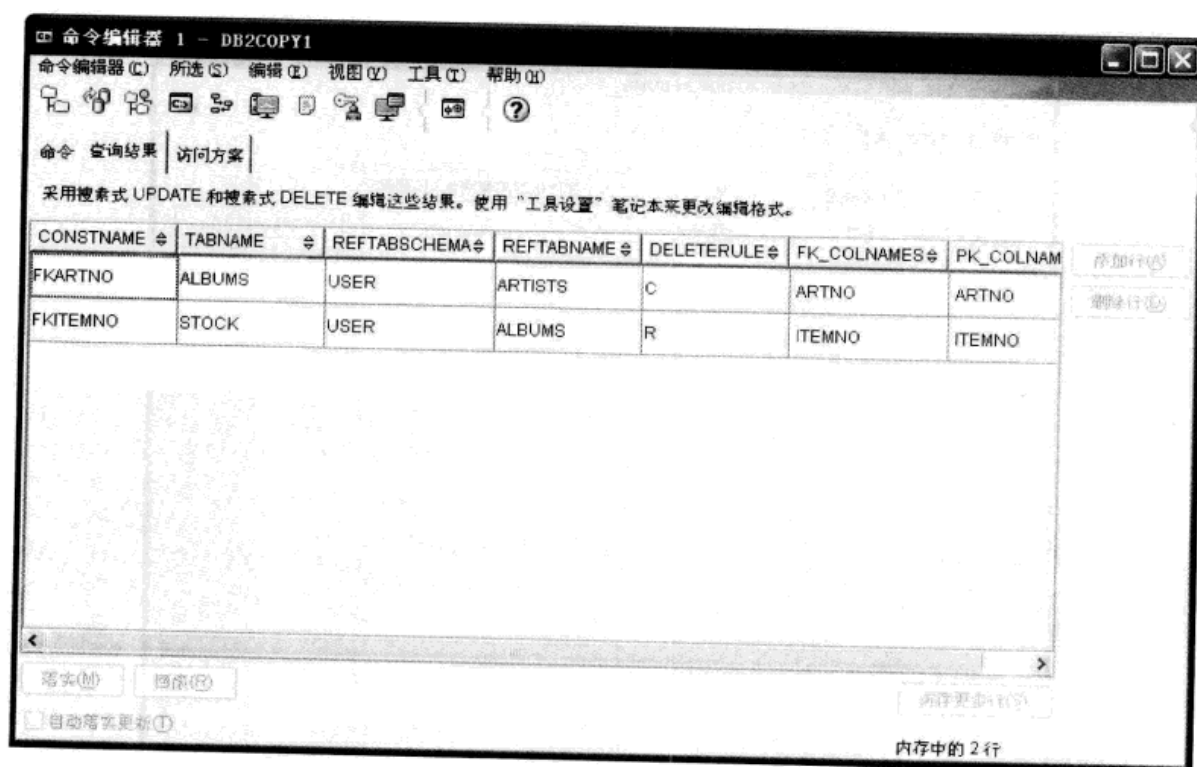


图 3-17 查看参照完整性约束的相关信息

```
select tabname, parents, children from syscat.tables where tabschema = 'USER' and (parents > 0 or children > 0) order by 2,3 desc
```

#### 5) 参照完整性约束定义结果检查

在 DB2 命令窗口中执行脚本 richk.sql, 通过 SYSCAT.TABLES 视图获得当前连接的用户模式参照完整性约束的相关信息, richk.sql 的内容如下:

```
select substr(tabname, 1, 18), parents, children
from syscat.tables
where tabschema = 'USER' order by tabname
```

```
db2 -tf richk.sql > richk.file
more richk.file
```

然后检查该输出文件的内容与 richk.master 有无差别:

```
fc /w richk.file richk.master | more
```

如果输出显示两文件内容存在差异, 则应仔细检查相关信息后重新修改表。

#### 注意:

数据库完整性指数据库中数据的正确性和相容性, 是应防止数据库中存在不符合语义规定的数据和防止因错误信息的输入输出造成无效操作或错误信息而提出的。数据库完整性由各种各样的完整性约束来保证, 主要分为 4 类: 实体完整性、域完整性、参照完整性、用户自定义的完整性。

实体完整性规定表的每一行在表中是唯一的实体。

域完整性是数据库表中的列必须满足某种特定的数据类型或约束。其中,约束又包括取值范围、精度等规定。

不同的关系数据库系统根据其应用环境的不同,往往还需要一些特殊的约束条件。用户自定义的完整性即是针对某个特定关系数据库的约束条件,反映某一具体应用所涉及的数据必须满足的语义要求。

现实生活中的实体之间是有联系的,这种联系在数据库中是通过建立主外键表现出来的。参照完整性就是用来确保两个表的主键和外键的数据对应一致。它确保了有主键的表中对应其他表的外键的行存在,即保证了表之间的数据的一致性,防止了数据丢失或无意义的数据在数据库中扩散。

DB2 数据库管理器基于参照完整性规则实施系统间的参照约束,这些规则包括 INSERT 规则、DELETE 规则、UPDATE 规则。

INSERT 规则:可以随时在父表中插入一行,而不必在从属表中执行任何操作。但是,不能在从属表中插入一行,除非在该父表中有一行其主键值等于要插入行的外键值,或该外键值是空(前提是该外键可以为空)。

DELETE 规则:可是随时在从属表中删除一行。当从父表中删除一行时,DB2 需要检查该行在从属表中是否有任何从属行的外键值与父表中待删除元组的主键值相同。若找到这样的元组,从属表可以按照下列四种删除规则来处理。

CASCADE:删除父表中的一行将自动删除从属表中的所有相关行。

RESTRICT:DB2 若找到任何从属行,则阻止删除父表中的删除操作。

NO ACTION:如果对于父表的删除操作完成后,在从属表中存在没有父表中主键值与之相同的外键值时,则 NO ACTION 将导致对父键值的删除操作被拒绝。

SET NULL:确保删除父表中某行时将从属表中相同的外键值设置为 NULL(如果该列允许空值的话),该行的其他部分保持不变。

UPDATE 规则:当更新一个从属表中的外键值,而该外键不为空时,它必须与该关系中的父表的主键的某个值相匹配。当更新父键某列中的值时,可以指定两条规则:RESTRICT 和 NO ACTION。如果执行了这两条准则之后,都没有实现对于父表主键值的更新,就要采取下面的方法:首先将和待更改主键值相同的从属表外键值所对应的元组删除,之后,更新父表,将删除的元组对应的外键值修改为父表主键更改后的值,最后将修改好的从属表元组添加到子表中。

## 6. 为 stock 表添加检查约束

### 1) 修改 stock 表并为该表添加检查约束 cctype

该约束确保 stock 表 type 字段的取值必须是字符'D'、'C'、'R'之一。

在控制中心中,右击 stock 表,在弹出菜单中选择“改变”选项,在“改变表—STOCK”对话框中选择“检查约束”标签,单击“添加”按钮,再在检查条件框中输入:

```
type in ('D', 'C', 'R')
```

约束名一栏输入 cctype,单击“确定”按钮关闭“添加检查约束”对话框,再单击“确定”按钮关闭“改变表—STOCK”对话框,如图 3-18 所示。

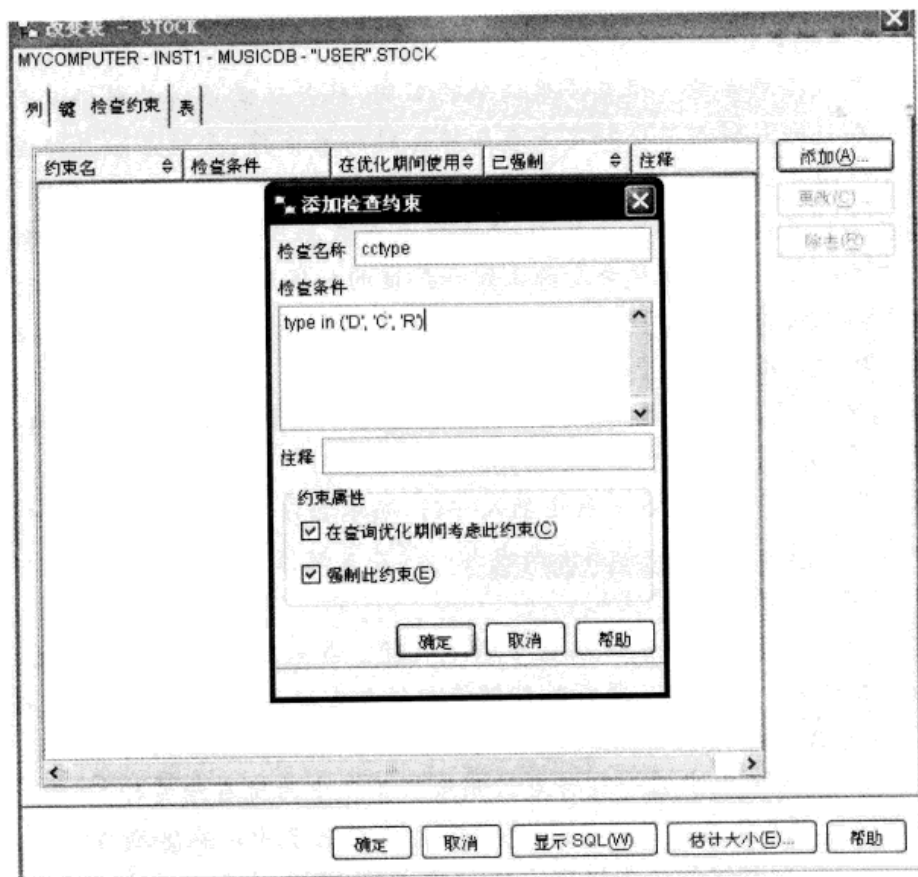


图 3-18 修改表

**注意：**

实现域完整性的一个主要方法是设置 CHECK 约束。这一部分进行的添加检查约束的操作就是图形化实现 CHECK 子句。使用 CHECK 子句可以对域做出强有力的限制,允许数据模式的设计者指定一个谓词,对类型属于该域的变量所赋的任意值都必须满足该谓词。CHECK 子句也可以用于限制域不允许空值或者使用 IN 子句可以限制域只包含指定的一组值。复杂的 CHECK 条件在希望确保数据完整性的时候是很有用的,但要谨慎使用,因为检测它们的开销会很大。

## 2) 通过系统视图查看约束的相关信息

检查约束的相关信息可通过 SYSCAT.CHECKS、SYSCAT.COLCHECKS、SYSCAT.TABCONST 和 SYSCAT.TABLES 这些视图获取。

在命令中心中输入：

```
select constname, tabname, colname from syscat.colchecks
```

## 3) 使用 SQL 语言查看约束的相关信息

在命令中心中输入如下 SQL 语句,列出的约束中,TYPE 为 K 表示是一个检查约束,TYPE 为 P 表示是一主键,TYPE 为 F 表示是一外键,如图 3-19 所示。

```
select constname, tabname, type from syscat.tabconst
```

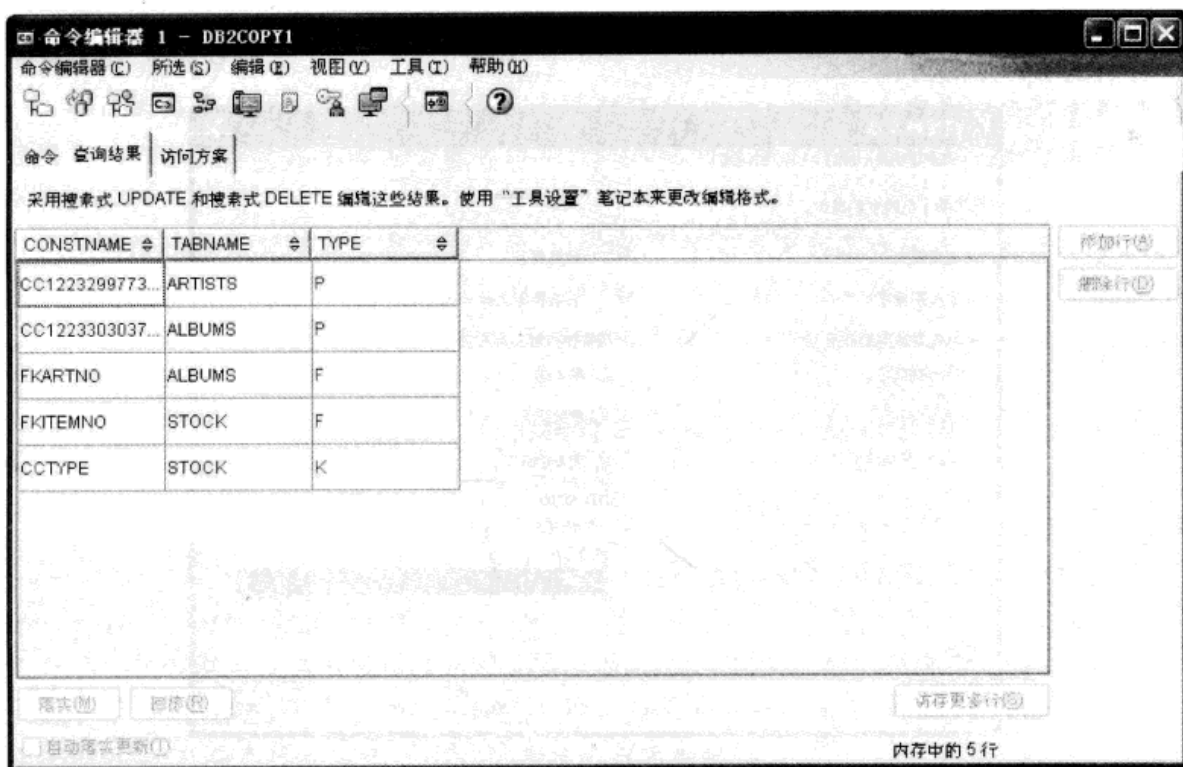


图 3-19 检查约束结果查询

#### 4) 执行脚本 ckchk.sql

DB2 命令窗口中输入如下命令执行脚本 ckchk.sql,并将输出结果重定向到 ckchk.file:

```
Db2 -tf ckchk.sql > ckchk.file
More ckchk.file
```

然后检查该输出文件的内容与 ckchk.master 有无差别:

```
fc /w ckchk.file ckchk.master | more
```

### 7. 为 reorder 表创建触发器

#### 1) 要建立的触发器应具备的性质

- 触发器名为 reorder;
- 触发时机为更新 stock 表某些记录的 qty 字段值后该字段值小于或等于 5;
- 新记录变量 new 通过 n 来引用;
- 该触发器将新记录变量的 itemno 值和当前时间戳插入到 reorder 表;
- 应为 for each row mode db2sql 触发器。

具体执行过程如下:

右击控制中心 MUSICDB 下的“触发器”项,在弹出菜单中选择“创建”选项。在创建触发器对话框中,输入触发器模式选为 USER,表或视图模式选为 USER,触发器名为 reorder,表或视图名称选为 stock,触发操作的时间选为“之后”选项,导致执行触发器的操作设为“更新列”并选择 qty 列,如图 3-20 所示。然后选择“已触发的操作”标签,在“新行的相关名”文本框中输入 n,在触发操作框内输入如下 SQL 语句:

```
when (n.qty <= 5)
insert into reorder values (n.itemno, current timestamp)
```

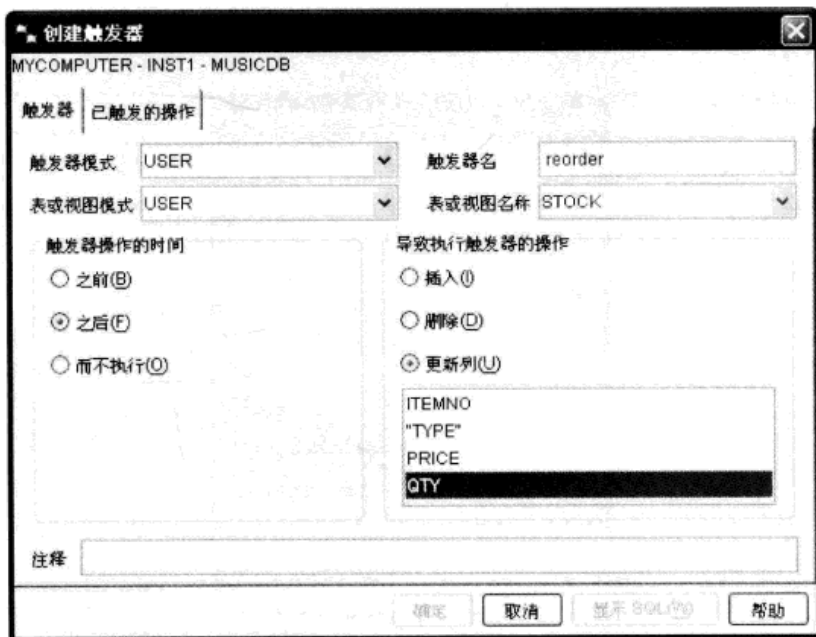


图 3-20 创建触发器

单击“显示 SQL”按钮查看完整的 SQL 语句。

单击“确定”按钮,然后在控制中心查看该触发器是否列出。

2) 通过系统视图查看触发器的相关信息

触发器的相关信息可以通过 SYSCAT. TRIGGERS 和 SYSCAT. TRIGDEP 获取。

在命令中心中输入如下 SQL 语句:

```
select trigename, tabname, trigenvent from syscat.triggers
```

其中, trigenvent 为 I 表示触发操作为 insert, D 表示触发操作为 delete, U 表示触发操作为 update。

在命令中心中输入下列 SQL 语句:

```
select trigename, btype, bschema, bname from syscat.trigdep
```

**注意:**

触发器是一种特殊类型的存储过程,当对数据库作修改时,自动被系统执行。要设置触发器机制,必须满足两个要求,指明什么条件下触发器被执行。它被分解为一个引起触发器检测的事件和一个触发器执行下去必须满足的条件,指明触发器执行时采取的动作。

触发器的主要作用是实现由主键和外键所不能保证的复杂的参照完整性和数据的一致性。除此之外,触发器还有其他许多不同的功能,强化约束、跟踪变化、级联运行和存储过程的调用。

触发器能够定义成按如下两种方式之一激活。

① 前触发器,在引发触发器的 SQL 语句执行之前对受语句影响的每一行都触发一次。

因此,在新数据插入到表中之前或表中数据被更新之间,触发器将检查这些新数据。

② 后触发器,可对受语句影响的每一行都触发一次,或在语句成功完成以后执行一次触发操作,这依赖于所定义的粒度。因此,触发器在一致的状态下(所有的事务都已完成)查看该表。触发器的另一重要特点是,触发器可以触发其他触发器(或它本身)或约束,这称做级联触发器。在触发器的执行期间,根据触发器的性质(前触发器或后触发器),触发器能访问新数据值或老数据值。通过使用触发器,用户能够获得下述好处:

理解触发器最好的方法是看一些起作用的触发器。下面来看一个触发器的例子。上面创建触发器的操作如果使用 SQL 如下来实现。

```
CREATE TRIGGER "USER".REORDER AFTER UPDATE OF QTY ON "USER".STOCK REFERENCING NEW AS n
FOR EACH ROW MODE DB2SQL
WHEN ( n.qty <= 5 )
insert into reorder values(n.itemno,current timestamp);
```

本例中 CREATE TRIGGER 语句创建了一个名为 reorder 的后触发器。

CREATE TRIGGER reorder 指定触发器的名字为 reorder。在 DB2 中,该触发器的名字可以在编目表 SYSCAT.TRIGGERS 和 SYSCAT.TRIGDEP 查找到。

AFTER 指定被触发的动作在表 Stock 的数据被更新之后发生。

如果指定 NO CASCADE BEFORE 选项,触发的动作会在更新发生之前执行;同时,NO CASCADE 指定触发动作不激发其他的触发器;激活触发器的 SQL 操作包括 INSERT,UPDATE,DELETE 等,本例为 UPDATE。

OF qty(表 Stock 中的列名)控制触发的粒度。触发粒度通常是表中的一列,而不是整个表;ON Stock(表名)指定应用触发器的基本表。

REFERENCING NEW AS n(n 是更新后的表的相关名)提供一个用于事务处理的相关名,该名要用在触发的 SQL 语句之前。更新前的表相关名可用 REFERENCING OLD AS M 定义,两者也可以用一个语句完成定义;REFERENCING NEW AS N OLD AS M。

FOR EACH ROW/STATEMENT 指定触发器在每行还是整个语句执行完后,执行相应的操作。

MODE DB2SQL 是 DB2 系统要求写的参数;WHEN(n.qty <= 5)(查找条件)指定当该条件满足时,触发器中定义的动作开始执行。

INSERT INTO reorder values(n.itemno, current timestamp)(SQL 过程语句)是当触发条件成立时触发器要执行的动作。

触发器可以用 DROP TRIGGER 语句删除。

前触发器与后触发器的使用目的以及它们的触发器本身中可以含有的 SQL 指令都不一样,上面已经创建了后触发器,读者现在可以试着自己创建前触发器。

## 8. 本数据库建立的表关系图

通过实验 4 创建了所有主体数据库对象,搭建起了数据库模式结构,如图 3-21 所示。

## MUSICDB: Data Model and Objects

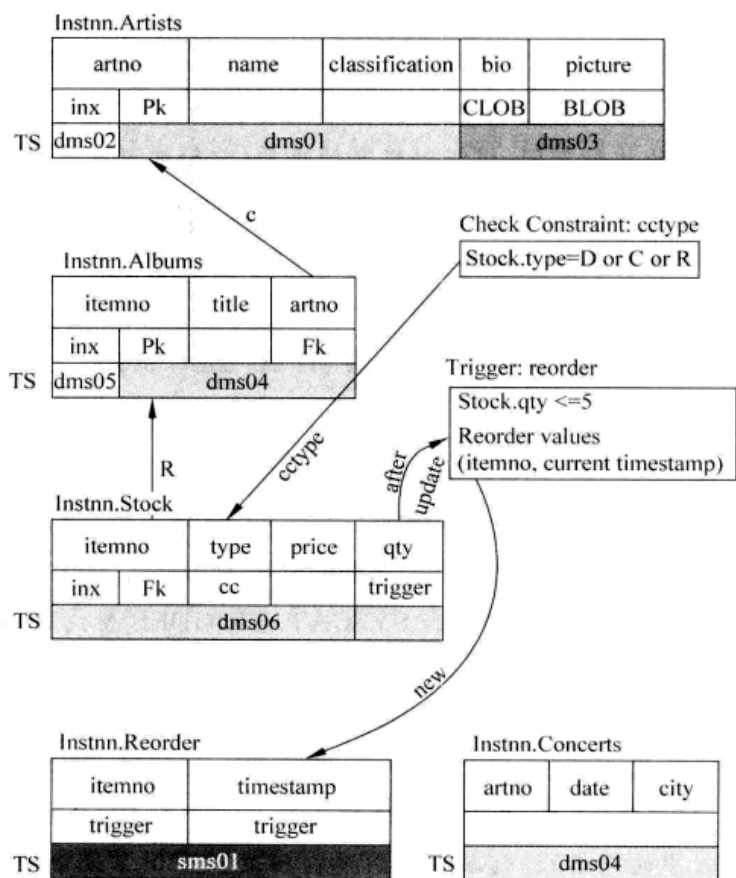


图 3-21 实验关系图

## 实验 4

## 移动与操纵数据

### 实验任务

- 学习 Insert、Select 和 Update 等 SQL 语句的应用；
- 使用 import 工具从一文件读取数据写入表；
- 使用 load 工具从一文件读取数据快速写入表；
- 学习如何建立异常表以及异常表的应用；
- 学习管理检查约束的应用，触发器的使用以及表的检查暂挂状态。

### 实验内容

#### 1. 数据插入、删除

1) 利用 SQL 语句(insert 命令)插入数据并查看结果

(1) 将下列数据插入到表 albums 中(一次插入一行),向 albums 表中待插入的数据如表 4-1 所示。

注意:

在“命令编辑器”中,输入 SQL 语句:insert into user. albums values('Greatest Hits', 1,1); insert into user. albums values('Voice in the Wind', 2,5);由于不满足前面定义的外键约束

(见图 3-21),则返回“插入值或更新值不等于父表中的父键的任何值,插入不成功”的提示(如图 4-1 所示)。在“控制中心”中,右击表 albums,选择“查询”选项,查看表中结果,albums 仍然是空表。

(2) 将下列数据插入到表 artists(向 artists 表中待插入的数据如表 4-2 所示)中。

表 4-1 albums 表待插入的数据

TITLE	ARTNO	ITEMNO
Greatest Hits	1	1
Voice in the Wind	2	5

表 4-2 artists 表待插入数据

ARTNO	NAME	CLASSIFICATION	BIO	PICTURE
99	Double Dare	R		
	Patti& Cartwheels	S		
1	Alabama	C		
2	Bogguss, Suzy	S		
3	Black, Clint	C		
	Brooks, Garth	C		



续表

ARTNO	NAME	CLASSIFICATION	BIO	PICTURE
5	Chapin-Carpenter, Mary	R		
6	Gill, Vince	C		
7	Jackson, Alan	C		
8	Judds	C		

