

高等学校信息管理与信息系统专业系列实验教材

多维数据分析

原理与应用实验教程

姚家奕 等编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY <http://www.phei.com.cn>

高等学校信息管理与信息系统专业系列实验教材

- ▶ 数据库应用基础实验教程
- ▶ 网络数据库实验教程
- ▶ 多维数据分析原理与应用实验教程
- ▶ Visual Basic 6.0 程序设计实验教程
- ▶ 网站开发技术实验教程
- ▶ 数据结构(C语言)实验教程
- ▶ 网站开发实践实验教程
- ▶ 信息系统开发实践实验教程
- ▶ 决策支持与专家系统实验教程
- ▶ 电子商务系统分析与设计实验教程
- ▶ 电子政务系统模拟实验教程
- ▶ ERP系统模拟实验教程
- ▶ 网络支付与结算模拟实验教程



策划编辑: 刘宪兰
责任编辑: 张燕虹
封面设计: 张 翌

本书贴有激光防伪标志, 凡没有防伪标志者, 属盗版图书。



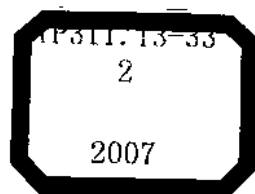
ISBN 978-7-121-04379-6



9 787121 043796 >

定价: 18.00 元

2007



高等学校信息管理与信息系统专业系列实验教材

多维数据分析原理与应用 实验教程

姚家奕 等编著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

本实验教程主要针对高等学校信息管理与信息系统专业和计算机专业的数据仓库课程的实验教学而编写。全书以数据仓库和 OLAP(联机分析处理)理论为基础,以 SQL Server 2000 Analysis Services 为实验工具,以一个人寿保险公司的数据仓库系统为背景,通过一个完整的案例,系统全面地介绍了数据仓库系统的分析、设计、实施、管理与维护的过程。

本书既可以作为高等学校信息管理与信息系统专业本科生或硕士生的教材,也可以作为从事数据仓库建设和研究人员的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

多维数据分析原理与应用实验教程/姚家奕等编著. —北京:电子工业出版社,2007.6

(高等学校信息管理与信息系统专业系列实验教材)

ISBN 978-7-121-04379-6

I. 多… II. 姚… III. 数据库系统—系统分析—高等学校—教材 IV. TP311.13

中国版本图书馆 CIP 数据核字(2007)第 067560 号

策划编辑:刘宪兰

责任编辑:张燕虹

印 刷:北京机工印刷厂

装 订:三河市鹏成印业有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:720 × 1000 1/16 印张:11 字数:197 千字

印 次:2007 年 6 月第 1 次印刷

印 数:4000 册 定价:18.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

高等学校信息管理与信息系统

专业系列实验教材

专家顾问委员会 (按姓氏笔画排序)

王金华	北京交通大学	教授
甘仞初	北京理工大学	教授
阮秋琦	北京交通大学	教授
关忠良	北京交通大学	教授
张润彤	北京交通大学	教授
陈禹	中国人民大学	教授
陈建斌	北京联合大学	副教授
屈波	北京交通大学	研究员
武刚	北京林业大学	教授
侯炳辉	清华大学	教授
高学东	北京科技大学	教授
贾素玲	北京航空航天大学	教授
黄磊	北京交通大学	教授
薛华成	复旦大学	教授

高等学校信息管理与信息系统

专业系列实验教材

编委会名单

编委会主席：李学伟（北京交通大学 副校长 教授）

编委会主编：张真继、姚家奕、刘世峰

编委会委员（按姓氏笔画排序）：王馨迪、王英、韦桂丽、刘红璐、刘世峰、陈学东、杜晖、邵丽萍、孟婕、苟娟琼、柯新生、张真继、姚家奕、宫小全、常丹

总序

从 20 世纪 80 年代开始,为了适应信息技术的迅猛发展和企业管理现代化的需要,我国一些高等学校开始设立信息管理类专业,旨在培养“既懂经营管理,又懂信息技术的复合型人才”。经过多年的发展和变化,原国家教委将经济信息管理、图书情报、管理信息系统等名称不同,但实质相似的专业统一为“信息管理与信息系统”专业。

“信息管理与信息系统”专业是一门应用之学、致用之学,实践性极强,涉及管理科学、经济学、数学、信息技术等多门类知识的专业,它的出现是多学科交叉综合发展的结果。它以管理为基础,以信息技术为手段,以实现管理现代化为总目标,力求将技术、经济、管理融为一体,以培养复合型现代管理人才。围绕上述目标,国内外各高校信息管理类专业一直将实验环节放在教学的首位。实践证明,只有紧紧围绕实验环节展开教学活动,才能真正培养出能实干、肯干、会干并有良好发展潜力的高素质人才;而如何将实验教学资源系统化、教师教学过程制度化、教学方法生动化是目前各高校“信息管理与信息系统”专业面临的普遍问题。