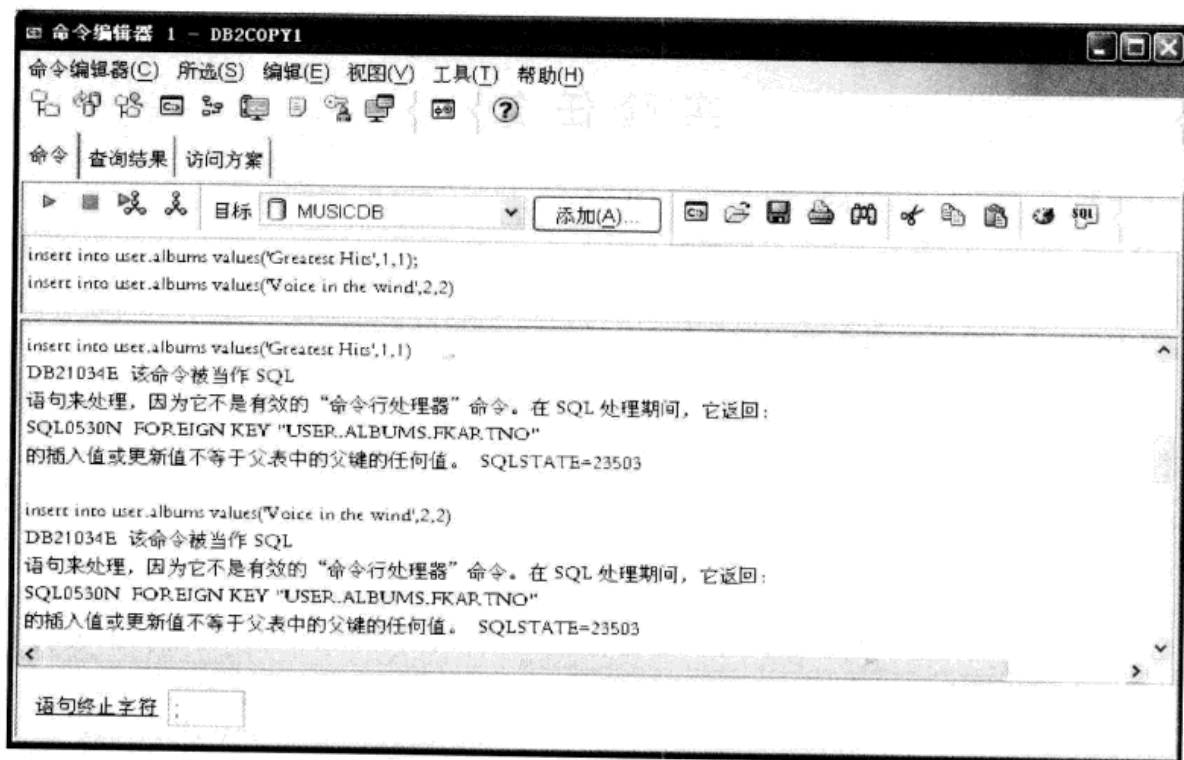


图 4-1 利用 insert 命令向 albums 表插入数据

在“控制中心”中,右击表 ARTISTS,选择“打开”→“添加行”选项,将上面数据依次添加到 ARTISTS 表中,选择“落实”,如图 4-2 所示。第 2 行和第 6 行数据插入时,由于不允许将 NULL 值赋给 NOT NULL 列,所以插入不成功。在“控制中心”,右击表 ARTISTS 项,选择“查询”选项,显示查询过程,如图 4-3 所示。

#### 注意:

与以前版本不同,db2 9 不仅提供了 insert 命令的插入方式;也提供了窗口录入方式,在录入过程中进行完整性约束检查,一旦不满足则插入不成功。

(3) 重新进行(1)的插入操作,并查看 albums 表中的插入结果,这时插入成功,如图 4-4 所示。

(4) 将下列数据一次插入到表 albums 中,向 albums 表中待插入的数据,如表 4-3 所示。



图 4-2 向 ARTISTS 表添加数据



图 4-3 向 ARTISTS 表插入数据的结果

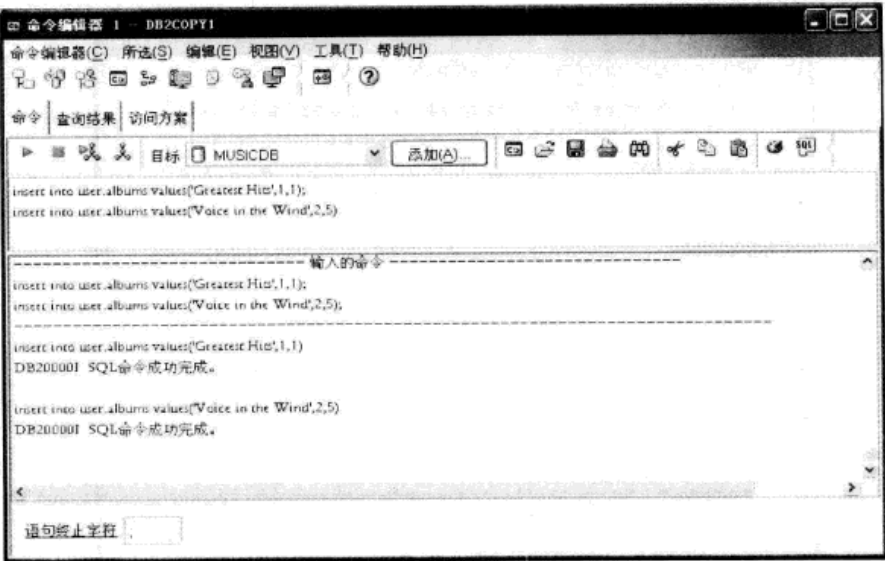


图 4-4 利用 insert 命令向 albums 表插入数据

表 4-3 albums 表待插入的数据

TITLE	ARTNO	ITEMNO
American Pride	1	
Something Up My Sleeve	2	4
Put Yourself in My Shoes	5	7

在“命令编辑器”中输入 insert into user. albums values ( ' American Pride ' , 1 ) , ( ' Something Up My Sleeve ' , 2 , 4 ) , ( ' Put Yourself in My Shoes ' , 5 , 7 ) ; 由于 itemno 列不能为空, 所以第一行插入不成功, 导致三行都插入不成功, 如图 4-5 所示; 重复上述命令, 插入后两行数据, 如图 4-6 所示。右击表 albums, 选择“查询”选项, 查看表中结果。

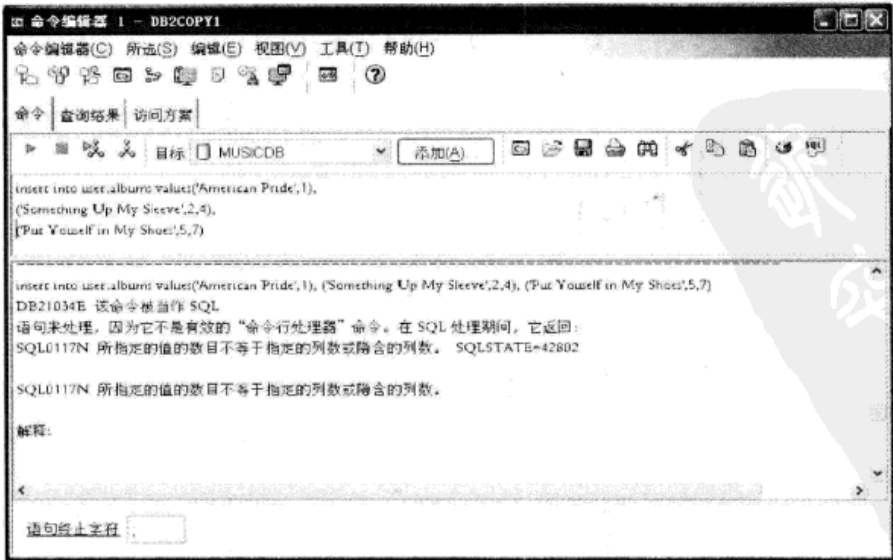


图 4-5 利用 insert 命令向 albums 表插入数据

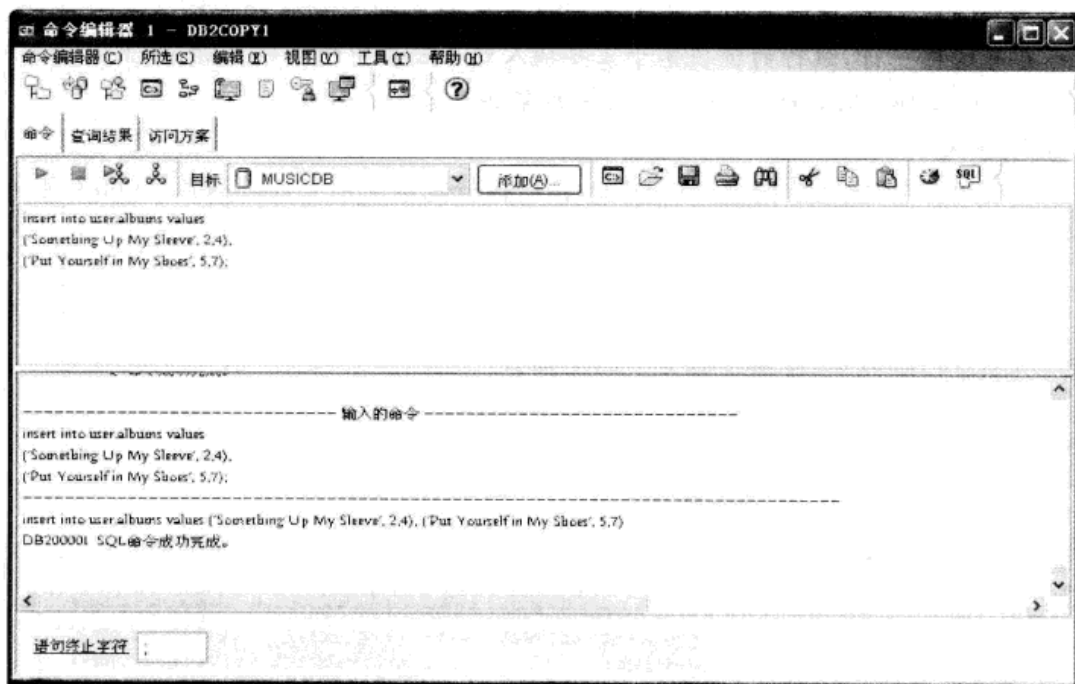


图 4-6 利用 insert 命令向 albums 表插入数据

## 2) 删除数据

(1) 删除表 artists 中 ARTNO 为 99 的记录信息。在“命令编辑器”中,输入 delete from user. artists where artno=99,执行并查看结果。

(2) 将表 albums 中 TITLE 为 Voice in the Wind 的记录删除。在“命令编辑器”中,输入 delete from user. albums where title='Voice in the Wind',执行并查看结果。

(3) 将 CLASSIFICATION 为 R 的演员的唱片集从 artists 表中删除。在“命令编辑器”中,输入 delete from user. artists where classification='R',执行并查看结果。

### 注意:

① 使用常量插入单个或多个元组,使用 INSERT 语句的语法格式如下。

```
INSERT INTO <表名> [( <属性列 1>, <属性列 2> ... )]
VALUES(<常量 1> [, <常量 2>] ... );
```

此语句的功能是将新元组插入到指定的表中。其中,新元组的属性列 1 的值为常量 1,属性列 2 的值为常量 2……没有在 INTO 子句中出现的属性列,新元组在这些列上将取空值。

② 删除语句的作用是将表中不再需要的元组从表中永久去除,不再保存在数据库中,其一般格式如下。

```
DELETE FROM <表名>
[WHERE <条件>];
```

DELETE 语句的功能是从指定表中删除符合条件的元组,如果没有 WHERE 子句,则删除所有元组。

③ 外键约束。DB2 确保只向定义了完整性约束的列插入有效数据。这意味着在父表中必须一直有这样一行,该行的键值等于正要插入到从属表中的行的外键值(上例插入不成功,albums 表插入数据时,artists 表中还无相应数据)。

## 2. 数据导入

使用 import 工具导入数据到 artists 表,稍后执行脚本文件导入数据到 albums 和 stock 表。首先用用户名 user 登录。

1) 使用 import 工具由 artists.exp 文件向 artists 表导入数据

(1) 在控制中心中,右击 artists 表,在弹出菜单中选择“导入”选项,如图 4-7 所示。在“导入表 artists”对话框中,按要求输入下列信息:



图 4-7 向 ARTISTS 表导入数据

- “导入文件”栏输入 X: \.....\cf23\artists.exp; 导入的源文件由 cf23 提供,文件名为 artists.exp(如图 4-8 所示)。
- “导入文件类型”设为集成交换格式(IXF)。
- “导入方式”设为 INSERT。
- “消息文件”栏输入 X: \cf23\art.msg,其中的 art.msg 是新生成的消息文件。

**注意:**

该导入过程的信息都返回到消息文件 art.msg 中,向 artists 表的导入数据的过程指定消息文件 art.msg,如图 4-8 所示。

- 选择“列”标签,选中“忽略代码页不匹配,且不进行转换”复选框,单击“确定”按钮,如图 4-9 所示。



图 4-8 ARTISTS 表的导入过程

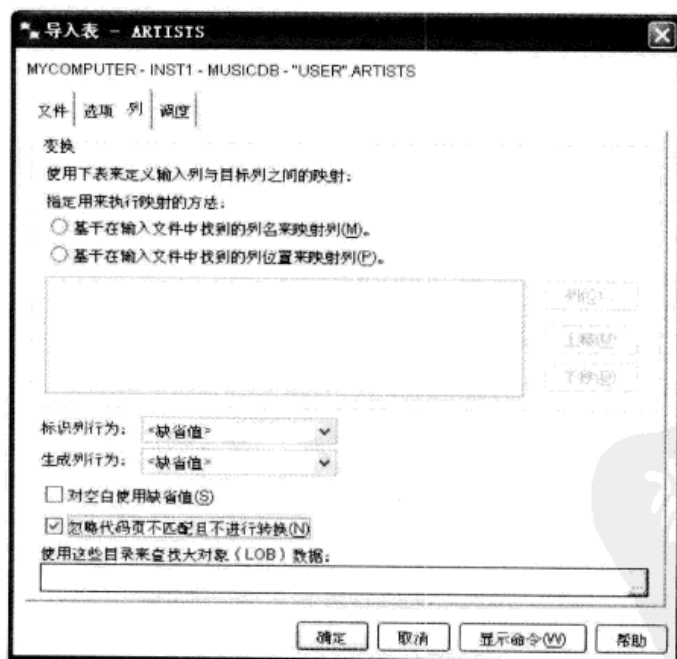


图 4-9 ARTISTS 表的导入过程“列”页的选择

注意：

① 把数据从一个数据库服务器移动到另一个数据库服务器是在开发过程和生产环境中经常要做的工作。DB2 提供了一些实用程序来帮助简化这些数据维护工作。要从数据库的表中提取数据，可以使用 Export 实用程序。Import 实用程序和 Load 实用程序能把输

② 数据移动实用程序所使用的文件格式有 5 种不同的文件格式。

- DEL。定界 ASCII。这种格式的文件是由行、列定界符分隔的 ASCII 字符流组成。逗号(,)是默认的列定界符、回车换行是默认的行定界符。使用双引号(" ")作为字符串的定界符。
- ASC。定长 ASCII。这种文件中每一列的长度都与对应表中列的长度定义相同。表中定义的变长字符列,在 ASC 文件中是采用以空格补齐到最长长度的方式存储的。
- IXF。集成交换格式。这种格式文件不能使用一般的文本编辑器查看和编辑。它使用一种能够可以用来在 DB2 数据库间移动数据的通用关系数据库交换格式——IXF 数据交换架构。因为 PC/IXF 是一种 IBM 专有的文件格式,所以 PC/IXF 只能用于在 DB2 数据库间移动数据。除了数据外,这种文件格式还包含了数据类型和表结构的定义,因此,这种文件可以用来先在数据库中创建表,然后再向表中导入数据。
- WSF。工作表格式。WSF 文件是一种 DB2 支持的 Lotus 1-2-3 和 Symphony 工作表文件格式。一般以 WKS、WK1、WRK、WR1 和 WJ2 作为扩展名。WSF 文件主要用于在 DB2 和工作表间移动数据。
- Cursor。Cursor 格式可以用来通过 Cursor 来向表中装入数据。在 load 命令中使用 Cursor 之前,Cursor 必须先在 SQL 查询语句中声明。只能在 load 实用程序中使用 Cursor 文件格式。

[illegible]

其中,

- INSERT, 不修改当前数据向数据表中增加新行。
- INSERT UPDATE, 向目标表增加新行, 或更新匹配主键已有的行。
- REPLACE, 从表中删除所有已有的数据并插入导入数据。
- REPLACE\_CREATE, 如果表不存在, 创建表和索引定义, 否则, 从表中删除所有已有的数据, 并插入导入数据, 不修改表或索引定义。
- CREATE, 创建表定义并插入新行。
- COMMITCOUNT n, 每导入 n 条记录就提交一次。
- IN tablespace-name, 指定创建表所在的表空间。
- INDEX IN tablespace-name, 指定表上索引所在的表空间, 只适用于 DMS 表空间。
- INTO table-name, 指定数据导入的表。
- MESSAGE, 指定写入警告或错误消息的文件, 假如不指定, 这些消息就写到标准输出。

可以指定输入文件名、文件的格式和目标文件名。例如:

```
import from employee.ixf of ixf
messages employee.out
insert into employee;
```

这个命令把 IXF 格式的文件 employee.ixf 作为输入, 把文件中的数据插入到 employee 表中。Import 实用程序支持 ASC、DEL、IXF 和 WSF formats 格式的文件。

(2) Windows 命令提示符窗口, 转到 cf23 目录下, 输入 more < art.msg, 查看消息文件中的警告或错误信息, 并注意有多少行数据成功插入表中, 如图 4-10 所示。

```
SQL3185W 当处理输入文件的第 "9" 行中的数据时发生先前的错误。
SQL3148W 未将输入文件中的行插入表中。返回了 SQLCODE "-803"。
SQL0803N INSERT 语句、UPDATE 语句或由 DELETE
语句导致的外键更新中的一个或多个值无效, 因为由 "1"
标识的主键、唯一约束或者唯一索引将表 "INST1.ARTISTS"
的索引键限制为不能具有重复值。 SQLSTATE=23505
SQL3185W 当处理输入文件的第 "10" 行中的数据时发生先前的错误。
SQL3110N 实用程序已完成处理。从输入文件读取了 "79" 行。
SQL3221W ...开始 COMMIT WORK。输入记录计数 = "79"。
SQL3222W ...COMMIT 任何数据库更改成功。
SQL3149N 从输入文件中处理了 "79" 行。将 "73" 行成功插入到表中。 "6" 行被拒绝。
```

图 4-10 消息文件 ART.MSG 中消息

## 2) 使用脚本文件向表导入数据

使用脚本文件 imp\_albu 和 imp\_sto 分别将数据导入到 albums 表和 stock 表中。

注意: imp\_albu 和 imp\_sto 由 cf23 提供。

(1) 在“命令编辑器”分别打开并运行脚本文件 imp\_albu(如图 4-11 所示)和 imp\_sto, 并执行, 如图 4-12 所示。该脚本文件将数据由源文件 albums.exp 和 stock.exp, 以插入的方式分别导入到表 albums 和表 stock 中。





图 4-11 打开脚本文件 imp\_albu

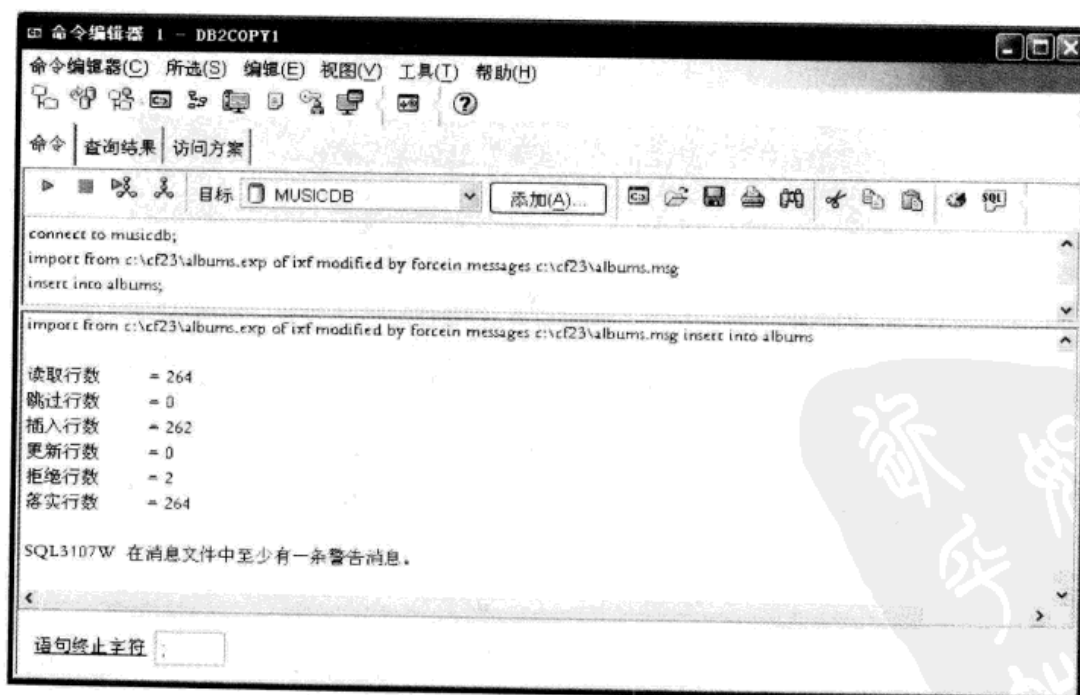


图 4-12 执行脚本文件 imp\_albu

(2) 脚本执行完毕后查看其相应的消息文件 albums.msg 和 sto.msg 是否有任何附加信息,并注意分别有多少行数据成功插入表中。

注意：

消息文件的查看请参照前面内容。向 albums 表导入数据后的结果,如图 4-13 所示;  
向 stock 表导入数据后的结果,如图 4-14 所示。

TITLE	ARTNO	ITEMNO
Voice in the wind	2	5
American Pride	1	2
Greatest Hits	1	1
Pass it All Down	1	3
Something Up ...	2	4
Hometown Girl	5	13
Come On Come ...	5	14
State of the Heart	5	15
I Dare You	99	302

图 4-13 脚本文件 imp\_albu 执行后表 album 中数据

ITEMNO	TYPE	PRICE	QTY
1	D	11.99	33
1	C	9.49	59
1	R	8.99	3
2	D	14.99	8
2	C	10.99	58
2	R	10.99	9
3	D	12.49	10
3	C	9.99	88
3	R	9.49	3
4	D	12.49	1
4	C	9.99	81
4	R	9.49	2
5	D	12.49	91

图 4-14 脚本文件 imp\_sto 执行后表 stock 中数据

### 3. 创建异常表——artists, albums, stock

所有的异常表将存储在 SMSEXP 表空间, 注意异常表的定义与原表非常相似, 前 N 列的列名和数据类型与原表精确匹配。第 N+1 列为可选列, 数据类型为 timestamp。第 N+2 列也为可选列, 且只有在第 N+1 列存在时才能被创建, 该列必须定义成 32KB 或更大的 CLOB 类型, 用来存放导致该行被拒的特定约束信息。表 artists 对应的异常表 artexp 结构, 如表 4-4 所示。

表 4-4 异常表 artexp 的结构

键	名 称	数 据 类 型	长 度	可 空
	ARTNO	SMALLINT	2	否
	NAME	VARCHAR	50	是
	CLASSIFICATION	CHARACTER	1	否
	BIO	CLOB	102400	是
	PICTURE	BLOB	512000	是
	TS	TIMESTAMP	10	是
	MSG	CLOB	32768	是

在“命令编辑器”中, 分别打开并运行脚本文件 crexptab, 如图 4-15 所示。该脚本分别为 artists、albums、stock 创建异常表 artexp、albexp、stoexp。



图 4-15 打开脚本文件 crexptab

#### 4. 备份数据库

后面的实验将使用 Load 命令以 Replace 方式装入数据(装入数据的目标表已经存在数据,要将这些数据用新数据替换掉),要使该操作可恢复,需要先将现有的数据进行备份。数据库备份与恢复的细节将会在以后的实验有所涉及。

在 db2 命令窗口中输入如下命令实现数据库备份操作如下。

- Cd\; 进入根目录。
- X; 进入 X 驱动器。
- Md X:\backup; 在 X 驱动器下建立新的文件夹 backup。
- Db2 force application all; 断开所有应用程序与数据库的连接。

Db2 backup db MUSICDB to X:\backup; 备份数据库 MUSICDB 到目录 X:\backup 下。

如果备份成功,返回备份映像的时间戳(格式为 yyyymmddhhmmss)。

#### 5. 使用 load 插入模式载入数据到 concerts 表

(1) 右击 concerts 表,在弹出菜单中选择“装入”项,如图 4-16 所示;在“类型”标签中,选择“将数据附加到表中”,在“文件”页,输入下列信息,如图 4-17 所示。

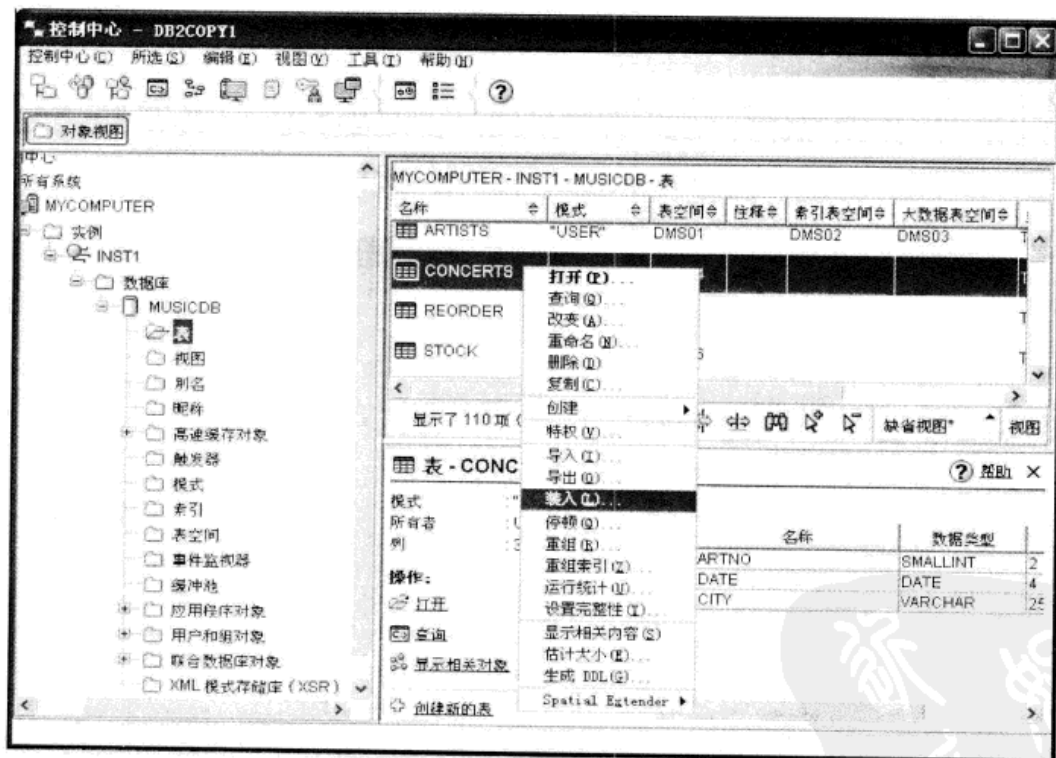


图 4-16 向 concerts 表装入数据的选择

- 输入文件格式为 ixf 格式;
  - 载入文件位置为服务器;
  - 输入文件的完整路径和文件名为 X:\cf23\concerts.exp;
  - 要存储进度消息的完整路径和文件名为 X:\cf23\concerts.msg;
- 在“列”页面中,选“忽略代码页不匹配且不进行转换”,如图 4-18 所示。