北京交通大学的“信息管理与信息系统”专业设立于 1986 年,其时正处于中国铁路信息化发展的大背景,北京交通大学“信息管理与信息系统”专业的教师一方面直接参与了铁路信息化建设课题的研究,另一方面则有针对性地设计了一些实验教学环节,加强对学生的科研素质、创新精神和动手能力的训练。在培养方案设计方面,除信息系统开发实践、商用 ERP 系统实验、网站开发实践、多维数据分析等专门的实验课外,所有理论课程均包含 8~16 学时的实验学时,从而可以将“实践能力培养”融入到整个教学过程,并无处不在。

但近年来,北京交通大学“信息管理与信息系统”专业在实践教学过程中也遇到了中国国内许多高校普遍遇到的问题,如实验资源分散于各个教师手中,共享程度不高;实验教学过程的规范化程度不高,缺乏系统性的实验教材,影响了学生的学习与实验效果。针对上述问题,2003 年,北京交通大学启动了“信息管理与信息系统专业实践(实验)教学资源整合”教改课题研究,以解决实验资源系统化、教学过程制度化、教学方法生动化的问题。整个课题运作经历了教学理念研究、教

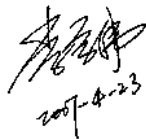
学体系设计、校内讲义编写、教学实践、学生反馈、讲义修订、校内外专家评审等若干环节,并在课题研究的基础上初步形成了本套系列实验教材。

本套系列实验教材首期计划出版 13 本,这 13 本教材内容大体分为三类实验:一是基础型实验,包括《数据库应用基础实验教程》、《网络数据库实验教程》、《多维数据分析原理与应用实验教程》、《Visual Basic 6.0 程序设计实验教程》、《网站开发技术实验教程》、《数据结构(C 语言)实验教程》6 本教材,旨在夯实学生对数据库和开发工具的掌握基础;二是设计型实验,包括《信息系统开发实践实验教程》、《决策支持与专家系统实验教程》、《网站开发实践实验教程》和《电子商务系统分析与设计实验教程》4 本教材,旨在加强学生对 B/S、C/S 等不同模式的信息系统的设计能力;三是综合型实验,包括《ERP 系统模拟实验教程》、《电子政务系统模拟实验教程》和《网络支付与结算模拟实验教程》3 本,旨在帮助学生促进对企业管理、商务管理等各类知识与信息系统知识的融合,提高学生的系统应用能力,并加强学生对所学习知识的感性体验。

实验教材建设是一项复杂、艰巨的系统工程,北京交通大学信息管理系在组织编写这套系列实验教材的过程中,得到了国内信息管理领域许多著名专家和学者的热情指导和鼎力帮助,他们为这套系列实验教材的整体设计和编写提出了很多非常好的建议,在此对他们表示衷心的感谢!希望有更多的同行为实验教材的建设提出宝贵的意见,以共同为建设好中国的信息管理类专业、培养高素质人才做出贡献。

电子工业出版社为这套系列实验教材的出版投入了大量的人力和物力,对参与这套系列实验教材出版工作的领导和编辑们表示由衷的感谢。

北京交通大学副校长
信息管理专业博士生导师、教授



2007-8-23

前言

随着数据仓库技术和产品的不断成熟和发展,数据仓库已成为高等院校信息管理与信息系统专业、计算机专业的主干专业课程,这是一门理论与技术密切结合,同时又非常强调实践与应用的课程。本书是一本基于数据仓库课程的特点,为高等学校信息管理与信息系统专业、计算机专业的学生,或者希望了解和掌握数据仓库系统开发技术的专业技术人员而编写的实验教材,是对数据仓库技术与应用课程理论教学的有力支撑和补充。

素质教育是高等学校教学改革的主要目标和任务,实验教学是素质教育的重要手段。一本好的实验教材,其内容上不仅要覆盖理论教学的所有知识点,还要对这些知识点进行有机的连接和串通,使得实验教材构成一个完整的体系。另外,更加重要的一点是,实验的内容要与实际相结合,虚拟的实验数据和背景会大大降低实验的效果和学生的兴趣。