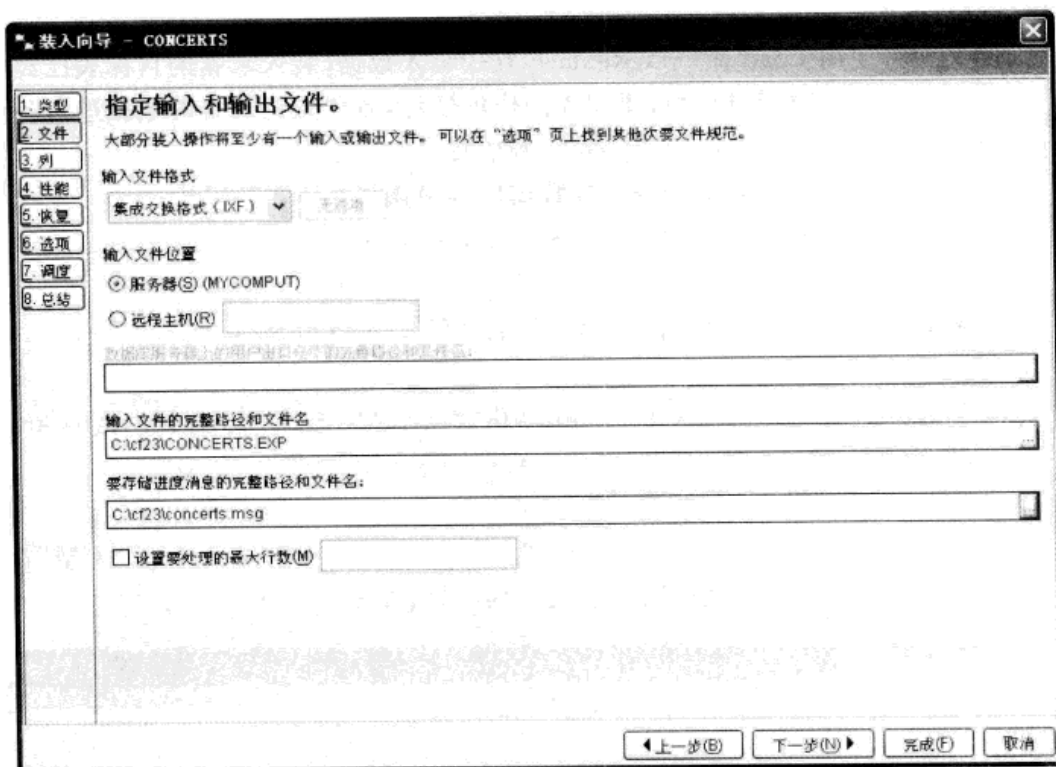


图 4-17 向 concerts 表装入数据“文件”页的选择

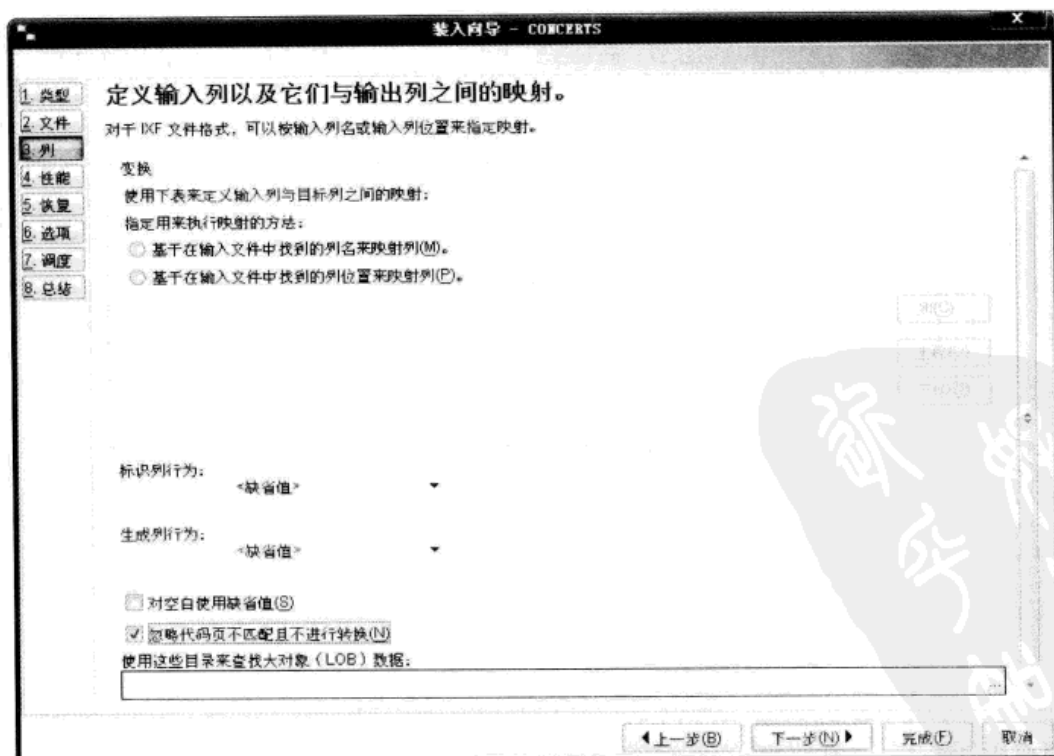
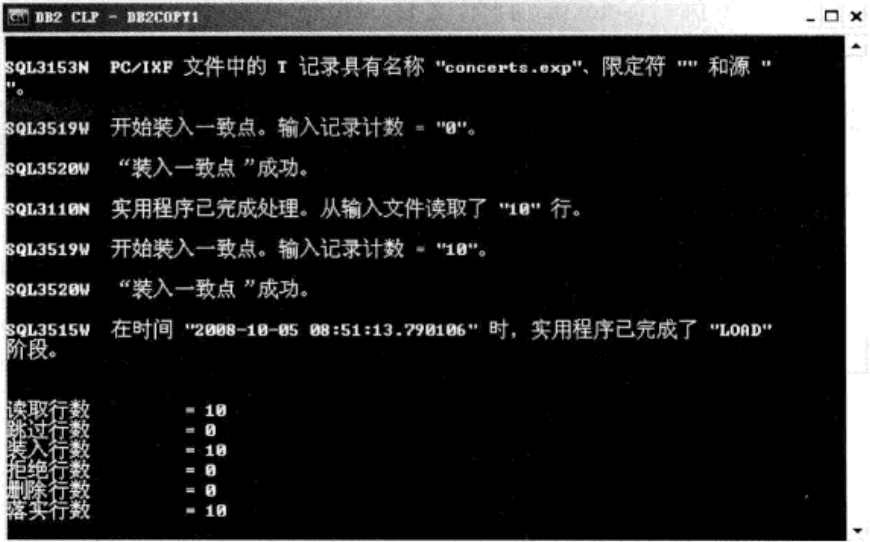


图 4-18 向 concerts 表装入数据“列”页的选择

在“调度”页面中,选择“立即执行而不保存历史任务”复选框。

在“总结”页面中,查看相应的 SQL 语句,并单击“完成”按钮。

(2) 查看消息文件 X:\cf23\concerts.msg,注意有多少行成功载入并插入到表中。在命令窗口中,在 X:\cf23 路径中,输入命令: more < concerts.msg,如图 4-19 所示。



```
SQL3153N PC/IXF 文件中的 I 记录具有名称 "concerts.exp", 限定符 "" 和源 ""。
SQL3519W 开始装入一致点。输入记录计数 = "0"。
SQL3520W "装入一致点" 成功。
SQL3110N 实用程序已完成处理。从输入文件读取了 "10" 行。
SQL3519W 开始装入一致点。输入记录计数 = "10"。
SQL3520W "装入一致点" 成功。
SQL3515W 在时间 "2008-10-05 08:51:13.790106" 时, 实用程序已完成了 "LOAD" 阶段。

读取行数      = 10
跳过行数      = 0
装入行数      = 10
拒绝行数      = 0
删除行数      = 0
落实行数      = 10
```

图 4-19 消息文件 concerts.msg 中的消息

### 6. 使用 load 工具的替换模式载入数据到表 artists 中

在“命令编辑器”中,打开并运行脚本文件 load\_art,将数据装入到 artists 表中。脚本文件 load\_art 在命令编辑器中的执行过程如图 4-20 所示。

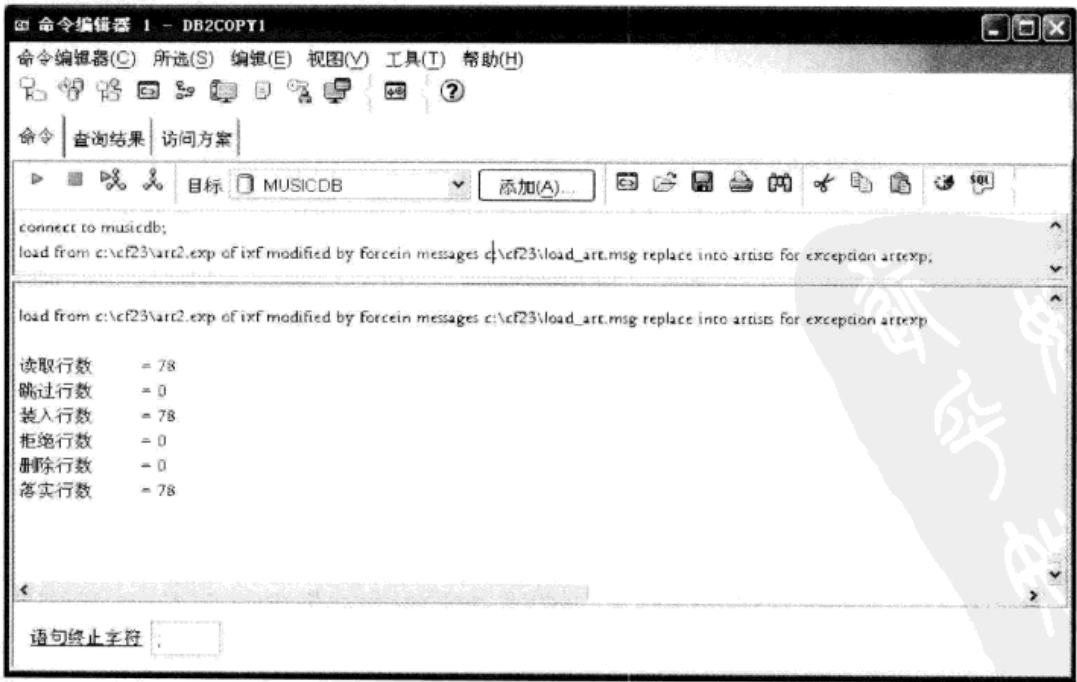


图 4-20 脚本文件 load\_art 的执行

- 
- 命令编辑器 3 - DB2COPY1
- 命令编辑器(C) 所选(S) 编辑(E) 视图(V) 工具(T) 帮助(H)
- 命令 查询结果 访问方案
- 目标 MUSICDB 添加(A)...
- ```

connect to musicdb;
select artno, ts, substr(msg,1,40) from artexp;
select artno, itemno, ts, substr(msg,1,40) from albexp;
select itemno, ts, substr(msg,1,40) from stoexp;

select artno, ts, substr(msg,1,40) from artexp

```
- | ARTNO | TS |
|-------|----|
| 3     |    |
- 0 条记录已选择。
- ```

select artno, itemno, ts, substr(msg,1,40) from albexp

```
- 语句终止字符

- 查看消息文件 X:\cf23\load\_art.msg 内容:

注意:

- ```
>>>LOAD-----FROM-----filename-----OF-----filetype----->
      '-CLIENT-'      '+pipeName---+'
                        '+device-----+'
                        '+cursorName-'
```

```

>----->
|          .-.-.-.-. |
|          V          |
|'-LOBS FROM----lob-path-+-'
>----->
|          .-.-.-.-. |
|          V          |
|'-MODIFIED BY----filetype-mod-+-'
>----->
|'-SAVECOUNT--n-' |'-ROWCOUNT--n-' |'-WARNINGCOUNT--n-'
>----->
|'-MESSAGES--message-file-'
>----->
|          .-.-.-.-. | +--INSERT-----+
|          V          | +--REPLACE-----+
|          |          | +--RESTART-----+
|          |          | +--TERMINATE-----+
|'-INTO--table-name-+-'
|          .-.-.-.-. |
|          V          |
|'- (----insert-column-+-)-'
>----->
|'-FOR EXCEPTION--table-name-'
>----->
|'-STATISTICS--+-USE PROFILE-+-'
|          '-NO-----'
>----->
|          .-NO-----+
| +--COPY--+-YES--+-USE TSM-+-+
|'-NONRECOVERABLE-----'
>----->
|'-WITHOUT PROMPTING-' |'-DATA BUFFER--buffer-size-'
>----->
|'-SORT BUFFER--buffer-size-' |'-CPU_PARALLELISM--n-'
>----->
|'-DISK_PARALLELISM--n-'
>----->
|'-INDEXING MODE--+-AUTOSELECT--+-'
|          +-REBUILD-----+
|          +-INCREMENTAL-+
|          '-DEFERRED-----'
|          .-ALLOW NO ACCESS-----
>----->
|'-ALLOW READ ACCESS-+-'
|          '-USE--tablespace-name-'
>----->
|'-CHECK PENDING CASCADE--+-IMMEDIATE-+-'
|          '-DEFERRED--'
>----->
|'-LOCK WITH FORCE-'

```



一个把 IXF 格式的数据文件 stock.ixf 装入到表 stock 的实例如下。

```
load from stock.ixf of ixf modified by dumpfile = stockdump.dmp
messages stock.out replace into stock for exception stockexp;
```

④ 与导入工具不同的是,在开始装入前,目标表必须存在于数据库中。对于数据库管理员来说,导入和装入工具最重要的不同在于性能。导入工具每次插入一行数据,插入的每行都要被检查是否满足约束条件,对数据的修改也记录在日志文件中。装入工具向表插入数据的速度要比导入工具快得多,它不是每次插入一行,而是使用从输入文件读出的行构建页,这些页直接写入数据库。已有的索引(包括主键或唯一索引)可在数据页插入后重建(可全部重建或基于新加入的数据部分重建),最后,所有不遵从唯一性约束或主键约束的重复行都从表中删除。所以在装入 artists 表后,该表处于挂起状态,这时该表是不可用的。这时可以利用 Set Integrity 命令来管理检查暂挂状态。检查后把不满足完整性约束的数据删除掉,该表由挂起状态恢复。由于各表(albums、stock 与 artists)之间存在着参照约束,所以在往一个表装入数据时,会导致其他表处于挂起状态。同样利用 Set Integrity 命令来管理检查暂挂状态,检查参照约束,使各表由挂起状态恢复。

⑤ 在装入期间,单个记录不记录在日志文件中。因为装入工具修改不记录日志,所以不能使用日志前滚。但是装入后可以使用源数据的备份来恢复数据。因此在装入之前,要将数据库做好备份。

## 7. 使用 SET INTEGRITY 命令管理检查暂挂状态

### 1) 用 Set Integrity 管理检查暂挂状态

(1) 在“命令编辑器”中,打开并执行脚本文件 selstab,该脚本文件将对表 artists、stock、albums 执行查询操作,查看执行结果。脚本文件 selstab 在命令编辑器中的执行过程如图 4-22 所示。

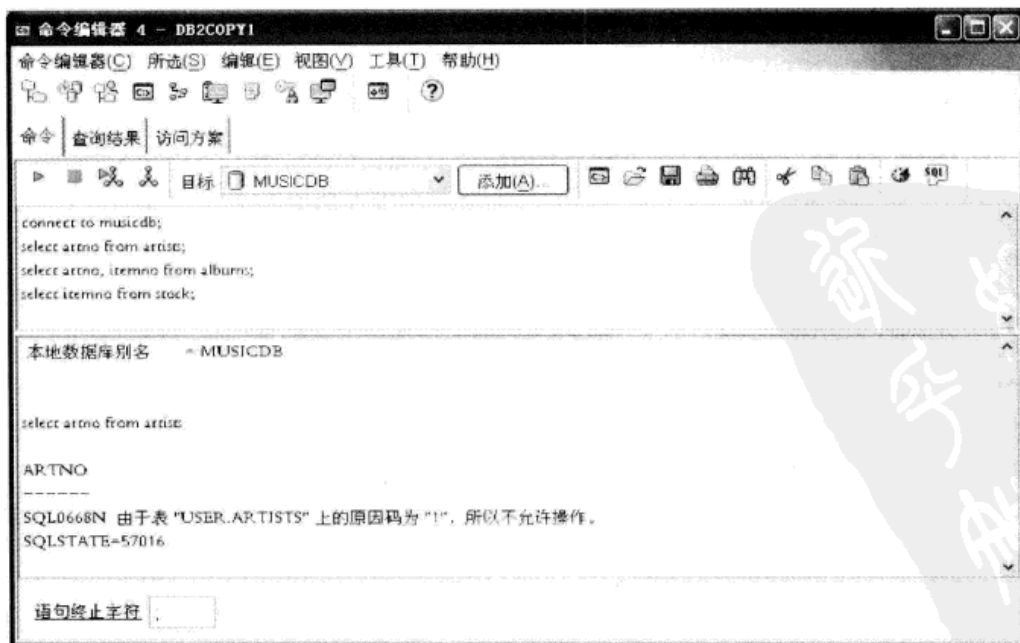


图 4-22 脚本文件 selstab 的执行结果

注意:

由于向 artists 表装入数据,但还未对待装入的数据进行完整性约束检查,所以 artists 表处于检查挂起状态,不能对 artists 表进行查询操作。

(2) 上述脚本文件执行后返回 SQL0668 原因代码 1,命令中心中输入:“? SQL0668”,该命令的执行结果如图 4-23 所示。



图 4-23 ? SQL0668 的执行结果

由于向 artists 表中以替换方式,装入数据,所以该表处于检查暂挂状态,对该表的数据操作都不能执行。可以执行 SET INTEGRITY 语句以消除表的检查暂挂状态。

(3) 在“命令编辑器”中,打开并执行脚本文件 listtbst 查看这些表上检查约束的状态。脚本文件 listtbst 在命令编辑器中的执行结果如图 4-24 所示。Artists 表处于检查暂挂状态。

(4) SET INTEGRITY 语句可被用来检查数据是否违反参照完整性约束,处于检查暂挂状态的表(artists 表)应被指定一个与它对应的异常表。在“命令编辑器”中,打开并执行脚本文件 setcsts\_art,如图 4-25 所示,该脚本文件用来检查 artists 表中完整性约束。

(5) 警告 sql3601。在“命令编辑器”中输入“? Sql3601”,由帮助信息可知 set integrity 语句导致一个或多个表处于检查暂挂状态。

(6) 在“命令编辑器”中,打开并执行脚本文件 listtbst,如图 4-26 所示,执行后可见 albums 和 stock 表处于检查暂挂状态,即 albums 和 stock 表不可用。由于 albums 和 stock 表与 artists 表存在参照关系,所以 artists 表恢复正常状态,而其他两个表处于挂起状态。

(7) 在“命令编辑器”中,打开并执行脚本文件 setcsts\_2,如图 4-27 所示,为 albums 和 stock 表设置完整性,检查它们与 artists 表的参照完整性,返回 SQL3602W 检查数据处理时发现约束违例并将它们移至异常表中。

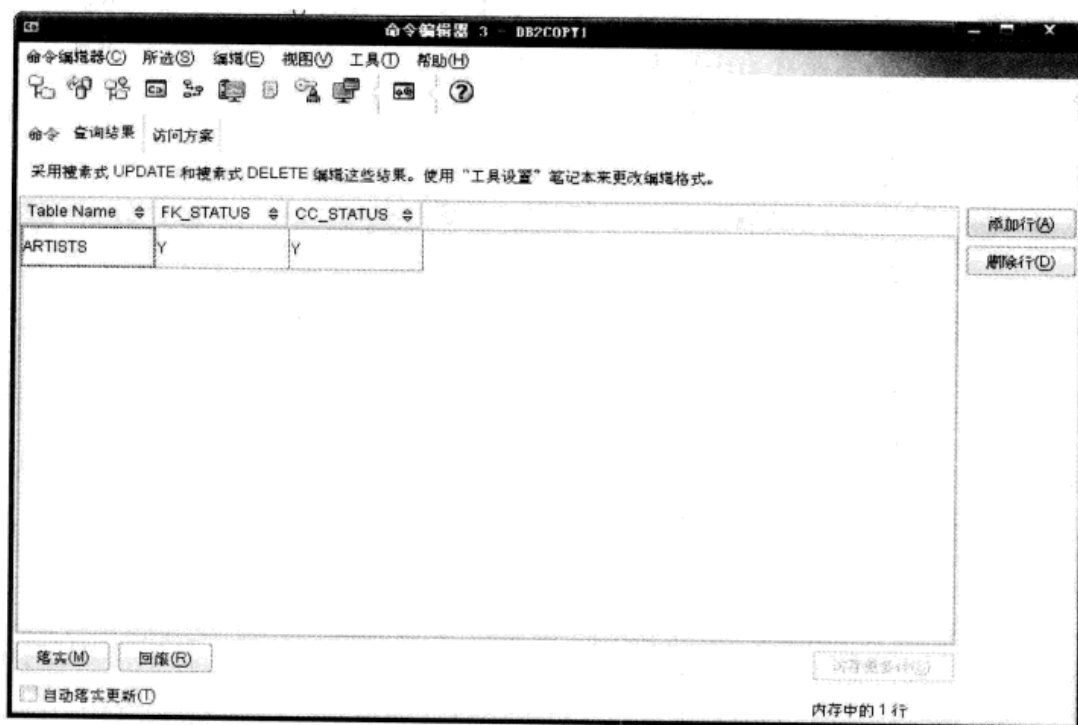


图 4-24 脚本文件 listtbls 的执行结果

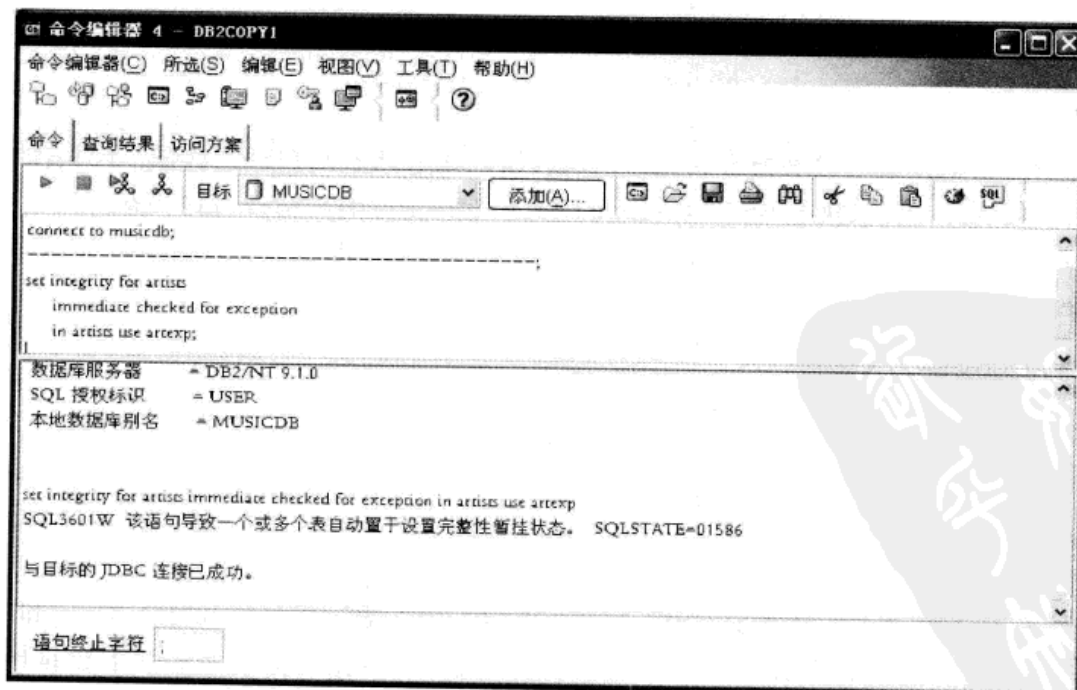


图 4-25 脚本文件 setests\_art 的执行结果

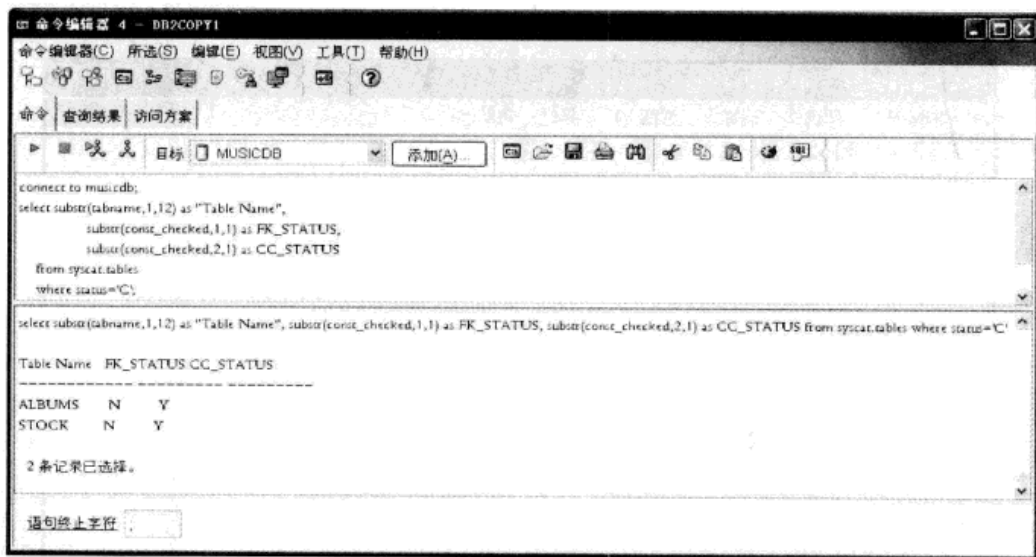


图 4-26 脚本文件 listtblst 的执行结果

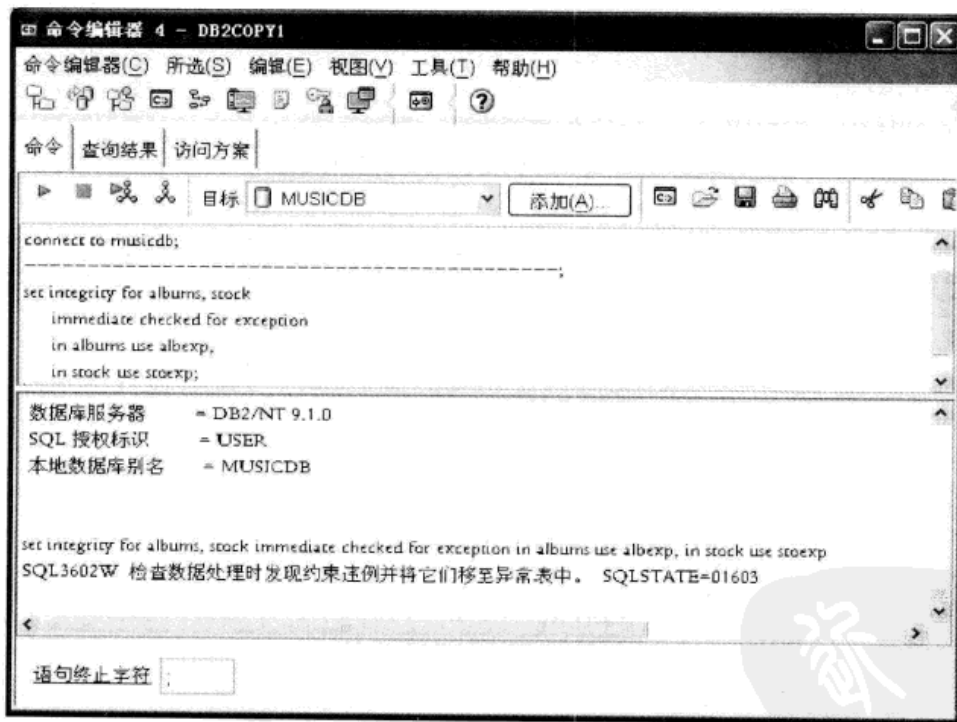


图 4-27 脚本文件 setests\_2 的执行结果

(8) 对 artists、stock、albums 表执行查询操作以确认这些表已消除检查暂挂状态。在“命令编辑器”中，打开并执行脚本文件 listtblst(如图 4-28 所示)和 selstab, 执行结果表明检查暂挂状态都已消除。

2) 将异常表中的数据放回原来表中

(1) 为解除检查暂挂状态, 那些违反约束的行被移动到异常表中。在“命令编辑器”中, 打开并执行脚本文件 selexp 查看异常表 albexp 和 stoexp 中的数据, 执行结果如图 4-29 所示。



图 4-28 脚本文件 listtblst 的执行结果

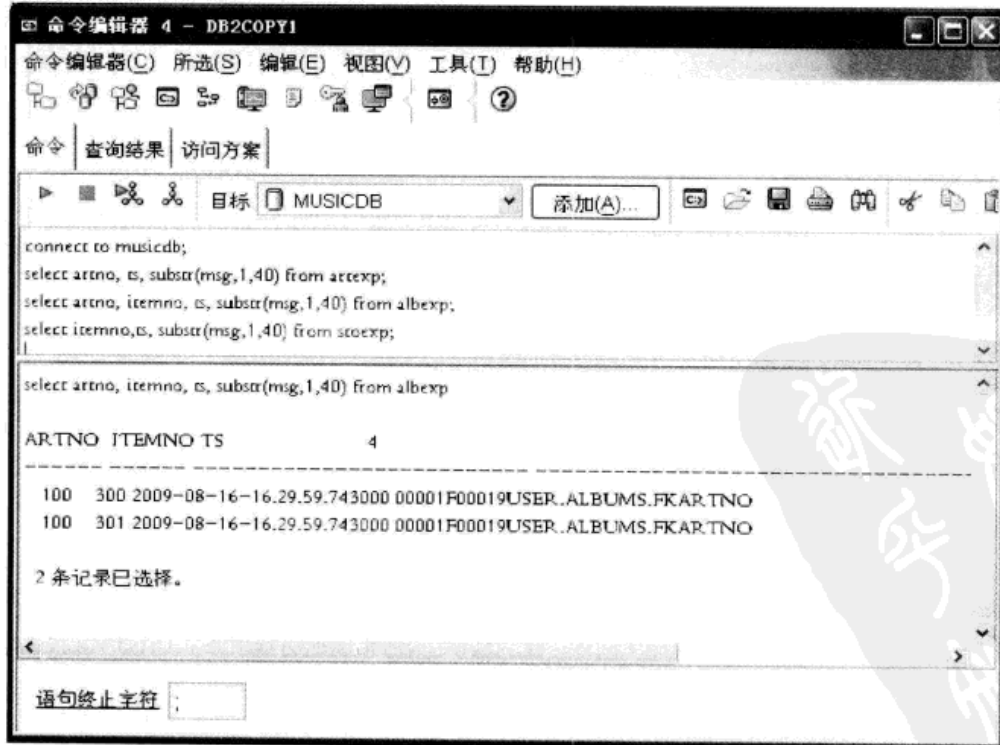


图 4-29 脚本文件 selexp 的执行结果

(2) 从 stock 移动到 Stoexp 的数据以及从 albums 移动到 albexp 的数据, 这些数据的外键值都无法满足参照完整性约束。执行下列 SQL 语句在 artists 表中插入一行数据:

```
insert into user.artists (artno, name, classification) values (100, 'Patti & Cart Wheels', 'S')
```

(3) 现在要将 stoexp 和 albexp 表中的行分别插入到表 stock 和 albums 中, 首先打开两异常表, 检查两异常表的 MSG 列, 注意并不需要将 N+1 和 N+2 列数据插入到 stock 和 albums 表。

(4) 在“命令编辑器”中, 打开并执行脚本文件 insexp, 将 stoexp 和 albexp 中的数据插入到表 stock 和 albums, insexp 脚本也会对 stock 和 albums 执行查询操作, 以检查异常表中数据是否成功插入到原表。insexp 脚本文件在命令编辑器中的执行过程如图 4-30 所示。

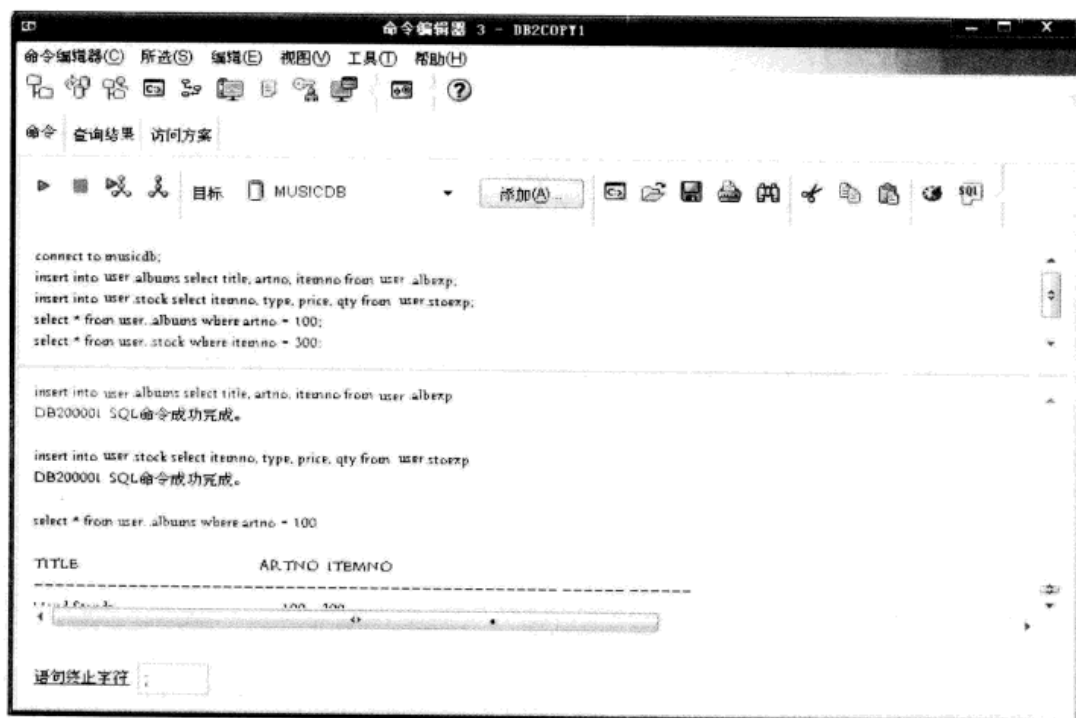


图 4-30 脚本文件 insexp 的执行结果

## 8. 理解检查约束的强制执行

(1) 在实验中, 为 stock 表的 TYPE 列添加的检查约束 cctype, 要查看该约束的内容, 可以在“控制中心”中右击表 stock 并在弹出菜单中选择“改变”选项, 再在“改变表-stock”选项卡中选择“检查约束”标签, 选择 cctype 约束然后单击“更改”按钮, 查看该约束的定义如图 4-31 所示。

(2) 在“命令编辑器”中输入并执行如下 SQL 语句:

```
insert into user.stock values (302, 'V', 100.00, 20);
```

输出结果返回 SQL0545, 如图 4-32 所示, 该错误代码表明因为插入的数据不满足检查约束 user.stock.cctype, 所以要请求的操作不被允许。



图 4-31 检查约束 cctype 的定义



图 4-32 向 stock 表插入数据的结果

(3) 在命令中心中输入并执行如下 SQL 语句:

```
insert into user.stock values (302, 'C ',100.00, 20);
```

因为'C'满足检查约束 cctype,所以该 SQL 语句成功执行。

## 9. 理解触发器的强制执行

(1) 回顾上节实验中在表 stock 的 QTY 列上创建的触发器 reorder。选择“命令编辑器”输入并执行下列 SQL 语句,执行结果如图 4-33 所示。

```
select substr(text,1,200) from syscat.triggers where tabname = 'STOCK';
```

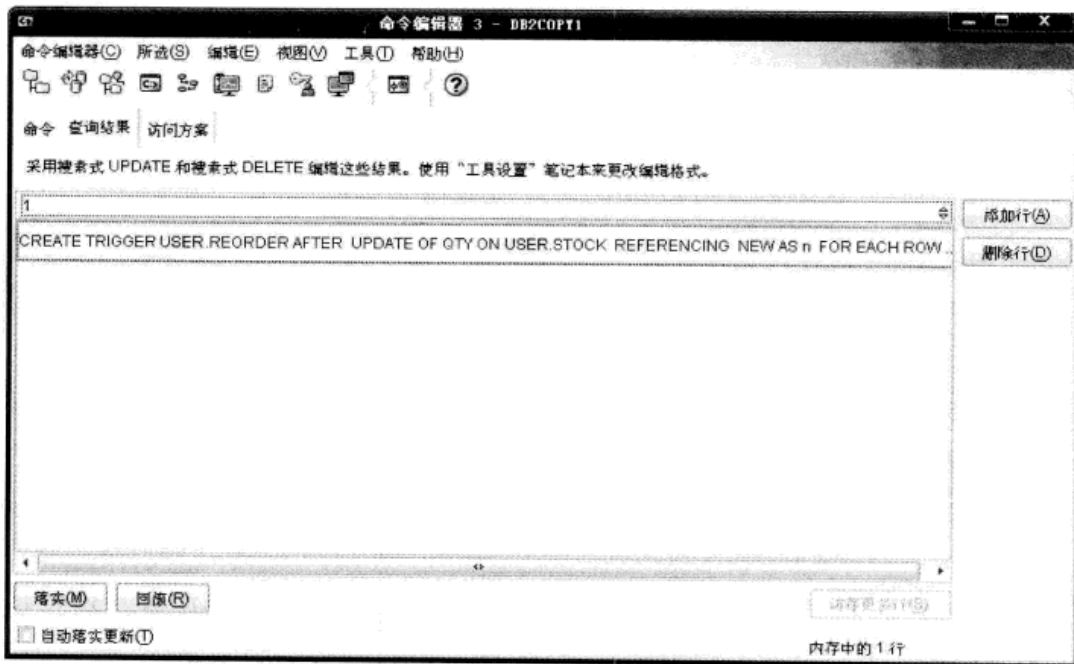


图 4-33 查看建立在 stock 表上的触发器

**注意:**

substr() 函数是取子串,对视图 syscat.triggers 的 text 列取子串,从第一个字符起,连续取 200 个字符。

(2) 在“命令编辑器”中输入并执行下列 SQL 语句来修改 stock 表中 itemno 等于 302 的元组的 qty 字段值:

```
update stock set qty = 3 where itemno = 302;
```

**注意:** 执行该语句后没有消息显示触发器是否被触发。

(3) 查看 reorder 表,检查在更新 stock 表 qty 列后触发器 reorder 是否被触发。在“命令编辑器”中输入:

```
terminate;  
connect to musicdb;  
select * from user.reorder;
```

执行结果如图 4-34 所示。

(4) 实验 3 为表 reorder 建立了别名 emptystock,在命令中心中输入并执行下列 SQL 语句:

```
select * from emptystock
```



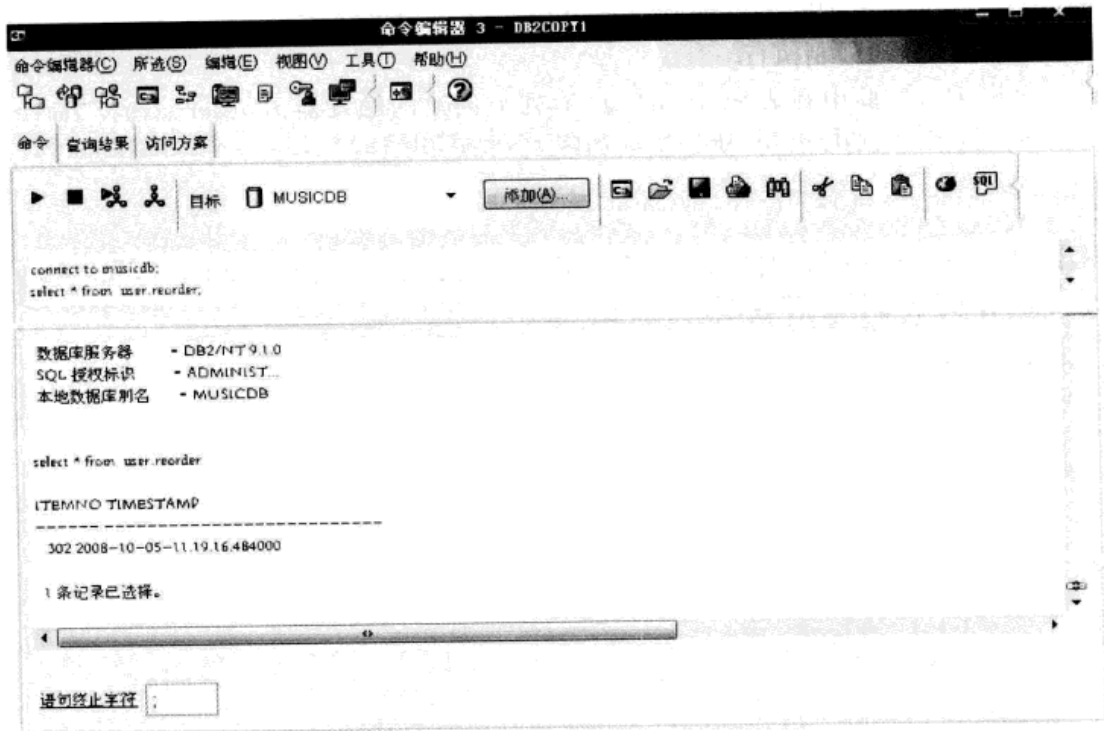


图 4-34 reorder 表的查询结果

输出结果如图 4-35 所示,表明对别名 emptystock 执行的查询操作会直接转化成对 reorder 表的操作。

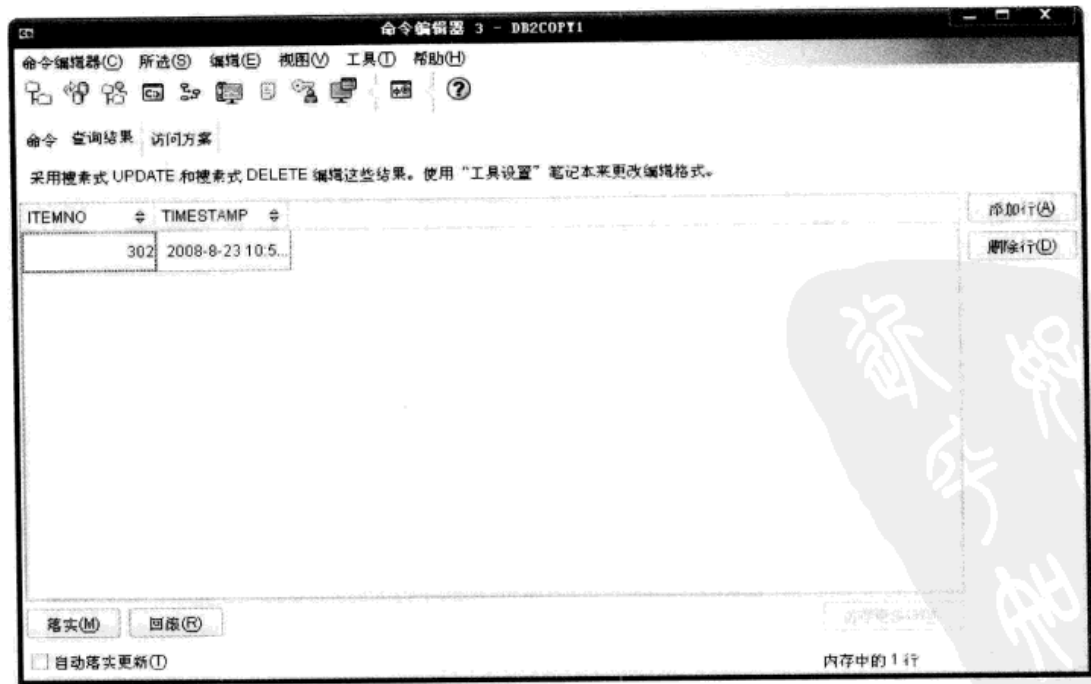


图 4-35 emptystock 和 reorder 的查询结果

## 实验 5

## 数据操纵——查询

### 实验任务

- 掌握简单查询和连接查询;
- 掌握子查询;
- 掌握列函数和标量函数应用;
- 掌握分组和排序。

### 实验内容

查询语句的功能是从数据库中检索满足条件的数据。SQL 语言提供了 SELECT 语句进行数据库的查询,该语句具有灵活的使用方法和丰富的功能。其一般格式为:

|                                                |         |
|------------------------------------------------|---------|
| SELECT [ALL DISTINCT] <目标列表达式> [,<目标列表达式>] ... | 需要哪些列   |
| FROM <表或视图名> [,<表或视图名>] ...                    | 来自于哪些表  |
| [WHERE <条件表达式>]                                | 根据什么条件  |
| [GROUP BY <列名 1> [HAVING <条件表达式>]]             | 分组(-筛选) |
| [ORDER BY <列名 2> -[ASC   DESC]];               | 排序      |

选择菜单“命令行工具”→“命令编辑器”选项,或在其他窗口中单击相应的图标来启动命令编辑器,如图 5-1 所示。在“交互式”页面中,输入 Connect to MUSICDB,或单击图中的“添加”按钮选择“指定目标”项,选中 MUSICDB 数据库,单击“确定”按钮。然后针对下列内容,写出相应的 SQL 查询语句。

#### 1. 常规数据查询

1) 掌握 SELECT 子句以及 WHERE 子句的使用方法

(1) 查看 artists 表中的所有信息。

```
SELECT *  
FROM ARTISTS
```

根据查询语句,其查询结果如图 5-2 所示。

(2) 从 artists 表中,查询 CLASSIFICATION 为'C'或'B'的演员的信息。

```
SELECT *  
FROM ARTISTS  
WHERE CLASSIFICATION = 'C' OR CLASSIFICATION = 'B'
```

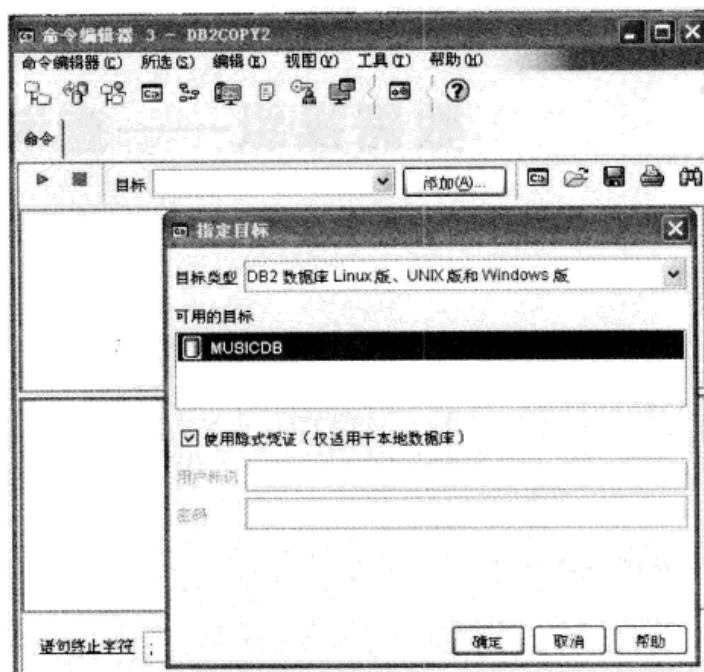


图 5-1 命令编辑器

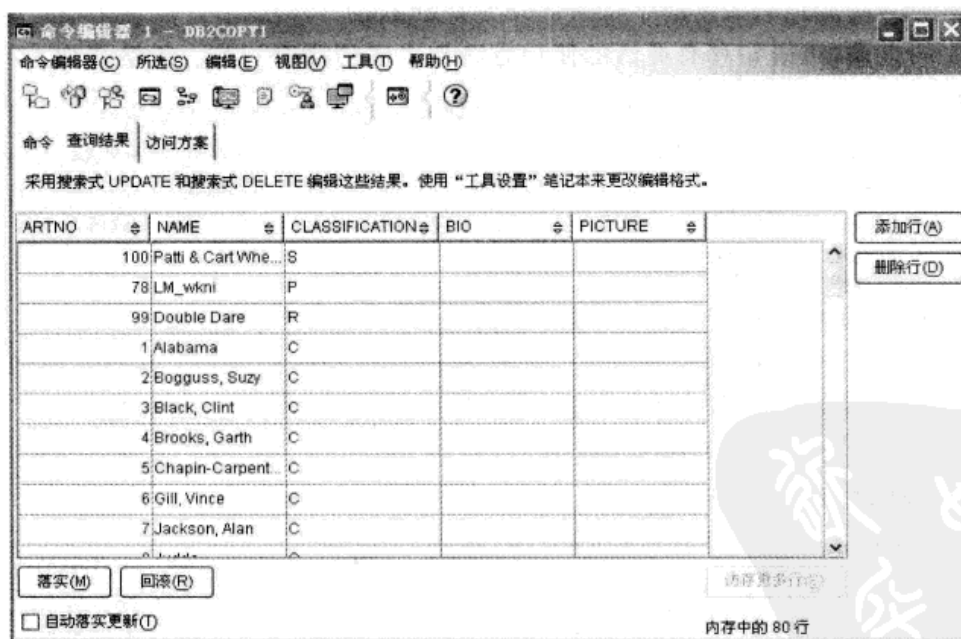


图 5-2 查询结果

根据查询语句,其查询结果如图 5-3 所示。

## 2) 学会应用 ORDER BY 子句

查询 CLASSIFICATION 为 'C' 的演员名字,并将结果按 ARTNO 升序排列。

```
SELECT NAME
FROM ARTISTS
```