作为长期从事信息管理与信息系统专业数据仓库的理论教学工作,并积累了较丰富的与数据仓库技术相关的科研和项目开发经验的教师,通过长期的课堂教学和实验教学,发现学生学习数据仓库技术的主要障碍和瓶颈并不是对技术本身的了解和掌握,而是如何运用这些技术解决实际问题,这正是由数据仓库系统的特点造成的。数据仓库技术主要用于构建面向企业或组织的管理者、决策者的数据分析与决策支持系统,很难用虚拟的数据和处理方式来模拟和仿真这类系统的实际需求。脱离实际需求的学习和实验,会使学生无法真正掌握数据仓库的核心技术和方法,也体验不到数据仓库技术的魅力所在。基于这种特点,我们结合实际数据仓库系统的开发背景,参考和借鉴了同类实验教材的宝贵经验,重新设计了数据仓库的实验教学环节和内容,体现了以需求为驱动的实验教学理念。本实验教材以作者主持的某人寿保险公司数据仓库系统为背景案例,将该系统的主题分析、立方体设计、数据访问、数据仓库管理等技术内容,由浅入深,由简到繁,有机地渗入到实验的每个环节和步骤。通过这样的实验,学生提高的不仅是对相关技术的掌握水平,更重要的是提高了对数据仓库系统的整体认识,增强了学生参与开发数据仓库系统的实际能力。

本书共 8 章。第 1 章“概论”介绍了本实验教材的主要内容、设计思路、实验环境配置及实验报告的要求等。第 2 章“源数据分析与 Analysis Services 配置”介绍了源数据的表结构、表之间的关联关系, Analysis Services(分析服务)的概念和功能, Analysis Services 数据库和数据源的设置方法等。第 3 章“创建多维数据集”介绍了建立各种维度的建立过程, 重点说明了如何在这些现有维度的基础上创建多维数据集, 并详细讲解了对维度和多维数据集进行编辑的方法等知识。第 4 章“复杂多维数据集的创建”重点介绍了父子维度、计算成员、成员属性和虚拟维度, 以及它们的建立方法与应用过程。第 5 章“多维数据集的分区、虚拟多维数据集”从多维数据集的处理效率方面考虑, 介绍了多维数据集的 3 种不同处理方法及分区与合并的概念、原理、实现方法、意义。另外, 在简单多维数据集的基础上, 进一步介绍了虚拟多维数据集的概念、建立、编辑和功能等。第 6 章“MDX 语言应用”介绍了数据仓库中的 MDX 语言, 主要包括 MDX 函数的使用、MDX 查询语句的使用等。第 7 章“建立 OLAP 的 Web 动态访问”则从直接使用 Analysis Manager(分析管理器)进行数据浏览、使用 Excel 作为前端分析工具及使用 OLAP 的 Web 动态数据透视方面, 介绍了数据的共享使用技术, 其中 Web 动态访问技术是本章的重点。第 8 章“系统维护与管理”主要介绍了数据库角色管理、数据仓库的备份和恢复、多维数据集的优化及系统环境的高级配置等内容。

本书由姚家奕主编。在本书的编写过程中得到了张润彤教授、张真继教授、柯新生副教授、王馨迪副教授和陈学东副教授的大力支持和帮助, 他们为本书的编写提出了很多建设性的意见, 在此一并表示感谢! 同时还要感谢为本书的实验准备、实验测试和文档编辑投入大量时间和精力的王洪莹、王巍、王一清、孔淑慧和石丹丹。

在本书的构思和编写过程中, 作者尽可能将数据仓库的实用技术和方法纳入到实验环节中, 但由于水平和能力所限, 书中还会有许多不足之处, 恳请同行、专家和读者提出宝贵意见。

读者可从华信教育资源网(www.huaxin.edu.cn 或 www.hxedu.com.cn) 下载本书电子教学参考资料包。

编著者

2007 年 4 月

目录

第 1 章 概论	(1)
1.1 实验设计说明	(2)
1.2 实验内容及要求	(3)
1.2.1 实验内容	(4)
1.2.2 实验要求	(5)
1.2.3 扩展实验	(6)
1.3 实验环境配置	(7)
1.3.1 源数据的分析与准备	(7)
1.3.2 SQL Server 2000 的安装	(7)
1.3.3 源数据库的还原	(10)
1.4 实验报告	(13)
1.4.1 实验概述	(13)
1.4.2 实验内容	(14)
1.4.3 小结	(15)
本章小结	(15)
第 2 章 源数据分析与 Analysis Services 配置	(17)
2.1 实验预备知识	(18)
2.1.1 数据库及数据库管理系统的相关概念	(18)
2.1.2 数据库管理系统的功能	(18)
2.1.3 “Analysis Services”的概念和功能	(19)
2.2 实验目的	(19)
2.3 实验要求	(20)
2.4 实验内容与步骤	(20)
2.4.1 数据分析的需求和目标	(20)
2.4.2 数据库的结构和内容	(21)
2.4.3 注册 Analysis Servers	(22)