WHERE CLASSIFICATION = 'C'  
ORDER BY ARTNO

根据查询语句,其查询结果如图 5-4 所示。

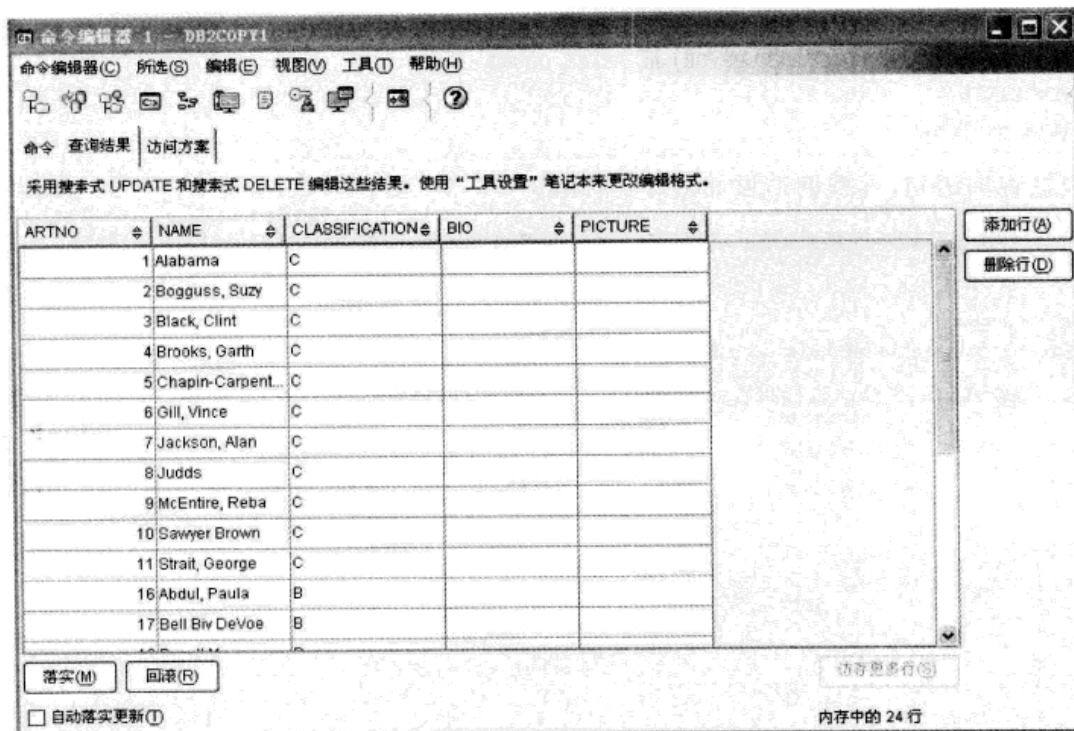


图 5-3 查询结果

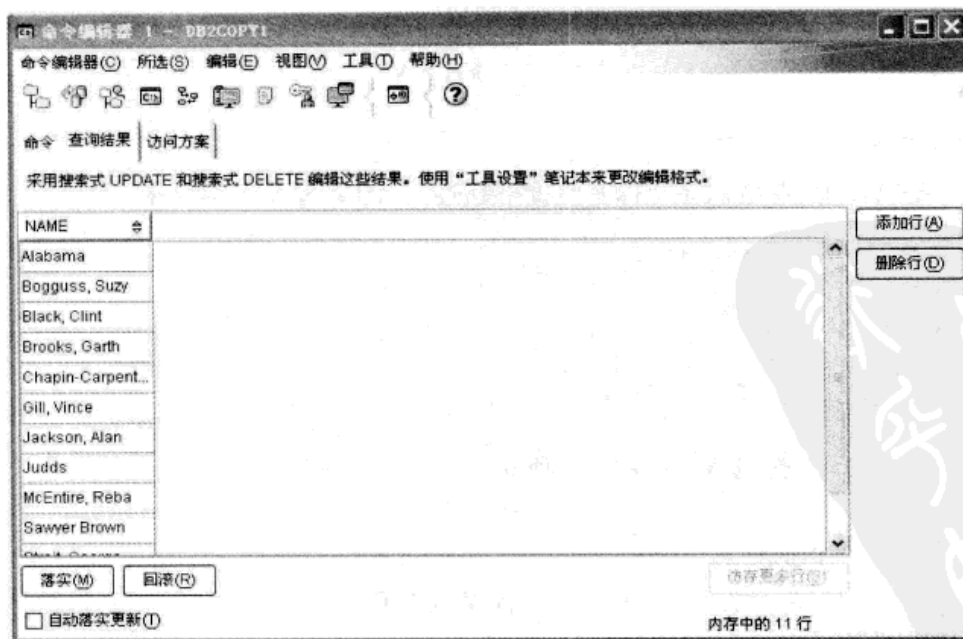


图 5-4 查询结果

### 3) 学习聚合函数的使用

按唱片类型(TYPE)分别统计唱片的数量、平均价格、最高价格及最低价格,以及每种类型唱片的总行数和每种类型唱片的价格种类。

```
SELECT TYPE, SUM(QTY) AS SUM, AVG (PRICE) AS AVG, MAX(PRICE) AS MAX, MIN (PRICE) AS MIN, COUNT ( * ) AS
TYPE_COUNT, COUNT (DISTINCT PRICE) AS PRICE_COUNT
FROM STOCK
GROUP BY TYPE
```

根据查询语句,其查询结果如图 5-5 所示。

| TYPE | SUM   | AVG           | MAX   | MIN   | TYPE_COUNT | PRICE_COUNT |
|------|-------|---------------|-------|-------|------------|-------------|
| C    | 13640 | 10.0170270270 | 10.99 | 7.99  | 259        | 6           |
| D    | 12600 | 13.1309266409 | 15.99 | 10.99 | 259        | 8           |
| R    | 2817  | 9.74868725868 | 10.99 | 7.99  | 259        | 6           |

图 5-5 查询结果

### 4) 分组统计

列出每类唱片中,单价高于 13、平均价格超过 15 的类型及平均价格。

```
SELECT TYPE, AVG (PRICE) AS PRICE
FROM STOCK
WHERE PRICE > 13
GROUP BY TYPE
HAVING AVG (PRICE) > 15
```

根据查询语句,其查询结果如图 5-6 所示。

## 2. 多表查询

### 1) 查询音乐会的信息

查询信息包括音乐会的时间、地点以及演员的名称。

```
SELECT DATE, CITY, NAME
FROM CONCERTS, ARTISTS
WHERE CONCERTS.ARTNO = ARTISTS.ARTNO
```

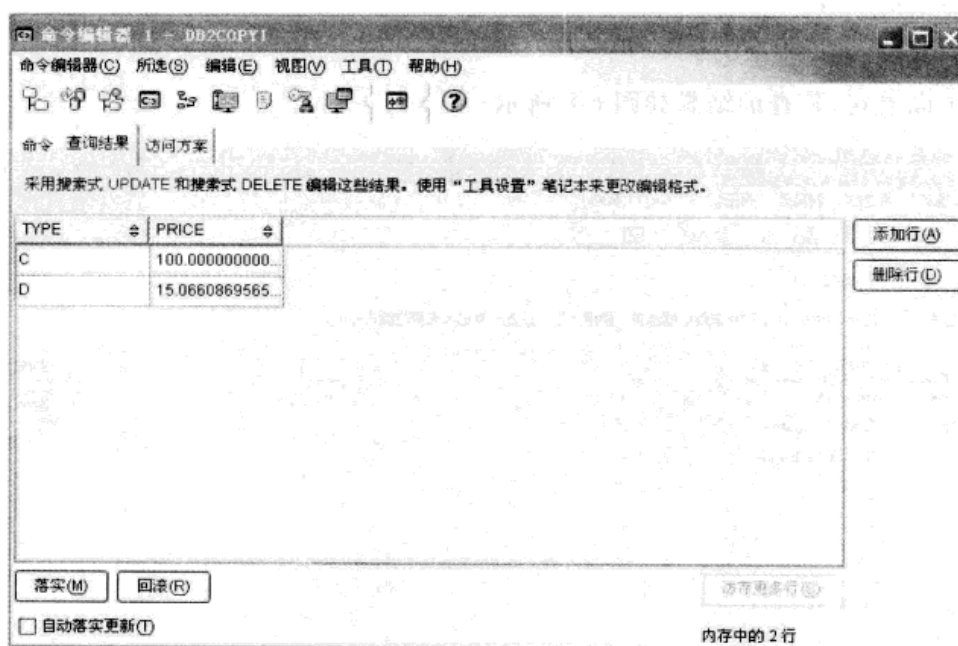


图 5-6 查询结果

根据查询语句,其查询结果如图 5-7 所示。

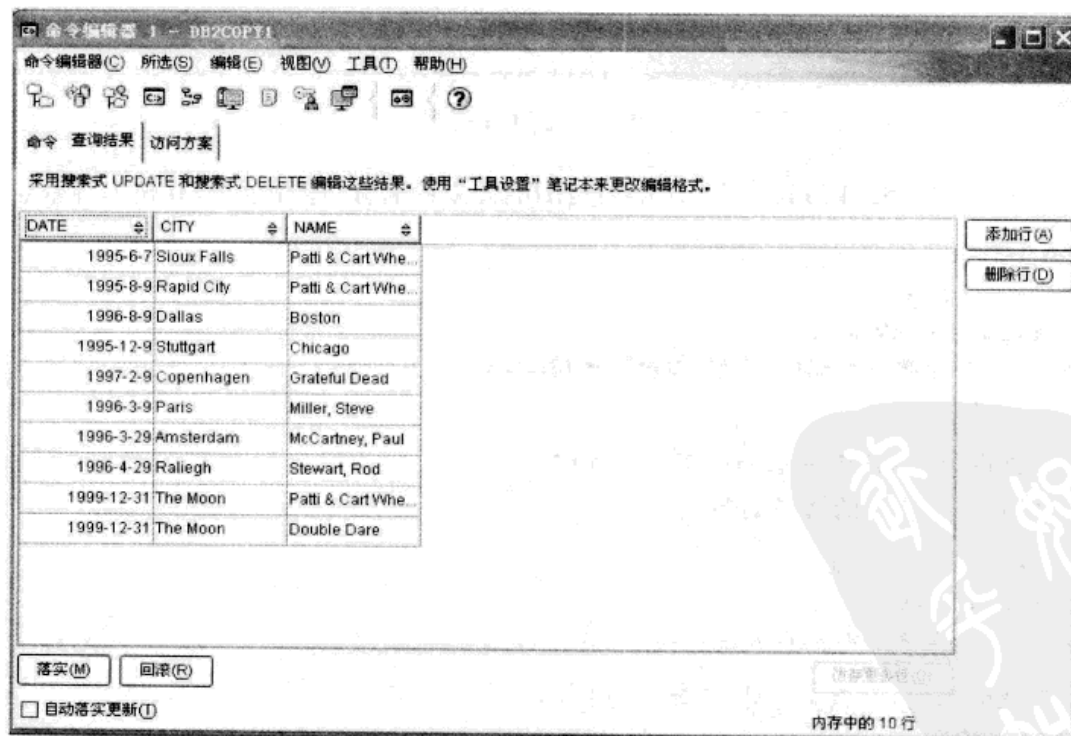


图 5-7 查询结果

2) 查询在 Paris 举办的音乐会的演员名字及其所出版的唱片集名称

SELECT NAME, TITLE

```
FROM CONCERTS, ARTISTS, ALBUMS
WHERE CONCERTS. ARTNO = ARTISTS. ARTNO AND CONCERTS. ARTNO = ALBUMS. ARTNO AND CITY = 'Paris'
```

根据查询语句,其查询结果如图 5-8 所示。

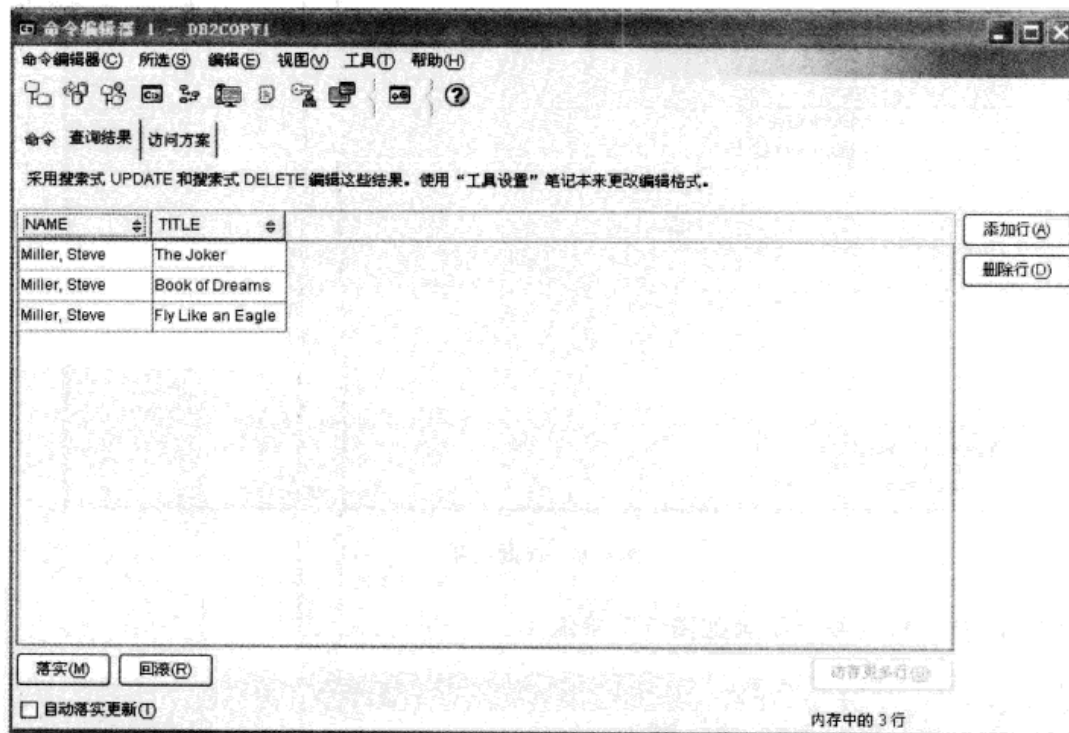


图 5-8 查询结果

3) 查询 CLASSIFICATION 为 'B' 的演员的唱片集的名称、库存数量,并按库存数量升序排列结果

```
SELECT TITLE, QTY
FROM ALBUMS, STOCK, ARTISTS
WHERE STOCK. ITEMNO = ALBUMS. ITEMNO AND ARTISTS. ARTNO = ALBUMS. ARTNO AND CLASSIFICATION = 'B'
ORDER BY QTY
```

根据查询语句,其查询结果如图 5-9 所示。

4) 查询有唱片集的演员的数量

```
SELECT COUNT (NAME) AS COUNT
FROM ARTISTS, ALBUMS
WHERE ARTISTS. ARTNO = ALBUMS. ARTNO
```

根据查询语句,其查询结果如图 5-10 所示。

### 3. 其他类型查询

1) 查询姓名 (NAME) 以 'LM\_' 开头的演员的所有信息

```
SELECT *
FROM ARTISTS
WHERE NAME LIKE 'LM $ _ % ' ESCAPE ' $ '
```



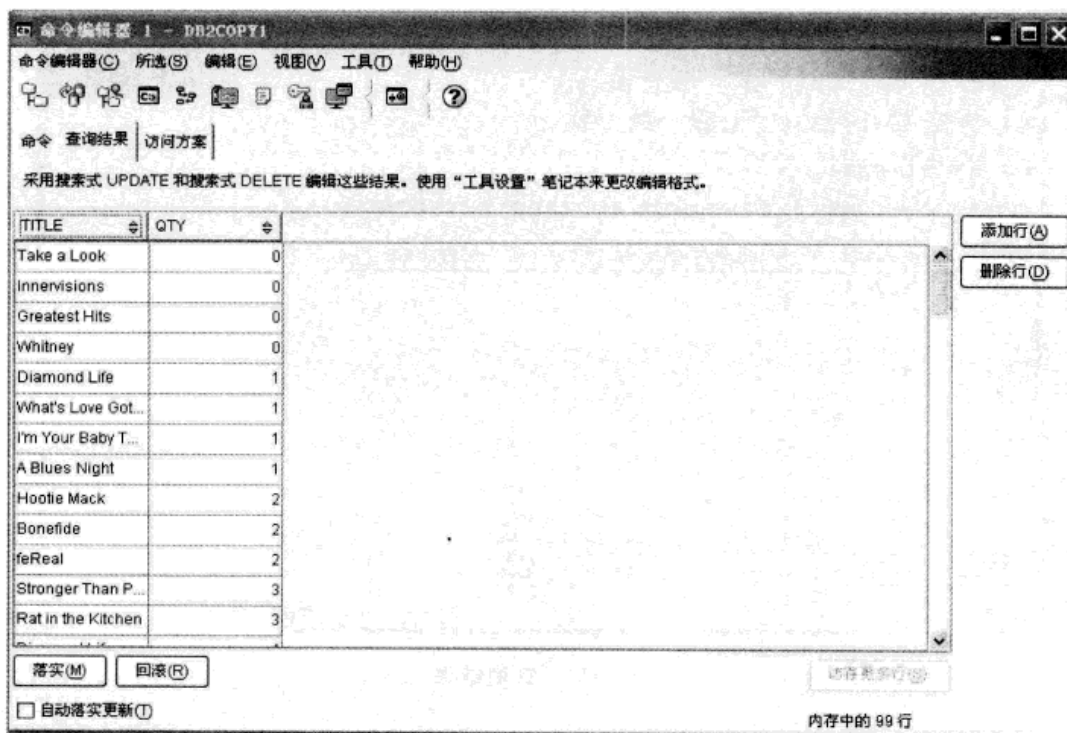


图 5-9 查询结果

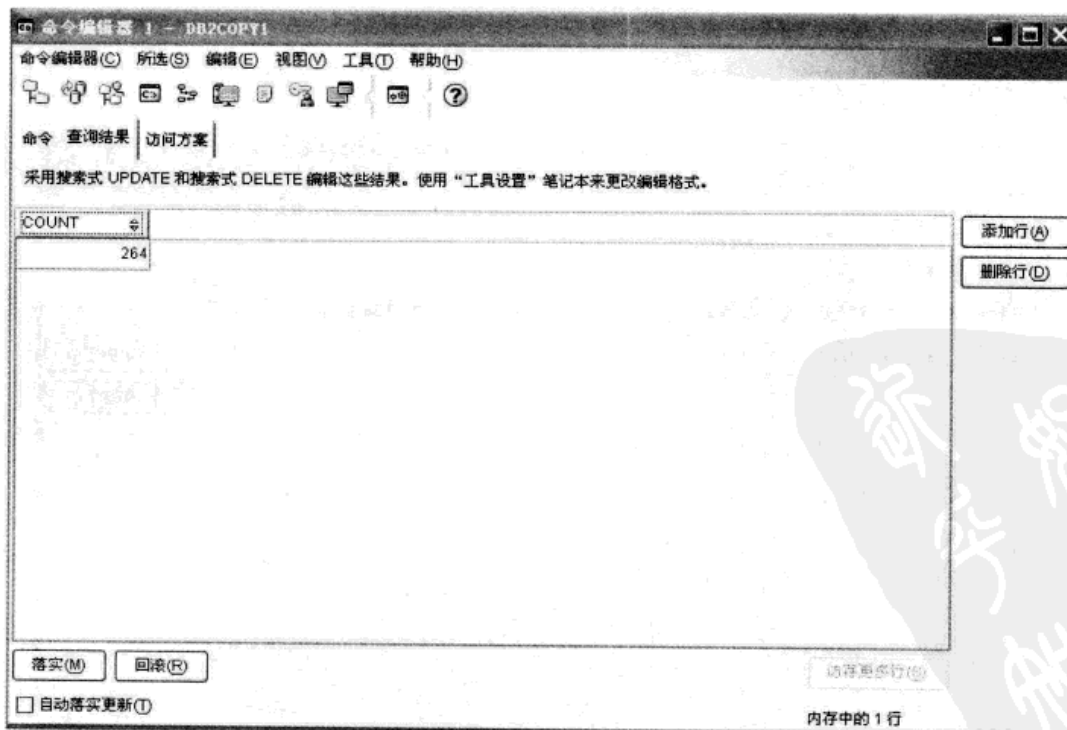


图 5-10 查询结果



根据查询语句,其查询结果如图 5-11 所示。

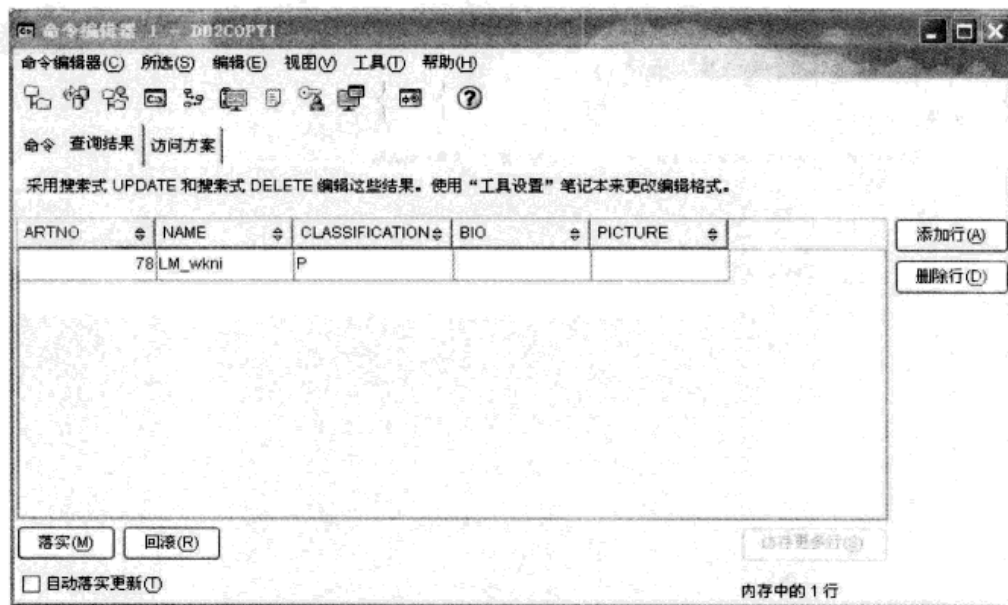


图 5-11 查询结果

2) 查询 ALBUMS 表中,所有唱片名字第三个字母不是'C'的唱片的信息

```
SELECT *
FROM ALBUMS
WHERE TITLE NOT LIKE '__a%'
```

根据查询语句,其查询结果如图 5-12 所示。

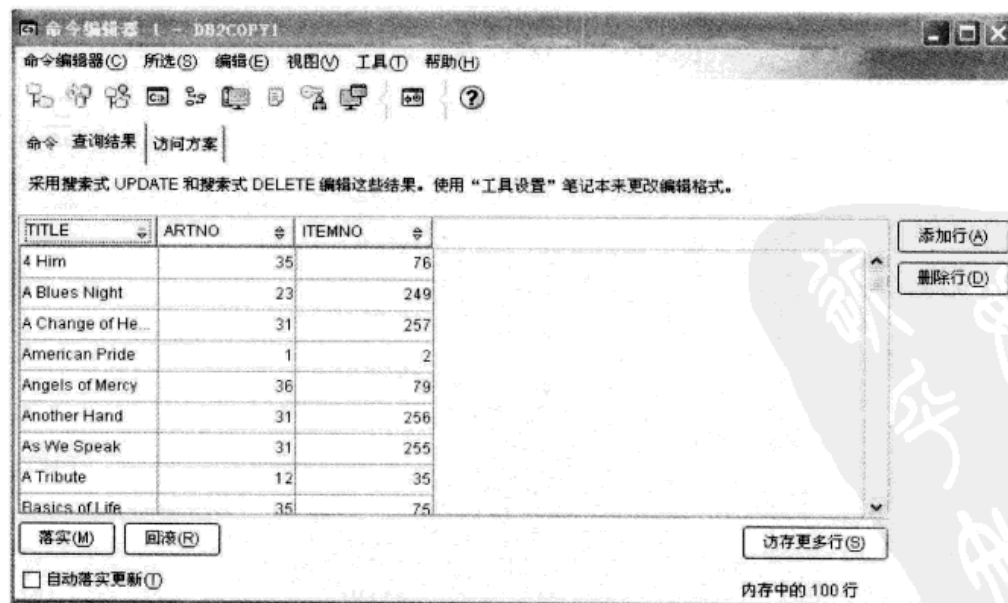
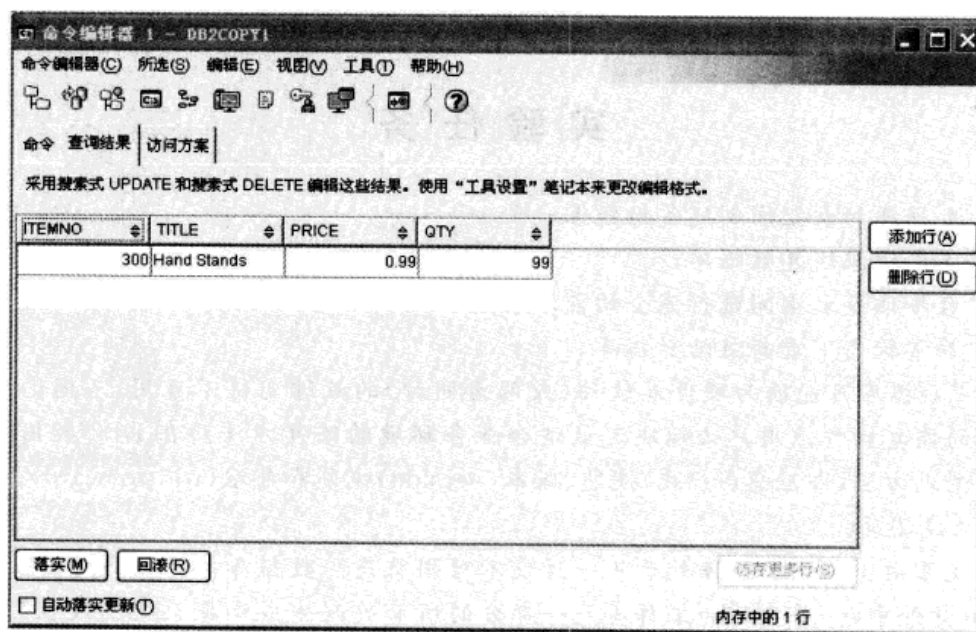


图 5-12 查询结果

3) 查询 STOCK 表中价格最低的唱片的编号、名称和数量

```
SELECT ALBUMS.ITEMNO, TITLE, PRICE, QTY
FROM STOCK, ALBUMS
WHERE STOCK.ITEMNO = ALBUMS.ITEMNO AND PRICE = (SELECT MIN (PRICE)
FROM STOCK)
```

根据查询语句,其查询结果如图 5-13 所示。



| ITEMNO | TITLE       | PRICE | QTY |
|--------|-------------|-------|-----|
| 300    | Hand Stands | 0.99  | 99  |