2.4.4 设置数据库	(22)
2.4.5 设置数据源	(23)
2.5 扩展实验	(24)
本章小结	(26)
第3章 创建多维数据集	(27)
3.1 实验预备知识	(28)
3.1.1 共享维度	(28)
3.1.2 多维数据集	(28)
3.2 实验目的	(28)
3.3 实验要求	(28)
3.4 实验内容与步骤	(29)
3.4.1 建立时间维度	(29)
3.4.2 建立单一层次的标准维度	(32)
3.4.3 建立多层次标准维度	(33)
3.4.4 建立多重关系型标准维度	(33)
3.4.5 维度的编辑	(36)
3.4.6 多维数据集的编辑	(38)
3.5 扩展试验	(47)
本章小结	(48)
第4章 复杂多维数据集的创建	(49)
4.1 实验预备知识	(50)
4.1.1 父子维度	(50)
4.1.2 计算成员	(50)
4.1.3 成员属性	(51)
4.1.4 虚拟维度	(51)
4.2 实验目的	(51)
4.3 实验要求	(52)
4.4 实验内容与步骤	(52)
4.4.1 父子维度的建立	(52)
4.4.2 计算成员的建立	(56)
4.4.3 成员属性的建立	(58)
4.4.4 虚拟维度的建立	(60)
4.5 扩展试验	(65)
本章小结	(66)

第 5 章 多维数据集的分区、虚拟多维数据集	(67)
5.1 实验预备知识	(68)
5.1.1 多维数据集分区	(68)
5.1.2 多维数据集分区的合并	(68)
5.1.3 虚拟多维数据集	(69)
5.2 实验目的	(69)
5.3 实验要求	(69)
5.4 实验内容与步骤	(69)
5.4.1 多维数据集分区的建立	(70)
5.4.2 多维数据集分区的应用	(74)
5.4.3 多维数据集分区的合并	(75)
5.4.4 虚拟多维数据集的建立	(76)
5.5 扩展实验	(79)
本章小结	(80)
第 6 章 MDX 语言应用	(81)
6.1 实验预备知识	(82)
6.2 实验目的	(82)
6.3 实验要求	(82)
6.4 实验内容与步骤	(83)
6.4.1 使用“MDX 示例应用程序”建立连接	(83)
6.4.2 使用成员表达式	(84)
6.4.3 坐标轴与子多维数据集	(87)
6.4.4 注释表达式与空数据集的处理	(91)
6.4.5 建立计算成员和命名集	(93)
6.5 扩展实验	(99)
本章小结	(101)
第 7 章 建立 OLAP 的 Web 动态访问	(103)
7.1 实验预备知识	(104)
7.2 实验目的	(104)
7.3 实验要求	(104)
7.4 实验内容与步骤	(105)
7.4.1 使用“Analysis Manager”展示数据	(105)
7.4.2 使用 Web 进行数据展示	(108)
7.5 扩展实验	(117)

本章小结	(118)
第 8 章 系统维护与管理	(119)
8.1 实验预备知识	(120)
8.2 实验目的	(120)
8.3 实验要求	(120)
8.4 实验内容与步骤	(121)
8.4.1 数据库角色管理	(121)
8.4.2 多维数据集角色管理	(128)
8.4.3 数据库的存档与恢复	(130)
8.4.4 多维数据集的优化	(134)
8.4.5 系统环境的设置	(138)
8.5 扩展实验	(140)
本章小结	(141)
附录 A 常用 MDX 函数介绍	(142)
附录 B “人寿实验.CAB”数据库文档的还原	(155)
附录 C 实验报告格式及说明	(157)
参考文献	(161)



第1章

概 论

1.1 实验设计说明

随着计算机技术的飞速发展和企业界不断提出新的需求,数据仓库技术应运而生。

传统的数据库技术是单一的数据资源,即以数据库为中心,进行从事务处理、批处理到决策分析等各种类型的数据处理工作。近年来,随着计算机的应用发展,网络计算开始向两个不同的方向拓展:一是广度计算,二是深度计算。广度计算的含义是尽量扩大计算机的应用范围,同时实现广泛的数据交流,互联网就是广度计算的特征;另一方面,人们对以往计算机的简单数据操作提出了更高的要求,希望计算机能够更多地参与数据分析与决策的制定等领域。特别是数据库处理可以大致地划分为两大类:操作型处理和分析型处理(或信息型处理)。这种分离,划清了数据处理的分析型环境与操作型环境之间的界限,从而由原来的以单一数据库为中心的数据环境发展为一种新环境:体系化环境。

数据库系统作为数据管理手段,从它的诞生开始,就主要用于事务处理。经过数十年的发展,在这些数据库中已经保存了大量的日常业务数据。传统的业务系统一般是直接建立在这种事务处理环境上的。随着技术的进步,人们试图让计算机承担更多的工作,而数据库技术也一直力图使自己能胜任从事务处理、批处理到分析处理的各种类型的信息处理任务。后来,人们逐渐认识到,在目前的计算机处理能力上,根本无法实现这种功能,而另一方面,事物处理和分析处理具有极不相同的性质,直接使用事务处理环境来支持决策是行不通的。

数据仓库技术以改进后的数据库技术作为存储数据和管理资源的基本手段,以统计分析技术作为分析数据和提取信息的有效方法,通过人工智能、神经网络、知识推理等数据挖掘方法来发现数据背后隐藏的规律,从而实现从“数据→信息→知识”的过程,为企业的管理阶层提供各种层次的决策支持。

SQL Server 2000 Enterprise Manager 是企业级关系数据库管理软件,是记录执行企业日常操作所需数据的高效系统。Analysis Services(分析服务)是在 SQL 2000 架构中,用来提供联机分析处理(OLAP)与数据挖掘(Data Mining)的中介层服务器。Analysis Services 为前一版本 SQL 7.0 中“OLAP Service”的升级功能。

Analysis Services 会预先计算 SQL 2000 数据仓库(Data Warehouse)的汇总数据,并以事先定义的方式(也就是“维度”),组织并存储到多维数据集中,以便快速

地响应决策分析者复杂而多维的数据查询。Analysis Services 也可以让用户从多维数据集或关系型数据库中建立数据“挖掘模型”,以进行数据的探索与预测。

本实验以数据仓库和联机分析处理理论、技术为基础,以 SQL Server 2000 Analysis Services 为工具,对一个寿险公司的团险部客户关系数据进行模拟分析,由浅入深地带领用户建立分析多维数据集,说明如何通过前端的分析工具来获取分析结果。通过读者上机实践,全面地复习、掌握实验内容体系中的各知识点,并通过综合实践系统将所学知识串接起来,让用户可以用最少的时间处理决策分析所需的资料,并从中得到各种不同的分析结果,以作为经营决策的最佳参考。

本实验教程是面向高等学校信息管理与信息系统专业学生学习及实践多维数据集的配套指导教材,也为其他学习爱好者提供了一个多维数据分析的学习空间,同时也适合作为多维数据集初学者的应用实践辅导书。为促进初学者综合系统分析设计的能力,本实验教程采用了“需求驱动式”教学实践模式,即学习每一阶段知识后针对知识体系给出相应的拓展实验,使读者及时地复习巩固。

实验内容设计的出发点是通过一个寿险公司“团险部客户关系管理系统”开发实例,将 SQL Server 2000 Analysis Services 开发工具的主要知识点贯穿起来,为读者提供一个较为全面的管理类应用系统实践环境和 Analysis Services 学习、演练、规范的空间,以充分体会知识点间的相互衔接和综合应用。

本实验教程以一具体的小型客户关系管理系统作为实验的内容,目的如下:

(1) 便于读者全面理解 SQL Server 2000 Analysis Services 开发工具的内容体系,并系统地掌握系统设计思路与方法。

(2) 力图使读者在完成实验后,可以熟练地掌握 SQL Server 2000 Analysis Services 系统设计思想和规范、方法和技巧,掌握各知识点在实际系统开发中的应用,并能独立创建应用系统。

(3) 弥补初学者缺乏实际系统开发经验,面对实际应用问题不知如何入手、如何组织的弊端,以循序渐进的过程分阶段、分步骤地逐步完成整个系统,培养独立分析问题、综合问题、解决问题的能力。

(4) 通过应用系统的实践,使多维数据集初学者在认真、独立地完成本书的实践任务后,可以胜任经济类、管理类应用系统的分析设计工作。

1.2 实验内容及要求

本节将阐述本教程实验的主要内容及相应的实验要求,以便读者对该实验有

一个整体的了解,从而在后续的实际设计过程中有一个完整的思路。

1.2.1 实验内容

本实验主要利用数据仓库和 OLAP(联机分析处理)理论与技术,通过 SQL Server 2000 Analysis Services 对一个寿险公司的团险部客户关系数据进行分析,为公司团险的日常管理提供综合、及时、准确的分析型数据,为公司各级管理者提供可靠的团险决策分析工具与支持数据,以达到通过各种不同角度进行数据分析,并帮助寿险公司管理者进行正确决策的目的。