图 5-13 查询结果

### 实验任务

- 为数据库和表空间创建备份镜像；
- 从备份镜像恢复数据库；
- 对数据库或表空间进行完全的前滚恢复；
- 在异常状态下重新激活数据库。

**注意：**数据库可能因为硬件或软件(或两者同时)的故障变得不可用,不同的故障情况需要不同的恢复操作。用户必须决定最适合业务环境的恢复方法。在 DB2 数据库中恢复有 3 种类型或方法,即应急(crash)恢复、版本(version)恢复和前滚(roll forward)恢复。

#### (1) 应急恢复

应急恢复用于防止数据库处于不一致或不可用状态。数据库执行的事务(也称工作单元)可能被意外中断,若在作为工作单元一部分的所有更改完成和提交之前发生故障,则该数据库就会处于不一致和不可用的状态。这时,需要将该数据库转化为一致和可用的状态。为此,需要回滚未完成的事务,并完成当发生崩溃时仍在内存中的已提交事务。如在 COMMIT 语句之前发生了电源故障,则在下一次 DB2 重新启动并再次访问该数据库时,需要回滚到执行 COMMIT 语句前的状态。回滚语句的顺序与最初执行时的顺序相反。

#### (2) 版本恢复

版本恢复指的是使用备份操作期间创建的映像来复原数据库的先前版本。这种恢复是通过使用一个以前建立的数据库备份恢复出一个完整的数据库。一个数据库的备份允许你把数据库恢复至和这个数据库在备份时完全一样的状态。而从备份建立后到日志文件中最后记录的所有工作事务单元将全部丢失。

使用版本恢复方法,必须定期调度和执行完整数据库备份。

#### (3) 前滚恢复

这种恢复技术是版本恢复的一个扩展,使用完整的数据库备份和日志相结合,可以使一个数据库或者被选择的表空间恢复到某个特定时间点。如果从备份时刻起到发生故障时的所有日志文件都可以获得的话,则可以恢复到日志上涵盖到的任意时间点。前滚恢复需要在配置中被明确激活才能生效。

无论何种恢复的方法都会用到两种技术,即数据转储和日志文件。所有数据库都有相关的日志,这些日志保存了有关数据库更改的记录。若将数据库复原至完整、离线备份之前的一个点,则需要日志才能将数据库前滚至故障点。DB2 数据库中有两类日志——循环日

志和归档日志,它们提供了不同级别的恢复能力。

### (1) 循环日志

循环日志是 DB2 数据库缺省的日志记录策略。循环日志文件按顺序使用,当日志文件中包含的所有工作单元已经提交或回滚时,日志文件可以重用,如图 6-1 所示。

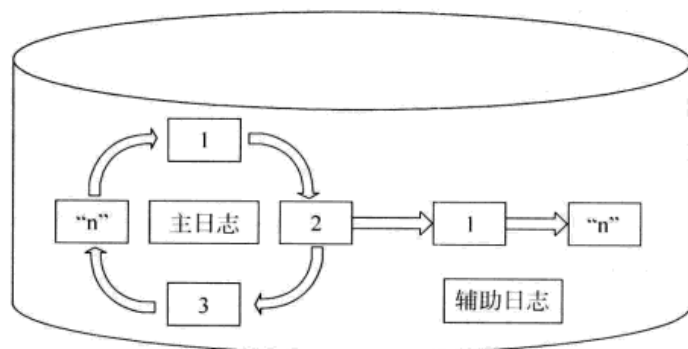


图 6-1 循环日志

循环日志使用两类日志文件,即主日志文件和辅助日志文件。主日志文件在数据库激活时分配,是预先分配的;而辅助日志文件则是在需要时分配。如果数据库管理程序按顺序请求下一个主日志,而该主日志不可重用,则将分配一个辅助日志文件。当该辅助日志文件满后,将再次检查下一个主日志文件是否可重用。如果仍然不可用,则将分配另一个辅助日志文件。此过程将一直继续,直到有主日志文件变为可重用。一旦数据库管理程序确定不再需要辅助日志文件,这些文件即被释放。

使用循环日志不能进行前滚恢复,但可进行崩溃恢复和版本恢复。

### (2) 归档日志

归档日志是一种日志管理技术,当日志文件处于非活动状态时就将它们归档。归档日志不是默认的日志方法。它是唯一支持前滚恢复和实现可恢复数据库的日志方法。有三种日志文件类型与这种方法相关联,如图 6-2 所示。

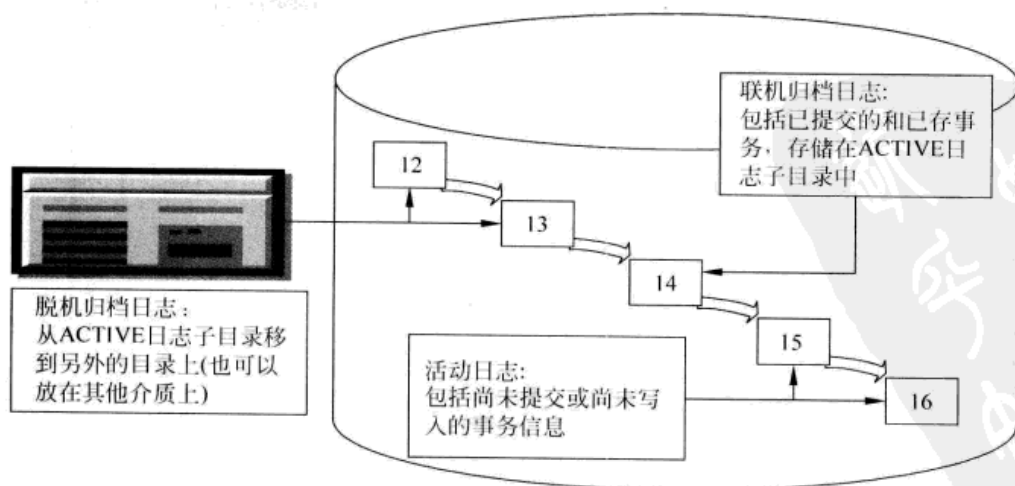


图 6-2 归档日志

归档日志文件可按如下分类：

### (1) 活动日志

如图 6-2 中数字 15 和 16 所示。这些日志文件包含了那些尚未提交(或回滚)的事务的相关信息。还包含了那些已经提交但改动尚未写入数据库文件的事务相关信息。

活动日志文件用于故障恢复,以防止产生数据库系统数据不一致的错误。RESTART DATABASE 状况(用于故障恢复)和 ROLL FORWARD 命令(用于到时间点的恢复或者到日志结尾的恢复)使用活动日志来重新应用或回滚事务单元,以使得数据库恢复到一个数据一致点上。

### (2) 联机归档日志

如图 6-2 中数字 14 所示。当活动日志记载的所有改动对正常的处理进程来说已经不再需要时,活动日志将关闭,并成为归档日志。

这些日志文件中包含了那些重启恢复时不再需要的已经完成的事务相关信息。称它们为“联机”是因为它们和活动日志存放在相同的子目录下。

### (3) 脱机归档日志

如图 6-2 中数字 12 和 13 所示。这些日志文件已经从活动日志所在的目录中移开了。移动这些文件的方法可以是手动的,也可以是通过诸如 USEREXIT 这样的过程调用来自自动进行。归档的日志文件可以是简单地通过把它们移到另外的目录或是存储到磁带或其他介质上而成为脱机模式,它们也可以通过外部的存储管理器来管理,例如 IBM 的 TSM 产品。

## 实 验 内 容

### 1. 查看 DB2 目前的日志配置参数

#### 1) 查看日志配置参数

本实验开始之前应先在命令编辑器中执行脚本 RECOVERST,将数据库的配置参数设置成本实验所需的值,如图 6-3 所示。

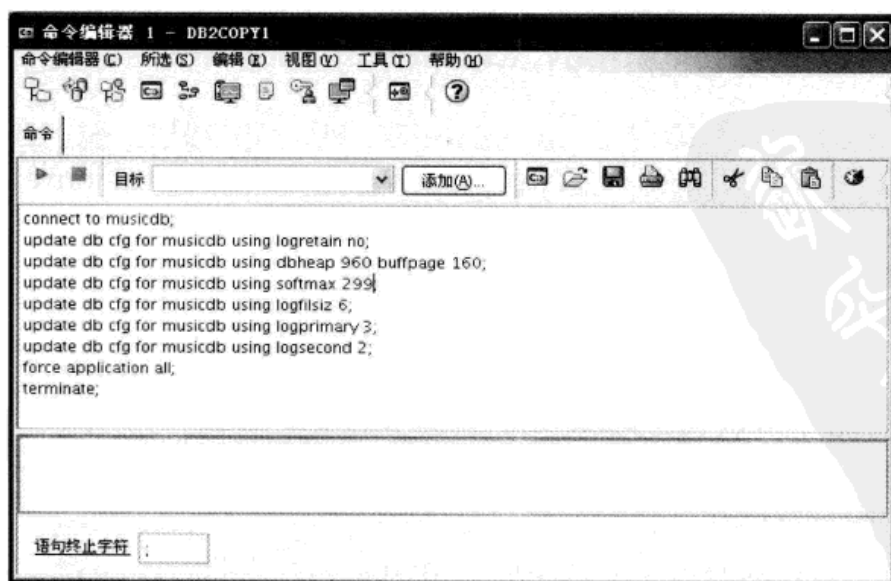


图 6-3 执行 recoverst 脚本修改配置参数

(1) 在命令编辑器中输入并执行 connect to musicdb, 激活 MUSICDB 数据库。

(2) 在控制中心中, 右击 MUSICDB 数据库, 在弹出菜单中选择“配置参数”选项, 再在“数据库配置—MUSICDB”对话框中查看“日志”选项卡的 LOGFILSIZ 的值, 该参数的值被设置为 6, 如图 6-4 所示, 小于 Windows 平台下的默认值 1024。



图 6-4 查看日志配置参数

(3) 查看主日志文件的数目, 考虑 DB2 将于什么时候分配这些主日志文件?

在“数据库配置—MUSICDB”对话框中的“日志”下可查看到 LOGPRIMARY 参数值为 3, 当发出 ACTIVE DATABASE 命令或第一个应用程序连接到数据库时, DB2 将会分配这三个主日志文件, 且其中每个日志文件的大小为 LOGFILSIZ+2 个 4KB 页, 因此所有主日志文件将占据  $3 \times (6+2) \times 4\text{KB}$ , 即 96KB 空间。

(4) 查看辅助日志文件的数目, 考虑 DB2 将于何时分配这些辅助日志文件?

辅助日志文件会在事务未提交但主日志文件写满的时候根据需要一次分配一个(直到数据库配置参数 LOGSECOND 规定的数目), 本实验中 LOGSECOND 的值设置为 2, 因此系统最多允许两个辅助日志文件。辅助日志文件的大小同样由参数 LOGFILSIZ 指定。

**注意:** DB2 数据库日志参数介绍。

① logretain——日志保留配置参数。

此参数确定是否保留活动日志文件以及这些文件是否可用于前滚恢复。

默认值 [范围]: No [Recovery; No]

No, 表示不保留这些日志。

Recovery, 指示保留这些日志, 并可用于前滚恢复。

② logfilsiz——日志文件大小配置参数。

此参数定义每个主日志文件和辅助日志文件的大小。在这些日志文件已满且需要新日志文件之前, 这些日志文件的大小限制可写入这些日志文件的日志记录数。

默认值 [范围]: Windows® 1000 [ 4~524 286 ]

计量单位: 页(4 KB)

③ logprimary——主日志文件数配置参数。

默认值 [范围]: 3 [ 2~256 ]

计量单位: 计数器。

分配时间: 创建数据库。

释放时间: 除非此参数减小, 否则不释放。如果此参数减小, 则在下一个数据库连接期间删除不需要的日志文件。

④ logsecond——辅助日志文件数配置参数。

默认值 [范围]: 2 [ 0~254 ]

计量单位: 计数器。

分配时间: 当 logprimary 不够时需要(详情请查阅帮助文档)。

释放时间: 数据库管理器确定将不再需要它们的结束时间。

2) 查看当前系统所使用的日志类型(循环日志还是归档日志)

LOGRETAIN 和 USEREXIT 可被用来设置系统采用何种日志类型, 本实验中这两个参数均被设为默认值“否”, 因此系统采用循环日志方式。如果 LOGRETAIN 被设为 Recovery 或者 USEREXIT 被设为 Yes, 那么系统将采用归档日志方式。

3) 查看日志文件的存放位置

配置参数中的 LOGPATH 指定日志文件的保存路径, 本实验的日志文件保存在 X:\INST1\NODE0000\SQL00001\SQLOGIR 下。

数据库的工作单元可能被意外地中断, 假如部分工作单元的所有变更完成和提交之前发生了电源故障, 那么系统就需要执行崩溃恢复再次回到一致状态。命令 restart database 可以实现这个功能。如果 AUTORESTART 被设置成开(默认方式), 那么崩溃发生后, 应用程序首次连接到数据库时将会自动启动崩溃恢复机制。本实验 AUTORESTART 参数被设置为开。

(1) Windows 命令提示符下, 进入目录 X:\INST1\NODE0000\SQL00001。

输入命令: dir /w, 查看该路径下有哪些文件和目录。

输入命令: cd SQLOGDIR 和 dir /w, 可以查看到该目录下有三个日志文件, 这与参数设置相符。

(2) 选择命令编辑器的“工具”标签, 再选择“工具设置”选项, 在“工具设置”对话框中选择“命令编辑器”标签, 确保“自动落实 SQL 语句”复选框未被选中, 如图 6-5 所示。

在命令编辑器中执行如下命令:

```
connect to musicdb;  
update stock set qty = qty + 1;
```

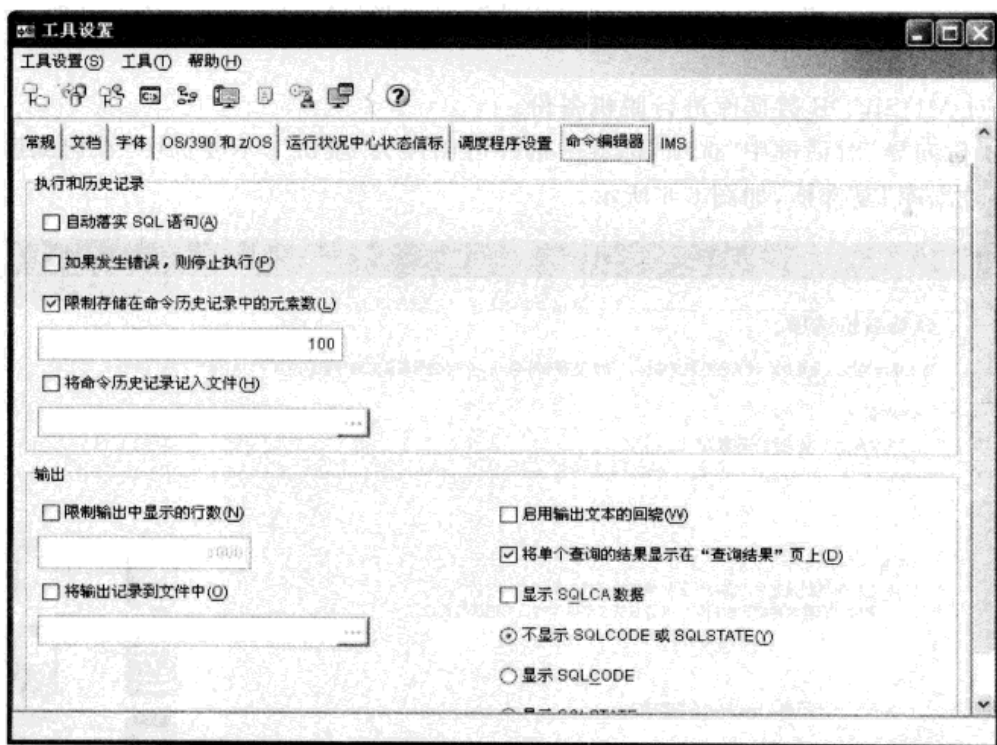


图 6-5 “工具设置”对话框

如果上步执行后系统显示错误则跳到下一步,如果没有显示错误信息,则继续执行:  
update stock set qty=qty-1,这时系统会返回 SQL0964C。

为查看详细的错误信息,输入并执行 ? SQL0964。回答为什么更新操作会失败。

(3) 选择命令编辑器的“工具”标签页,再选择“工具设置”选项,在“工具设置”窗口中选择“命令编辑器”标签页,选中“自动落实 SQL 语句”。

本实验中日志文件故意被设置得很小,所以上面的更新操作未能成功完成,在命令编辑器中输入并执行如下命令 rollback,进入日志保存目录,查看该目录下的文件如下:

```
cd X:\INST1\NODE0000\SQL00001\SQLOGIR
dir /w
```

可查看到该目录下现在多了两个文件,当采用循环日志且主日志文件已满时,DB2 就会分配这两个多出来的文件作为辅助日志文件。

## 2. 循环日志条件下的数据库恢复

在 Windows 命令提示符下输入 md \restore,在 X 驱动器下建立一个目录用来保存数据库备份。

### 1) 使用控制中心备份数据库

(1) 试图对 MUSICDB 数据库进行联机备份。

在控制中心中右击 MUSICDB 数据库,在弹出菜单中选择“备份”选项。

在“备份向导”对话框中,选择“映像”标签,选择“介质类型”为文件系统,单击“添加”按钮,路径一栏输入 x:\restore。



单击向导的“选项”标签,这时注意“可用性”下的“联机”复选框为灰色不可选,只能进行脱机备份。

(2) 对 MUSICDB 数据库进行脱机备份。

在“备份向导”对话框中“选项”标签中选择可用性为“脱机”,不要选中“在启动脱机操作之前停顿数据库”复选框,如图 6-6 所示。

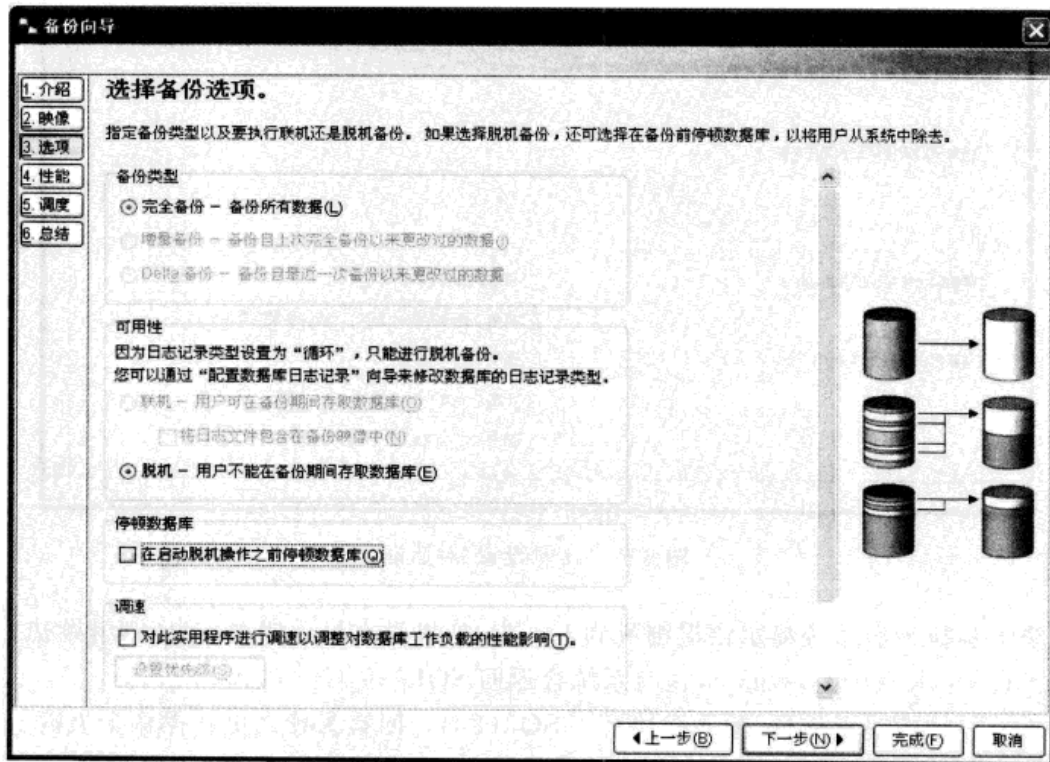


图 6-6 “备份向导”对话框

单击备份向导的“调度”标签,选择“立即运行而不保存历史任务”复选框,然后单击“完成”按钮。

系统返回 SQL01035 表明备份失败,根据提示消息可知目前正有另一个程序在连接并使用要脱机备份的数据库,单击“DB2 消息”窗口的“关闭”按钮。

为了确保所要备份的数据库的完整性,脱机备份要求独占该数据库。因为,通过命令中心连接到数据库且该连接一直保持到现在,所以脱机备份不能成功。在命令编辑器中输入如下命令强制所有用户或应用程序切断与该数据库的连接:

```
force application all;
terminate;
```

按照(2)重新脱机备份数据库 MUSICDB,这次备份成功。

(3) 记录这次备份的时间戳。

在控制中心中,选择“工具”→“日志”菜单,在“日志”对话框中选择“数据库历史”标签,单击“数据库”一栏右边的“...”按钮,选择相应的系统名、实例名、数据库名,然后单击“确定”按钮,如图 6-7 所示。



图 6-7 选择要查看的数据库

按“开始日期”排序 MUSICDB 所有事件列表,那么本次备份应在列表最上端,如图 6-8 所示,记录下该备份的开始日期及时间。

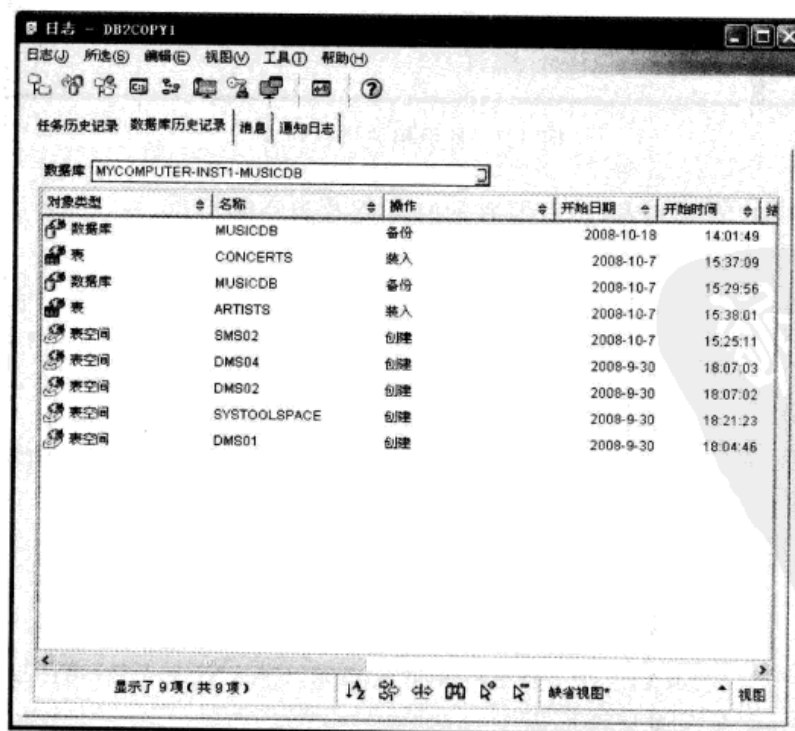


图 6-8 数据库历史记录

关闭“日志”窗口,然后回到控制中心来。

另一种查看备份相关信息的方法是使用 list backup 命令,回顾实验 5 中使用 load 工具之前也备份了数据库 MUSICDB。命令编辑器中输入如下命令:

```
terminate ;
list backup all for musicdb ;
```

备份相关信息列表如图 6-9 所示。

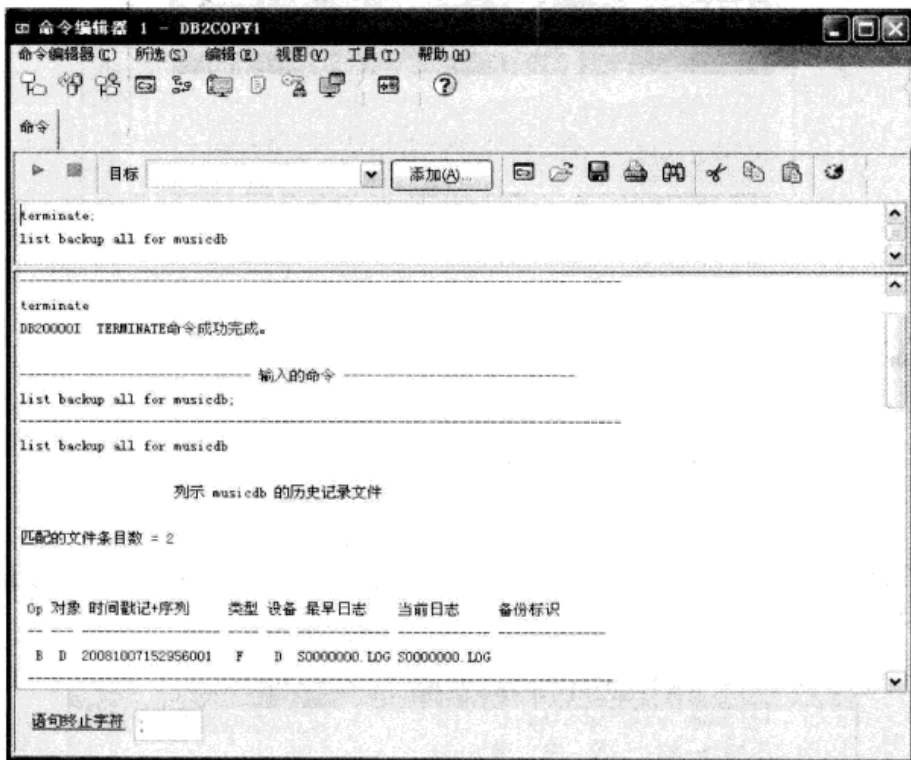


图 6-9 备份相关信息列表

查看上步命令的输出列表,注意所有的时间戳及表空间信息。

**注意:**

在 list backup 命令中可以不使用 ALL 关键字,可以指定一个特定的时间戳或对象名来限制输出结果,例如要查看某时间以后对 MUSICDB 进行的所有备份,可以输入:

```
list backup since yyyymmddhh for MUSICDB
```

(4) 在 Windows 命令提示符下输入如下命令,可以发现备份的目录结构及文件名与备份的数据库名、实例名、时间戳之间的对应关系:

```
x:
cd restore
dir
```

显示文件名为 MUSICDB.0.INST1.NODE0000.CATN0000.20081018140149.001。

2) 使用 restore 命令恢复数据库

如果要使用 restore 命令而不是控制中心来恢复数据库,且备份目录下存在数据库的多

个备份镜像,那么,为了指定用哪一份(何时产生的)备份文件,备份的时间戳信息就是必需的。

如果使用磁盘备份数据库,那么备份镜像的文件名就包含着时间戳信息,如果使用磁带或者 Tivoli Storage Manager 管理的设备作为备份的存储介质,那么备份镜像的头部也包含着相关的时间信息。

#### (1) 更新数据库中的数据。

本实验中数据库 MUSICDB 备份后其中数据还未经过任何改动,现在的数据与备份中的数据完全相同。下面对数据库中数据做一些改变。

首先执行下面 SQL 语句选择出编号为 77 的艺术家的名字和他的专辑名:

```
select name,title from artists a,albums b where a.artno = 77 and a.artno = b.artno;
```

将编号为 77 的艺术家的名字改为“Melanie and the Mechanics”,并且将编号为 261 的专辑名改为“Unmaterial Girl”:

```
update artists set name = 'Melanie and the Mechanics' where artno = 77;
```

```
update albums set title = 'Unmaterial Girl' where itemno = 261;
```

执行下列连接查询查看上步的修改结果:

```
select name,title from artists a,albums b where a.artno = 77 and a.artno = b.artno;
```

可见这次的输出与前面的查询结果不同;

#### (2) 恢复数据库。

假如现在 MUSICDB 所在的磁盘出现故障,将使用刚刚创建的备份镜像对 MUSICDB 进行恢复。在命令编辑器中输入“? restore”命令获得恢复命令的帮助信息。

恢复前为确保当前没有用户或应用程序连接到 MUSICDB 数据库,命令编辑器中输入并执行如下命令:

```
force application all;  
terminate;
```

从 X:\restore 目录下读取备份文件对 MUSICDB 进行恢复。

在控制中心中右击 MUSICDB 数据库,在弹出菜单中选择“复原”。

选择“复原数据向导”对话框的“简介”标签,选择“复原到现有数据库”复选框。

选择向导的“可用的映像”页面,选择刚刚备份的映像,然后单击“>”按钮,如图 6-10 所示。

选择向导的“调度”标签,选择“立即运行而不保存历史任务”复选框。

选择向导的“总结”标签,单击“显示命令”按钮查看全部 restore 命令。

单击向导的“完成”按钮,等待一段时间后系统弹出“DB2 消息”对话框显示命令成功完成,单击“确定”按钮关闭该消息窗口。

#### (3) 查看恢复后的数据库。

重新连接到数据库,输入并执行下列连接查询语句:

```
connect to MUSICDB;  
select name,title from artists a,albums b where a.artno = 77 and a.artno = b.artno;
```

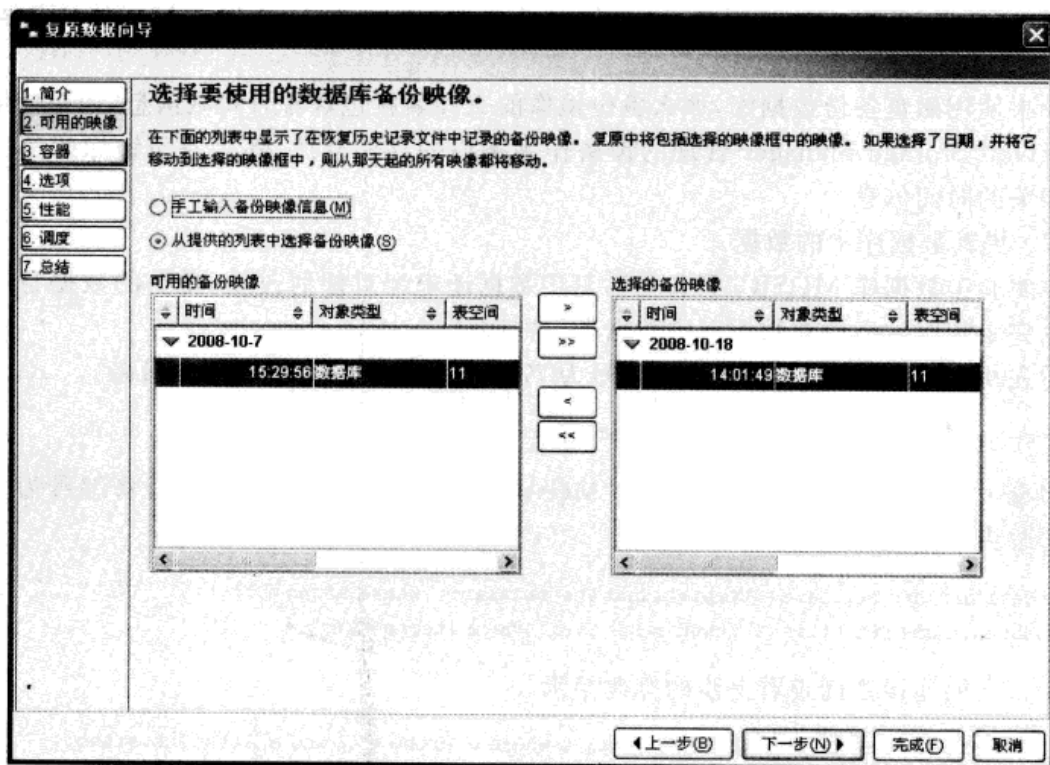


图 6-10 选择数据库备份映像

可见这次查询的输出结果反映的是更新前的原数据,所以恢复操作成功完成。

上面的恢复操作虽然复原了更新前的数据,但是因为 DB2 使用了循环日志方式,无法前滚,所以无法救回备份后对数据库所作的更新。

### 3. 归档日志条件下的数据库恢复

在 Windows 命令提示符下输入如下命令创建数据库的备份目录(X:\RESTORE):

```
MD \RESTORE
```

更改数据库配置文件以采用归档日志方式。

在命令中心中输入并执行下列命令:

```
update db cfg for musicdb using logretain recovery
```

### 4. 重新设置实验环境

完成下列任务以确保剩余实验的环境设置正确,在命令中心中导入并执行 X:\cf23\CLREC 脚本。

检查数据库配置文件,确保目前数据库采用循环日志方式。

在命令编辑器中输入并执行下列语句:

```
connect to musicdb;
get db cfg for musicdb;
```

可见输出结果中“恢复状态的日志保留”为 NO。

## 实验 7

## 管理权限

### 实验任务

- 熟悉 DB2 UDB 中 PUBLIC 组的默认特权;
- 学会管理 DB2 权限(SYSADM,SYSCYRL,SYSMANT 和 DBADM),熟悉各种权限的操作能力;
- 能够对单个用户和组授予特权;
- 清楚何时组特权可用和何时需要个人特权等。

**注意:**安全管理是管理数据库的重要工作。当用户对 DB2 下达任何指令,或是执行任何 DB2 的应用程序时,都需要经过两道安全控管的程序:

(1) 身份鉴别:验证用户的身份。

(2) 权限控制:在确认用户的身份后,再检查用户是否有权执行所下达的指令。

此外,还可以通过以下三种方式来进行安全性控制:

(1) 视图:把要保密的数据对无权存取的用户隐藏起来,从而自动地对数据提供一定程度的保护。

(2) 审计追踪:对重要的数据实行审计追踪技术,监测可能的不法行为。审计追踪是一个日志,记录对数据库的所有更新行为,以及哪个用户执行了更改和什么时候执行的更改等。

(3) 数据加密:使用加密技术对重要的数据进行加密。

这里我们简单介绍一下前两种安全验证方法。

#### 1. 身份鉴别

(1) 鉴别机制:DB2 用户的身份鉴别是由 DB2 所在的操作系统或是操作系统上的安全管理软件来进行。

(2) 鉴别时机:在用户连接数据库时和下达 DB2 指令时。

例如,执行以下指令时,将进行身份鉴别。

```
CONNECT TO SAMPLE USER DAVY USING P123
```

```
BACKUP DATABASE SAMPLE TO D: \ BACKUP USER DAVY USING P123
```

如果不指定用户名和密码,则使用用户登录系统时所用的账号和密码进行验证。

(3) 身份验证的方式:

主要是指,在 C/S 系统中,客户登录的时候,验证是在客户端检查还是在服务器端检查、客户端和服务端传送账号和密码的方式是什么?

具体方式请参考实验 1 中的介绍。

可使用 GET DBM CFG 来检查身份鉴别方式的设定方式。

## 2. 权限控制

通过身份验证后,下一个步骤是分辨这个用户的权限等级、是否有权执行他所下达的命令。

DB2 中的权限控制主要分权限和特权两个等级。权限是在数据库上、实例层进行控制的;特权是在数据库内进行控制的。

(1) 权限。

① SYSADM: DB2 实例中的最高权限。有此权限的用户可以下达任何 DB2 指令、存取任何数据库中的任何数据。

② SYSCTRL: 数据库维护者的权限。可以进行一些数据库维护的工作,但是不能存取数据库中的数据。

③ SYSMAINT: 与 SYSCTRL 类似。

④ DBADM: 是一个数据库的最高权限者。拥有这个权限的用户可以控制数据库中的任何对象,并执行一些优化命令等。

(2) 特权有 SELECT、INSERT、UPDATE、DELETE 等。

(3) 权限和特权的授予和收回。

SYSADM、SYSCTRL、SYSMAINT 都是以组为授予对象,且 SYSADM 权限,是在建立实例时,就授予了已默认设置的组。WINDOWS 下是 ADMINISTRATOR 组,UNIX 下是实例拥有者所属的组。

可以使用 UPDATE DBM CFG 来设定 SYSADM、SYSCTRL、SYSMAINT 权限。

例如:

```
UPDATE DBM CFG USING SYSADM_GROUP GROUP1
UPDATE DBM CFG USING SYSCTRL_GROUP GROUP2
UPDATE DBM CFG USING SYSMAINT_GROUP GROUP3
```

可以使用 GET DBM CFG 来查看权限设定的方式。

对于 DBADM 权限,则需要由拥有 SYSADM 的权限者,授权给他人。

例如,用户 USER1 拥有 SYSADM 权限,在他登陆到数据库后,可以将 DBADM 权限给用户 USER2。

```
CONNECT TO SAMPLE USER USER1 USING PASSWORD
GRANT DBADM ON DATABASE TO USER2
```

数据库权限及对象的特权都可以使用 GRANT 和 REVOKE 两个指令进行授予或收回。

(4) PUBLIC 组。

PUBLIC 组是 DB2 中的一个特别的组,它代表所有的用户。当一个数据库建立时,会自动授予 PUBLIC 组下列的权限。

CREATETAB: 创建表的权限。

BINDADD: 允许在数据库中创建新的程序包的权限。

CONNECT: 连接到数据库的权限。

IMPLICIT\_SCHEMA: 创建表格时, 如果表格的模式不存在, 可以自动建立模式的权限。

USERSPACE1: 表空间的使用权。

所有系统表格的 SELECT 权限。

如果不想让 PUBLIC 组拥有对数据库的这些默认权限, 在建立数据库之后, 可以用 REVOKE 命令去除这些默认的权限。

(5) 也可以使用 GUI 接口进行权限和特权的设定。

权限的设定: 在控制中心中, 右击实例名→配置参数→进行 SYSADM、SYSCTRL、SYSMAINT 的设置。

特权的设定: 在控制中心中, 右击某个对象名→特权→, 对某个用户或组进行对此对象特权的设定。

## 实 验 内 容

### 1. 默认的 PUBLIC 权限

#### 1) 建立新用户 user1

(1) 使用 Windows 控制面板中的用户管理程序按下列要求创建一个新用户。

用户名: user1。

密码: user1。

全名: USER1。

描述: DB2 User。

(2) 在命令编辑器中输入如下命令确保没有应用程序或用户连接到 INST1 实例。

```
force application all;  
terminate;
```

(3) 将 stock 表的更新权限授予 user1 用户。

```
connect to musicdb;  
grant update on stock to user1;
```

#### 2) 从 DB2 的控制中心中, 确定当前数据库的所有用户的默认权限

在控制中心中, 选择 MUSICDB 数据库, 右击选择“权限”, 然后分别选择“组”和“用户”标签, 查看有哪些组 and 用户以及它们分别具有什么样的权限, 如图 7-1 和图 7-2 所示。

由图 7-2 可以看出, 具有授权的任一个 ID 用户可以对 MUSICDB 数据库进行的操作包括: 创建数据库、添加新的 PACKAGES, 连接到数据库, 以及隐式地创建模式等。

(1) 如图 7-3 所示, 对于 PUBLIC 组, NOFENCE 列包含一个 not 符号, 这表示 PUBLIC 组不具有创建 NOT FENCED 用户自定义函数的权限, 和创建用户自定义函数相关的另一个命令 CREATE EXTERNAL ROUTINE 也不缺省授予 PUBLIC 组。

(2) 在 LOAD 列上也具有 not 符号, 表示 PUBLIC 组不能够进行 LOAD 操作, 只有 SYSADM, DBADM, 或在数据库上具有 LOAD 及与 LOAD 操作相关的其他操作权限的用户, 可以进行 LOAD 操作。



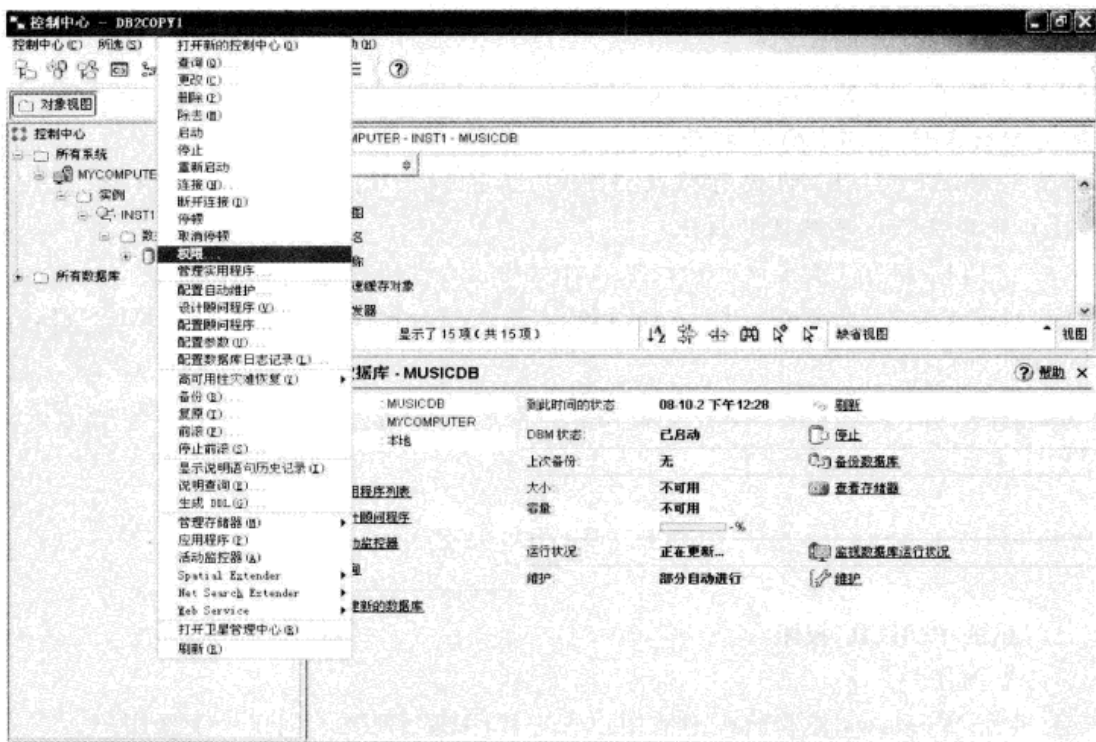


图 7-1 打开数据库的权限选项

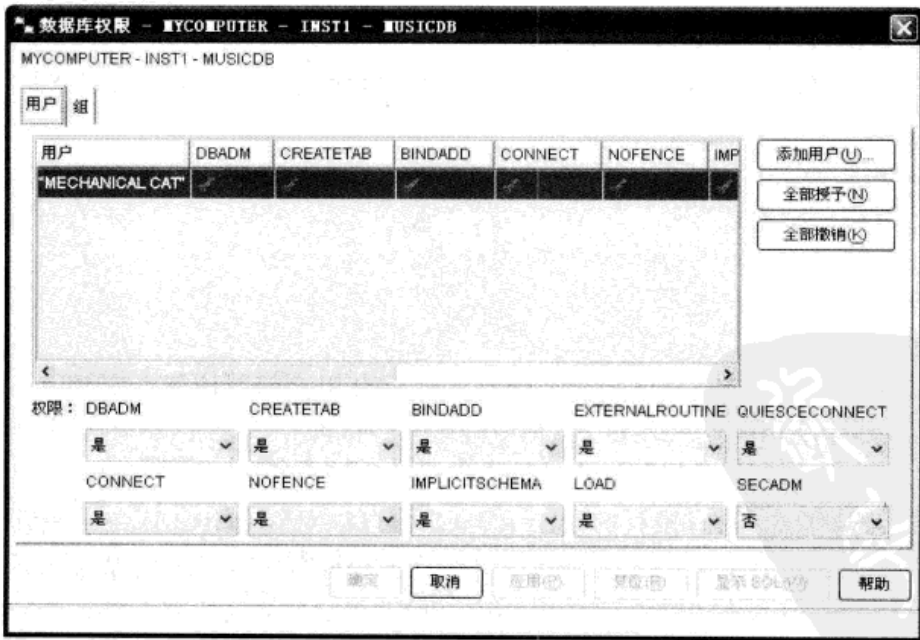


图 7-2 查看用户的权限

- (3) 要查看所有的授权信息,可以通过 SYSCAT.DBAUTH 视图来查看。
- 3) 建立一个以不同用户连接到数据库 MUSICDB 的对话
- 打开 DB2 命令窗口,输入 title user1。以 user1 用户的身份连接到数据库 MUSICDB;

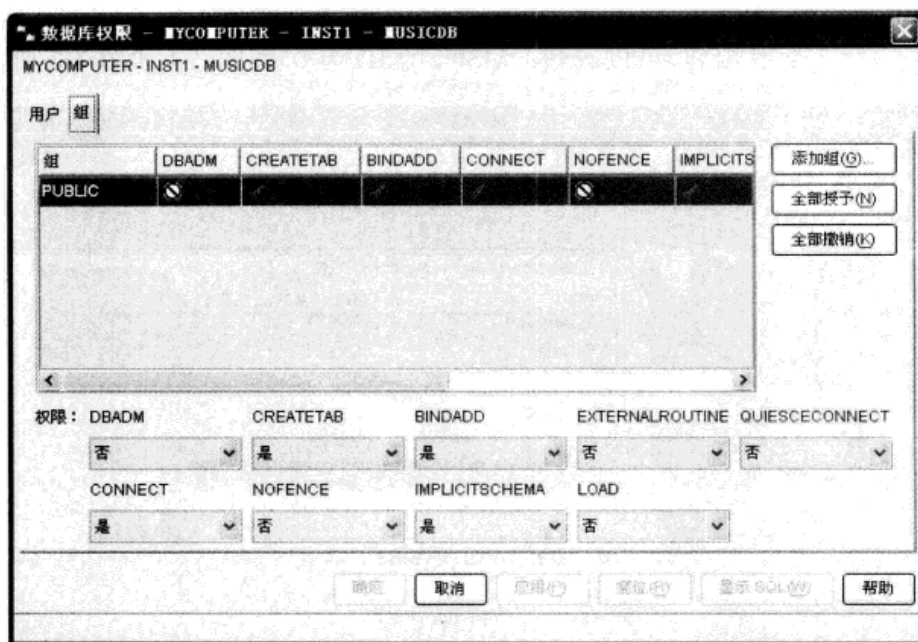


图 7-3 查看组的权限

```
db2 connect to musicdb user user1 using user1
```

输入下列 SQL 语句：

```
db2 "select substr(tspace,1,18) from syscat.tablespace "
```

对于 MUSICDB 数据库,由于 PUBLIC 组中,SELECT 是默认的特权,自然 user1 用户也能够执行查询操作。

#### 4) 验证建表权限

(1) 在命令窗口输入下列 SQL 语句：

```
db2 "create table test1 (partno integer,subpart integer) in DMS04"
```

该语句不能够成功执行,因为 user1 用户在 DMS04 表空间上没有 USE 特权。

(2) user1 用户要想实现上述操作,可以通过其他用户对其授权实现,假设 corp\_spy 用户具有相应的权限,可以实现对 user1 用户的授权。在 DB2 的命令窗口中输入下列命令即可：

```
title corp_spy
db2 connect to musicdb
db2 grant use of tablespace dms04 to user user1
```

(3) user1 用户可以重新输入下列命令来实现上述操作：

```
db2 "create table test1 (partno integer,subpart integer) in DMS04"
```

此时,该语句就会成功执行。

#### 5) 在命令编辑器中,列出表空间 DMS04 中所创建的所有表

(1) user1 可以通过下列方法来查看 DMS04 表空间的 ID。

在控制中心中的 MUSICDB 数据库下选择视图,然后选择 TABLESPACES 视图,右击选择“打开”选项,查看其内容,可以看到 DMS04 的 ID 为 6,如图 7-4 和图 7-5 所示。

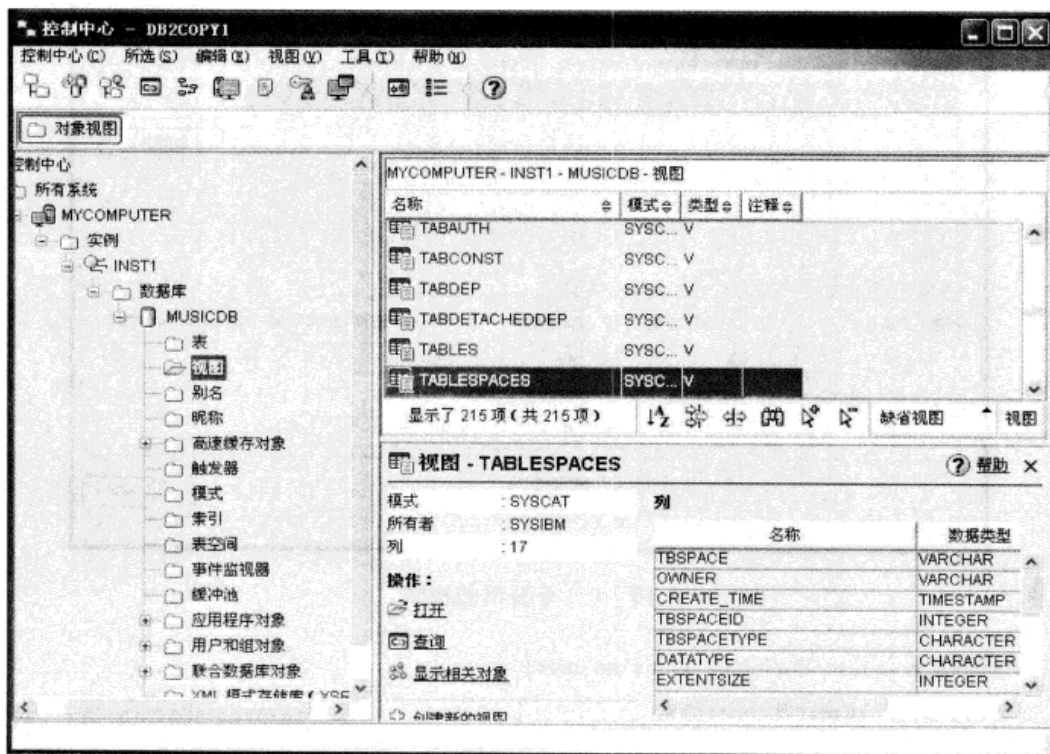


图 7-4 TABLESPACES 视图



图 7-5 查看 DMS04 的 ID

(2) 在命令编辑器的交互页面,输入下列命令。

```
minate
connect to musicdb
select tabschema, name from syscat.tables where tbpaceid = 6
```

6) 以用户 user 身份,通过 GUI 将表 user1.test1 从数据库中删除

这里需要清楚的是,user 之所以能够删除该表,是因为该用户具有系统管理员权限。

7) 撤销 PUBLIC 的默认权限

在控制中心中,在 MUSICDB 数据库的“用户和组对象”选项左侧单击“+”号,选择 DB Group,然后在内容部分选择 PUBLIC,右键选择 change,然后去掉其连接数据库、创建表、创建包等选项。

8) 以 user1 用户身份创建表 test1

```
db2 create table test1 (partno integer, subpart integer) in DMS04
```

由于 CREATETAB 特权被撤销,所以用户 user1 不能够再创建该数据表。此时,user1 与数据库的连接仍然处于活动状态。

## 2. DB2 的权限

(1) 使用 user 身份创建 4 个新用户。现在,user 既是 Windows 的管理员,又是 DB2 的系统管理员。用 user 的 Windows 管理权限创建 4 个新的用户: usersys, userctl, usermnt 和 userdba,密码与用户名相同。在 DB2 的命令窗口,输入下列命令:

```
net user usersys usersys /add
net user userctl userctl /add
net user usermnt usermnt /add
net user userdba userdba /add
```