本实验依据寿险公司团体保险业务的既往或实时产生的业务数据、客户资料及从多渠道收集的商业信息,通过本系统进行微观和宏观两个层面、多角度的分析和整理,达到了解寿险公司团体客户和业务发展现状,预测业务发展趋势,最终为团体业务营运和客户管理的远期发展规划服务的目标。

目前寿险公司团险业务数据资料有客户资料、业务员资料、保单资料、险种资料及销售渠道资料等,但是这些资料都很松散,无法通过这些现成的资料一目了然地了解寿险公司客户关系的总体情况和详细状况。

寿险公司的管理决策者需要从企业零散的资料信息中获得以下的统计、分析的数据信息:

- (1) 考察不同产品的历年收入情况,如保费、管理费。
- (2) 考察不同产品的历年支出情况,如退保、赔款。
- (3) 考察渠道、销售模式(个/团/卡折)方面的销售情况。
- (4) 考察各类、各个产品历年赔付率状况,并对高赔付率的产品提出警示标志。
- (5) 分析各营业单位销售业绩情况、实际贡献情况、业务发展规律、产品分布情况。
- (6) 分析各销售员的销售业绩情况及业务发展规律、销售偏好情况、业务质量情况。

决策者通过了解以上的一些信息,根据分析统计的数据,从日常纷乱的资料中,提取有用的决策信息,从而可以针对长、短期险业务数据,从行业、业务机构、产品等多种角度出发分析收支流量情况,进而改进寿险公司的经营策略,使其可以满足客户的需求,增强自身的竞争力。

模拟决策者希望分析的对象,也就是所谓的“度量值”,最基本的应该是希望

统计“预收保费”、“新单保费”、“续期保费”、“退还保费”、“实付赔款”、“实付退保金”、“实付生存金”、“实付红利”,等等;而所使用的“分析角度”(维度)可能为行业、单位性质、保单状态、机构-业务员、承保形式、收付费时间、险种情况、渠道分布等。

本实验的主要内容有:

- (1) 设置分析服务器、数据仓库的数据源类型和设置。
- (2) 共享维度的建立、多维数据集的建立和编辑。
- (3) 父子维度、计算成员、成员属性、虚拟维度的建立和应用。
- (4) 多维数据集分区的建立和编辑、虚拟多维数据集的建立。
- (5) MDX 语言应用。
- (6) 建立 OLAP 的 Web 动态访问,使用 Analysis Manager 展示数据;使用 Web 进行数据透视。
- (7) 用于数据库角色管理的主要操作,进行数据库的存档与恢复,进行多维数据集的优化工作,进行系统环境的常规配置。

1.2.2 实验要求

(1) 系统环境配置:设置新的 Analysis Servers(分析服务器),设置新的数据库,设置数据源链接。

(2) 创建多维数据集:创建时间维度、单一层次的标准维度、多层次标准维度、多重关系型标准维度,掌握共享维度建立的方法和要求,理解多维视图使用维和度量进行数据分析的方法,学会根据不同的应用选择相应的多维数据存储模型。

(3) 复杂多维数据集的创建:熟练掌握父子维度的建立方法和特点,理解和掌握计算成员在维度指标中的应用,了解成员属性的意义,掌握成员属性的建立方法,掌握通过成员属性来建立虚拟维度的方法。

(4) 多维数据集分区、虚拟多维数据集:掌握多维数据集分区的建立和编辑的方法,掌握虚拟多维数据集的建立方法。

(5) MDX 语言应用:使用“MDX 示例应用程序”建立连接,使用成员表达式、坐标轴与子多维度数据集,注释表达式与空数据集的处理,建立计算成员和命名集。

(6) 建立 OLAP 的 Web 动态访问:使用 Analysis Servers 展示数据,使用 Web 进行数据展示。

(7) 系统维护与管理:管理数据库角色,包括数据库角色的添加、修改和删除等。

1.2.3 扩展实验

读者完成每章中的基本实验环节后,为了进一步巩固所学的知识和技术,本书为读者设置了扩展实验环节,读者可以根据学时安排、难易程度自行决定完成扩展实验中的部分或全部内容。全书的扩展实验分为如下几个方面:

(1) 通过事实数据表“dim_保单类型”、“dim_险种类别分类”、“DIM_中间表”、“DIM_法人客户表”等,建立共享维度“保单类型”、“生效时间”、“险种_险种类型分类”、“开户年.法人客户”等标准维度;对已有的维度进行简单的编辑处理。

(2) 创建“法人客户价值分析保单数量”和“法人客户价值—虚拟维度”多维数据集,实现对法人客户价值的分析。

(3) 在多维数据集“法人客户价值分析保单数量”的基础上,创建虚拟多维数据集“业务经理专用”,屏蔽一些机密数据,实现多维数据集的安全设置。

(4) 对多维数据集“法人客户价值分析保单数量”进行编辑,包括维度的添加、度量值的建立、维度“成员属性”的建立等;可以对所编辑的多维数据集进行简单的存储设计,最后完成对多维数据集的成功处理。

(5) 可以通过“DIM_业务员等级关系”表,实现共享父子维度“管理层次”和私有父子维度“业务员等级”的建立,将该私有父子维度加入“法人客户价值分析保单数量”多维数据集。

(6) 在“法人客户价值分析保单数量”多维数据集中,新建度量值计算成员“客户数量”和父维度为“单位性质”的计算成员“非公有制”。

(7) 对多维数据集“法人客户价值分析保单数量”进行分区设计、编辑、存储设计和处理,然后对已有的分区进行合并,来了解分区的原理和意义。

(8) 练习使用 MDX 语言进行多维数据查询,包括数据的简单查询、嵌套查询和 MDX 常用函数(成员函数、集合函数、维度函数和数值函数等)的使用。

(9) 练习使用“Analysis Servers”来对多维数据集中的数据进行浏览,并且可以实现使用 Web 进行数据集的动态访问,从而实现数据资源的共享。

(10) 掌握系统的基本维护和管理技术,包括使用权限的设置和角色管理,进行数据库的备份与恢复,同时能够了解多维数据集进行优化的基本操作以及系统环境的一些常规设置。

1.3 实验环境配置

1.3.1 源数据的分析与准备

了解数据库表的基本结构和内容、表与表之间的联系以及数据分析的主要目标和任务。

关于源数据的分析与准备详细内容请读者参阅第2章中的2.4.1节“数据分析的需求和目标”以及2.4.2节“数据库的结构和内容。”

1.3.2 SQL Server 2000 的安装

1. 安装环境

1) 硬件要求

SQL Server 2000 对硬件的最低要求如下:

(1) 处理器: Intel 兼容处理器, 奔腾 166 以上。

(2) 内存: 安装 SQL Server 2000 一般需要 64MB 以上的内存, 如果机器上内存较少, 则应关闭其他应用程序及所有不必要的服务程序。

(3) 硬盘: SQL Server 2000 有 Data Server(数据库服务器)、Analysis Services(分析服务)和 English Query(英语查询)三个部分, 其中数据库服务器的完全安装需要 180MB 的硬盘空间, 典型安装需要 170MB 磁盘空间, 最小安装需要 65MB 磁盘空间, 如果只安装客户端工具则需要 90MB 磁盘空间; 安装 Analysis Services 和 English Query 则分别需要 50MB 和 12MB 的硬盘空间。以上提到的硬盘空间要求仅是安装基本的 SQL Server 2000 程序及管理工具文件所需的, 不包括用户数据库所需的。安装 SQL Server 2000 的机器还应有足够的空间来存放数据。

SQL Server 2000 没有专有的硬件兼容性列表, 它可以运行在满足以上提到的最低系统要求的且与 Windows NT 兼容的硬件上。

2) 软件要求

(1) 操作系统需求: 该教材中所做的实验用的是 SQL Server 2000 Enterprise Edition 版本, 需要的操作系统为 Windows 2000 Server、Windows 2000 Advanced Server 以及其他更高版本的系统。

(2) Internet 软件需求: 任何 SQL Server 2000 的安装都需要 Microsoft Internet Explorer 5.0 或更高版本。

(3) 网络软件需求:SQL Server 2000 可以使用 Microsoft Windows NT、Windows 2000、Windows 98 或 Windows 95 安装好的通信协议。一般来说,在安装 SQL Server 2000 之前不需要其他额外的网络协议。

2) 安装步骤

将 SQL Server 2000 的光盘插入 CD-ROM 驱动器之后,SQL Server 的安装程序会自动启动。如果该光盘不自动运行,请双击该光盘根目录中的 Autorun.exe 文件,打开安装选项窗口。

第1步 选择“安装 SQL Server 2000 组件”选项,再选择“安装数据库服务器”。出现安装向导后,单击“下一步”按钮,出现“计算机名”对话框。“本地计算机”是默认选项,其名称就显示在该对话框中,其默认的名称为“BJTU-2”(在操作时采用默认的计算机名称,但要注意在后面的实验中用到的服务器名称也应当是其默认的名称),单击“下一步”按钮。

第2步 在“安装选择”对话框中,同样采用其默认项“创建新的 SQL Server 实例,或安装客户端工具”,单击“下一步”按钮;在出现的对话框中自定义用户名、公司名,在经过软件许可协议后,出现“安装定义”对话框。同样,采用其默认项“服务器和客户端工具”,单击“下一步”按钮。