如图 7-6 所示,4 个用户创建完成。

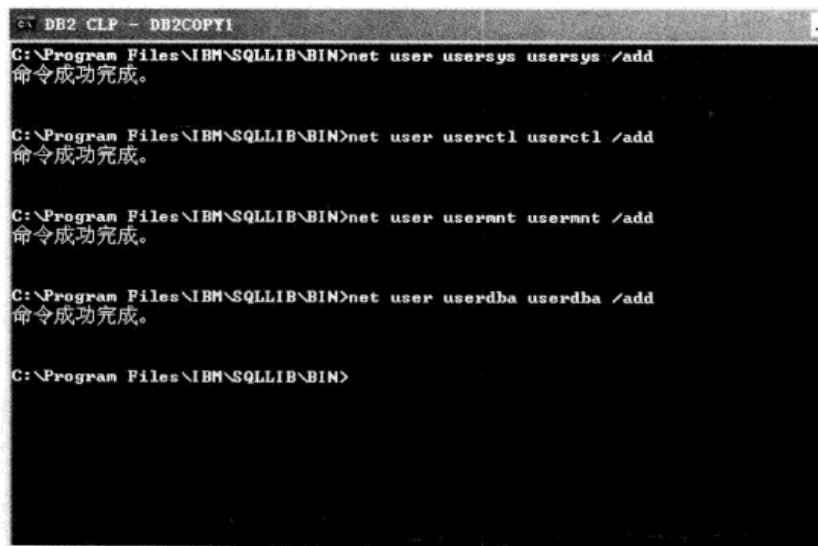


图 7-6 创建用户

然后输入命令: net localgroup STAFF /add,建立一个新组 STAFF。如图 7-7 所示,组 STAFF 创建完成。

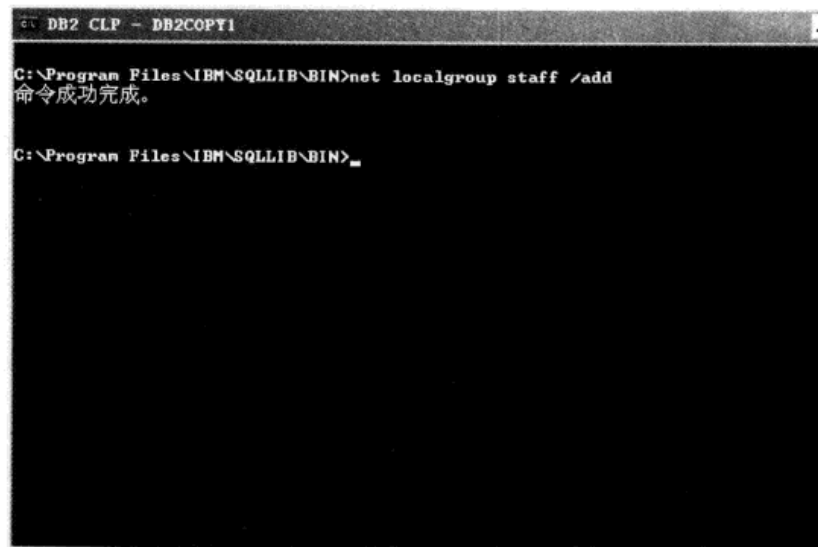


图 7-7 添加新组

接着执行命令:

```
net localgroup STAFF userdba /add
```

此命令将 userdba 加入到组 STAFF 中。

下面以 usersys 来做数据库系统管理员。首先通过 inst1 实例的数据库管理器配置参数,查看一下当前的系统管理员组是否是 ADM1。

(2) 在 DB2 命令窗口中输入下列命令,查看 ADM1 组中的用户,如图 7-8 所示。

```
net localgroup ADM1
```

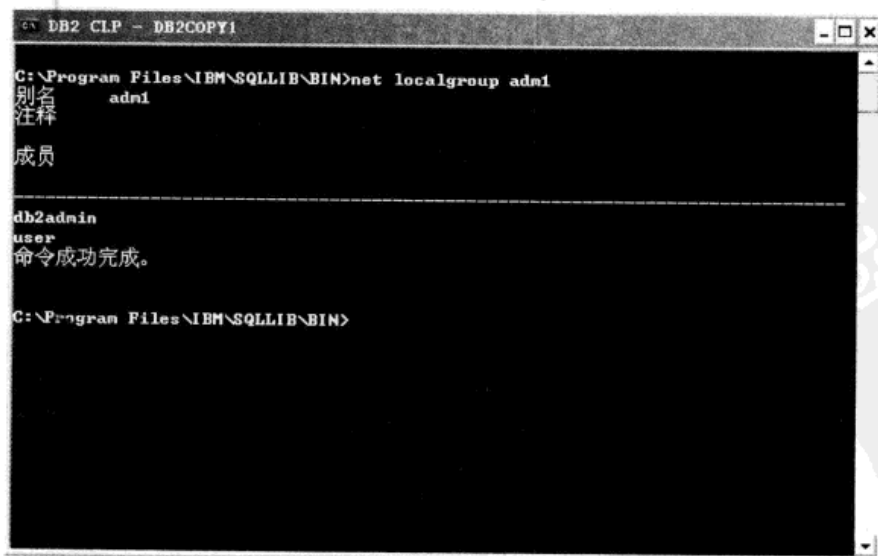


图 7-8 查看 ADM1 组成员

这些用户具有数据库管理员权限。要是其他用户也要具有该权限,需要将其加入到 ADM1 组中。

(3) 在 DB2 命令窗口中输入下列命令,将 usersys 添加到 ADM1 组中,并可通过 net localgroup ADM1 命令查看加入 usersys 用户后 ADM1 组中用户的变化,如图 7-9 所示。

```
net localgroup ADM1 usersys /add
```

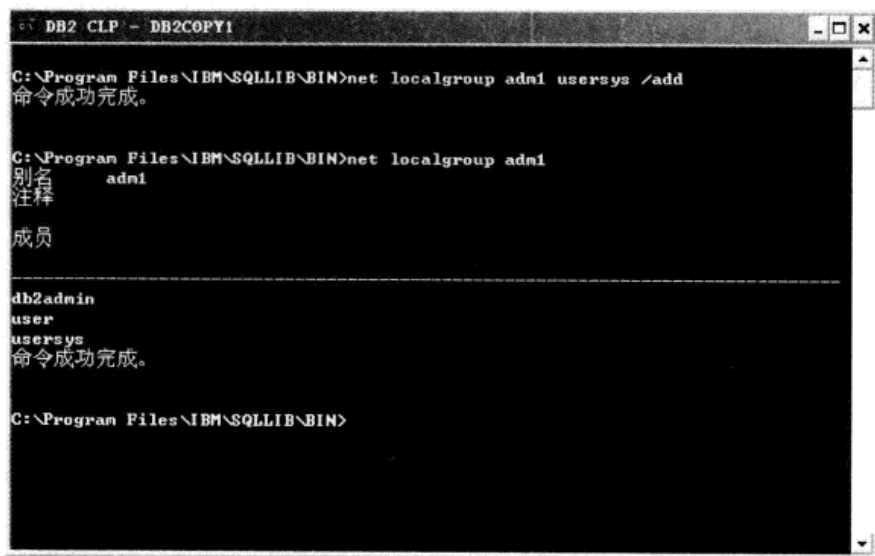


图 7-9 重新查看 ADM1 组成员

(4) 在 DB2 命令窗口中输入 title usersys。

(5) 以 usersys 身份完成下列命令,可以确认 usersys 已经具有相应的权限。

```
db2 connect to musicdb user usersys using usersys
db2 select name from user.artists
db2 grant delete on user.artists to public
db2 revoke delete on user.artists from public
db2 get snapshot for locks on musicdb |more
```

虽然在第一部分中已经撤销了部分 public 特权,但这些语句全部能够成功执行,因为数据库系统管理员可以完成任何工作。

(6) 作为具有 SYSADM 权限的用户,usersys 可以更新数据库管理器配置文件。如:

```
db2 update dbm cfg using sysctrl_group ctrl1 sysmaint_group maint1
```

该更改生效,需要执行下列语句:

```
db2 force application all
db2 terminate
db2stop
db2start
```

从控制中心,可以查看到更改后的配置参数。

(7) 在 DB2 命令窗口,创建一个名为 CTRL1 的组,并为之添加一个成员 userctl,然后

创建一个 MAINT1 组,并添加一个用户 usermnt。

```
net localgroup CTRL1 /add
net localgroup MAINT1 /add
net localgroup CTRL1 userctl /add
net localgroup MAINT1 usermnt /add
```

(8) 在实验 1 中,为实例 inst1 配置了管理参数 SYSADM\_GROUP 的值为 ADM1,这表示 ADM1 组中的成员都具有 SYSADM 的权限。可以用同样的方法使参数 SYSCTRL\_GROUP 和 SYSMANT\_GROUP 的值为 CTRL1 和 MAINT1,则 CTRL1 和 MAINT1 就具有 SYSCTRL 和 SYSMANT 权限,如图 7-10 所示。

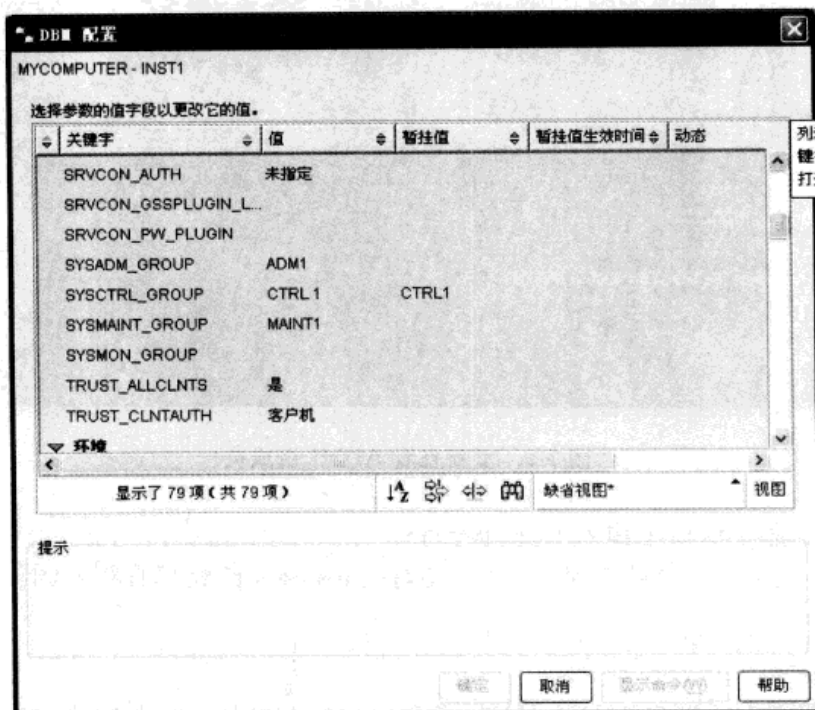


图 7-10 命令中心中配置 SYSCTRL\_GROUP 参数

同时,也可以在 usersys 的命令窗口中,以命令的方式来设置,如图 7-11 所示。

```
db2 update dbm cfg using sysctrl_group ctrl1 sysmaint_group maint1
```

可以使用 db2 get authorizations 命令来查看 usersys 的权限,如图 7-12 所示。

**注意:** 直接权限意味着此权限是明确授予此用户的。间接权限意味着此用户属于具有此权限的组。如果用户被明确授予了此权限,同时又属于具有此权限的组,那么他同时具有直接和间接权限。

(9) 在 userctl 的命令窗口中,输入下列语句以查看该用户具有的权限:

```
db2 connect to musicdb user userctl using userctl
db2 get authorizations
```

可以发现,该用户具有 SYSCTRL 权限,即系统控制权限。

(10) 在 usermntl 的命令窗口中,输入下列语句以查看该用户具有的权限:

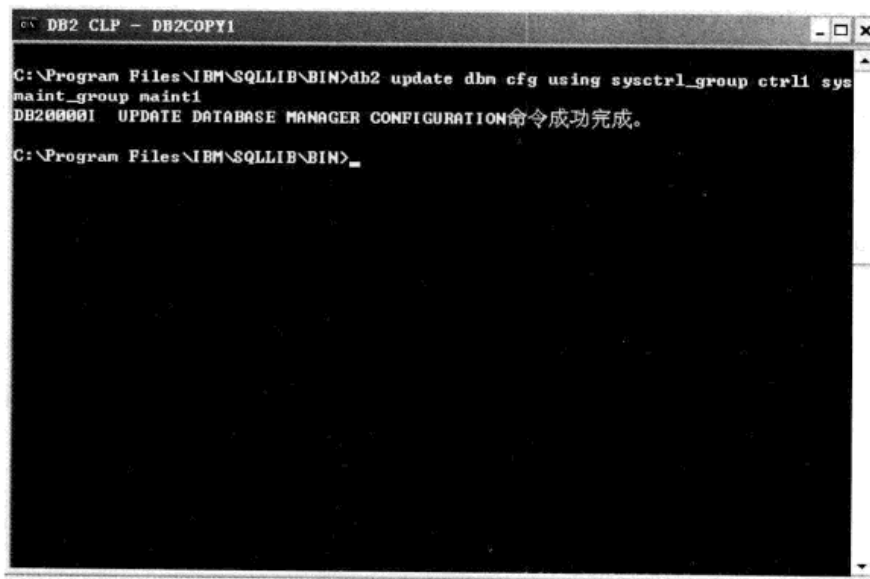


图 7-11 命令窗口中配置 sysctrl\_group 参数

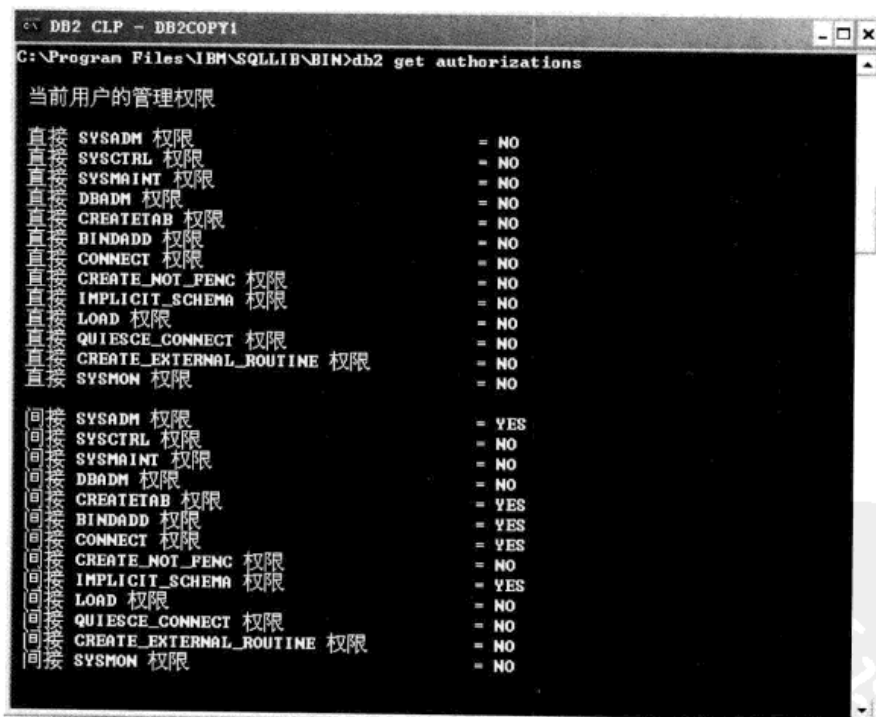


图 7-12 查看 usersys 的权限

db2 connect to musicdb user usermnt using usermnt  
db2 get authorizations

可以发现,该用户具有 SYSMAINT 权限,即系统维护权限。

(11) 在 userctrl 的命令窗口中,输入下列语句(此时不要做其他任何授权操作):

db2 create table test1 (col1 char(1))



```
db2 drop table inst1.reorder
db2 select * from inst1.music
```

这三个语句均不能够成功执行,都会遇到权限错误。因为 SYSCTRL 权限不能够访问数据库对象,而 SYSMANT 权限是 SYSCTRL 的子集,对 SYSCTRL 不能完成的操作,SYSMANT 也不能实现。

(12) 在 userctl 的命令窗口中,执行下列 SQL 语句:

```
db2 select name from user.artists
```

该语句会成功执行,因为从表 artists 中查询数据的特权被授予了 PUBLIC 组,该特权可以通过控制中心进行查看。需要清楚的是,和 SYSCTRL 权限相关的特权不是对特定表的,而是面向实例的。

(13) 在 userctl 的命令窗口中,为 MUSICDB 数据库指定一个新的别名:

```
db2 catalog db musicdb as ctrlsdb
```

(14) 在 userctl 和 usermnt 的命令窗口中输入 db2 terminate 命令。

(15) 将 userdba 添加为数据库管理员。在控制中心,右击 MUSICDB 数据库,选择 Authorities,然后单击 ADD 命令,在用户页面,选择用户 userdba,并为其添加 DBADM 权限。添加后可以通过控制中心,在 DBAUTH 视图中查看到该用户的权限,如图 7-13 所示。

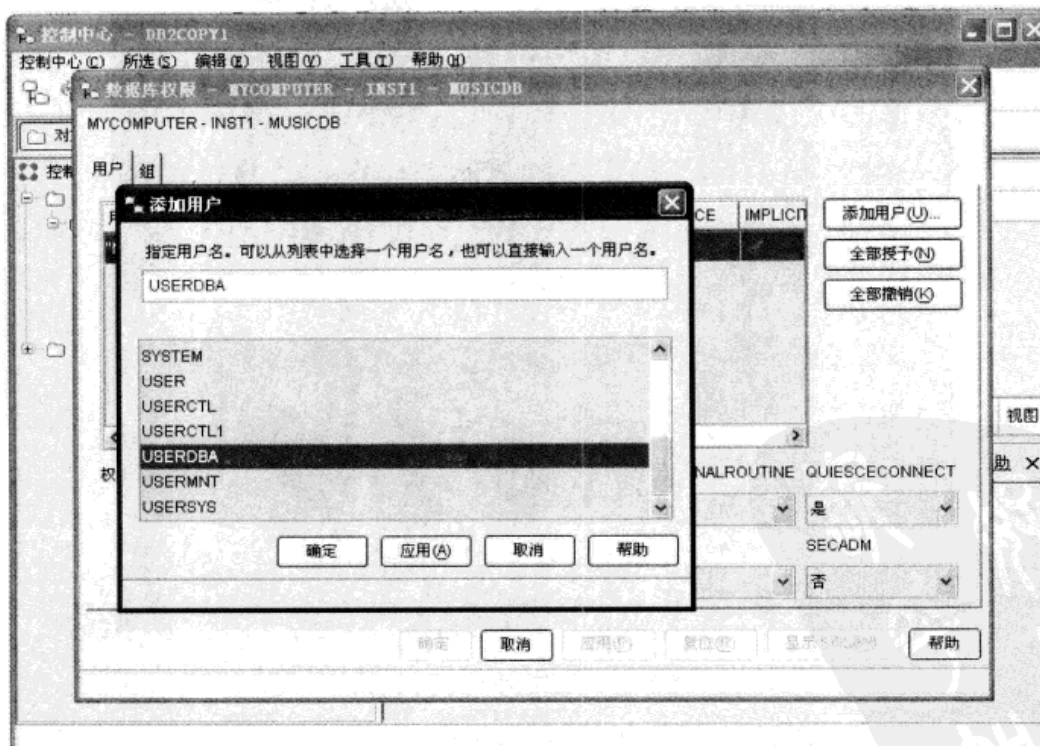


图 7-13 配置 DBADM 权限

(16) 将 userctl 的命令窗口标题改为 userdba。

Title userdba

(17) 在 userdba 的命令窗口中,输入下列命令,连接到 MUSICDB 数据库,并执行如下操作:

```
db2 connect to musicdb user userdba using userdba
db2 select name from user.artists
db2 create table test1 (col1 char(1))
db2 create table user.dlc (col1 char(1))
db2 drop table user.dlc
```

### 3. DB2 用户和组特权

(1) 以 userdba 用户身份,将下列内容插入到 user.artists 表中。

```
db2 insert into user.albums values('DB2 Connect is My Life',99, 310)
db2 insert into user.albums values('Rustle Up', 99, 311)
```

(2) 系统维护权限用户 usermnt 没有 SQL 特权,从 userdba 的命令窗口对其授予 delete 特权:

```
db2 grant delete on inst1.albums to usermnt
```

(3) 从 usermnt 的命令窗口,连接到数据库 MUSICDB,并执行表 user.albums 的删除行操作:

```
db2 connect to musicdb user usermnt using usermnt
db2 delete from inst1.albums where itemno = 310
```

(4) 如果上述删除操作被 user 发现,并且 user 通过与授权相似的方式,将 userdba 的 DBADM 权限收回。同时,userdba 想将删除的内容重新插入:

```
db2 insert into user.albums values('DB2 Connect is My Life',99, 310)
```

但是该操作是被禁止的,因为 userdba 不再具有数据库管理权限,不能再对其他用户模式下创建的数据库对象进行操纵。

(5) 如果 userdba 是数据库对象的创建者,当被撤销数据库管理权限时,仍可执行如上的操作。如对 userdba 创建的表 test1,仍可进行下列操作:

```
db2 select * from test1
db2 drop table test1
```

(6) 通过 db2 terminate 和 exit 命令分别关闭用户 userdba、usermnt、usersys 等对话窗口。

## 参 考 文 献

1. 杨鑫华,丁传华,赵慧敏. 数据库原理与 DB2 应用教程. 北京: 清华大学出版社, 2007.
2. 庄济诚, DB2 数据库管理与应用教程. 北京: 清华大学出版社, 2004.
3. IBM 公司. DB2 通用数据库 (UDB) V5 实用指南. 北京: 电子工业出版社, 1998.
4. IBM 公司. DB2 Universal Database Administration Workshop for Windows. IBM Certified Course Material (Course Code CF231).
5. IBM 公司. Relational Database Design. IBM Certified Course Material (Course Code CF18).
6. IBM 公司. DB2 Family Fundamentals. IBM Certified Course Material (Course Code CF03).
7. IBM 公司. DB2 SQL Workshop. IBM Certified Course Material (Course Code CF12).



## 读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室 计算机与信息分社营销室 收  
邮编：100084 电子信箱：jsjc@tup.tsinghua.edu.cn  
电话：010-62770175-4608/4409 邮购电话：010-62786544

教材名称：DB2 数据库实践基础教程

ISBN 978-7-302-21221-8

### 个人资料

姓名：\_\_\_\_\_ 年龄：\_\_\_\_\_ 所在院校/专业：\_\_\_\_\_

文化程度：\_\_\_\_\_ 通信地址：\_\_\_\_\_

联系电话：\_\_\_\_\_ 电子信箱：\_\_\_\_\_

您使用本书是作为：☐指定教材 ☐选用教材 ☐辅导教材 ☐自学教材

您对本书封面设计的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议\_\_\_\_\_

您对本书印刷质量的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议\_\_\_\_\_

您对本书的总体满意度：

从语言质量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

从科技含量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

本书最令您满意的是：

☐指导明确 ☐内容充实 ☐讲解详尽 ☐实例丰富

您认为本书在哪些地方应进行修改？（可附页）

\_\_\_\_\_

您希望本书在哪些方面进行改进？（可附页）

\_\_\_\_\_

## 电子教案支持

敬爱的教师：

为了配合本课程的教学需要，本教材配有配套的电子教案(素材)，有需求的教师可以与  
我们联系，我们将向使用本教材进行教学的教师免费赠送电子教案(素材)，希望有助于教学  
活动的开展。相关信息请拨打电话 010-62776969 或发送电子邮件至 jsjc@tup.tsinghua.edu.cn  
咨询，也可以到清华大学出版社主页 (<http://www.tup.com.cn> 或 <http://www.tup.tsinghua.edu.cn>) 上查询。

## 21 世纪高等学校计算机教育实用规划教材 系列书目

| 书 名                           | 作 者       | ISBN 号        |
|-------------------------------|-----------|---------------|
| 32 位微型计算机原理·接口技术及其应用(第 2 版)   | 史新福等      | 9787302134039 |
| AutoCAD 实用教程                  | 张强华等      | 9787302127260 |
| AutoCAD 实用教程(配光盘)             | 张强华等      | 9787302127260 |
| Internet 实用教程——技术基础及实践        | 田力        | 9787302110668 |
| Java 程序设计实践教程                 | 张思民       | 9787302132585 |
| Java 程序设计实用教程                 | 胡伏湘等      | 9787302109600 |
| Java 语言程序设计                   | 张思民       | 9787302144113 |
| Visual Basic 程序设计基础           | 李书琴等      | 9787302132684 |
| XML 实用技术教程                    | 顾兵        | 9787302142867 |
| 大学计算机公共基础                     | 阮文江       | 9787302143307 |
| 大学计算机网络公共基础教程                 | 徐祥征等      | 9787302130161 |
| 多媒体技术教程——案例、训练与课程设计           | 胡伏湘等      | 9787302126201 |
| 汇编语言程序设计教程与实验                 | 徐爱芸       | 9787302143413 |
| 计算机操作系统                       | 颜彬等       | 9787302141471 |
| 计算机网络实用教程——技术基础与实践            | 刘四清等      | 9787302104513 |
| 计算机网络应用技术教程                   | 孙践知       | 9787302118893 |
| 计算机网络与 Internet 实用教程——技术基础与实践 | 徐祥征等      | 9787302106593 |
| 计算机网络与 Internet 实用教程——技术基础与实践 | 徐祥征等      | 9787302106593 |
| 实用软件工程                        | 陆惠恩       | 9787302125594 |
| 数据库及其应用系统开发(Access 2003)      | 张迎新       | 9787302128281 |
| 数据库技术与应用——SQL Server          | 刘卫国等      | 9787302143673 |
| 数据库技术与应用实践教程——SQL Server      | 严晖、刘卫国    | 9787302142317 |
| 数据库应用案例教程(Access)             | 周安宁、张新猛 等 | 9787302146056 |
| 网络技术应用教程                      | 梁维娜等      | 9787302134848 |
| 网页制作教程                        | 夏宏等       | 9787302105916 |
| 微型计算机原理及应用导教·导学·导考(第 2 版)     | 史新福等      | 9787302133995 |





● 本书是《数据库原理与DB2应用教程》的配套教材，全书共分为七个实验，内容安排由易到难。

● 可作为计算机、软件工程、电子商务、信息管理  
等本科生数据库实验课程的教材，也可作为参加  
IBM DB2专业认证考试的辅导教材。

ISBN 978-7-302-21221-8



9 787302 212218 &gt;

定价：16.00元