第3步 选择默认的实例名称,这时 SQL Server 的实例名称将与 Windows 2000 服务器的名称相同。SQL Server 2000 可以在同一台服务器上安装多个实例,即用户可以重复安装几次,如果你的计算机上已经安装了数据库实例,默认实例可能不可选择,这时就需要选择不同的实例名称了。实例名称会出现在各种 SQL Server 和系统工具的用户界面中,名称越短,越容易读取。实例名称不能是“Default”等 SQL Server 的保留关键字,单击“下一步”按钮。

第4步 在“安装类型”对话框中,可以设定多个选项,如设定安装组件的数量以及安装的路径等,可根据实际需要进行选择。一般选择“典型安装”。在“服务账户”对话框中,选择“使用本地系统账户”。(不建议选择“使用域用户账户”,以避免以后如果修改 Windows 的登录用户名和密码,导致 SQL Server 2000 不能正常启动,另外,域用户账户管理相对复杂些)单击“下一步”按钮。

第5步 在“身份验证模式”对话框中,选择“Windows 身份验证模式”,单击“下一步”按钮开始复制文件,再单击“下一步”按钮。

第6步 在“选择许可模式”对话框中,根据你购买的 SQL Server 2000 软件的类型和数量进行选择。“每客户”表示同一时间最多允许的连接数,“处理器许可证”表示该

服务器最多能安装多少个 CPU。例如,可选择“每客户”“5”个设备,如图 1.1 所示。

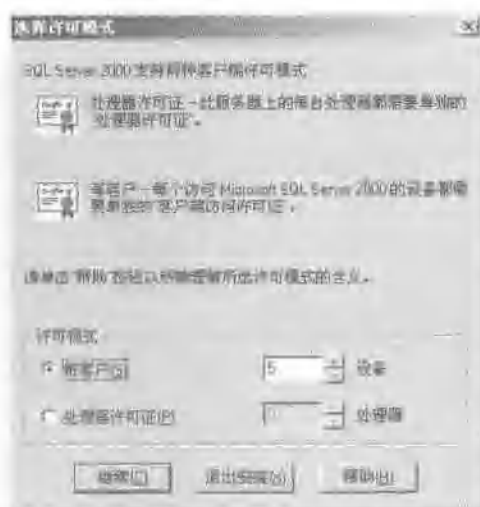


图 1.1 “选择许可模式”对话框

第 7 步 完成所有设定后,单击“继续”按钮,安装程序开始向硬盘复制必要的文件,开始正式安装,大约 3min 后,安装完成。然后,回到“安装 SQL Server 2000 组件”对话框,继续安装 Analysis Services,其步骤与安装数据库服务器类似,这里不再重述。注意:安装完成后应重新启动计算机。然后,查看 SQL Server 2000 的启动情况。成功安装 SQL Server 2000 后,在 SQL Server 正常启动后,计算机桌面右下角出现的 SQL Server 服务监视图标显示为一个带绿色三角的服务启动标志。若该绿色图标变成了红色的方块,则表明 SQL Server 已停止了服务。可通过“SQL Server 服务管理器”程序来启动 SQL Server。双击该图标,将出现如图 1.2 所示的“SQL Server 服务管理器”对话框,可通过该程序停止、启动 SQL Server 的后台服务。

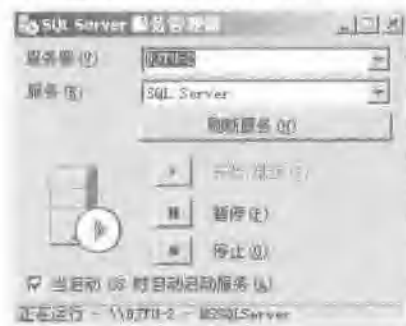


图 1.2 “SQL Server 服务管理器”对话框

第8步 如果在计算机桌面右下角没有出现 SQL Server 服务监视图标,依次单击“开始”→“程序”,即可看到 Microsoft SQL Server 2000 的程序组件,如图 1.3 所示。



图 1.3 Microsoft SQL Server 2000 的程序组件

单击级联菜单中的“服务管理器”选项,可以启动 SQL Server 服务。

1.3.3 源数据库的还原

本实验教程的数据源要从华信教育资源网(www.huaxin.edu.cn 或 www.hxedu.com.cn)下载,文件名称是“dorm_simple”。文件下载到本地的计算机后,按下列步骤完成源数据库的还原工作。

第1步 单击“开始”→“程序”→“Microsoft SQL Server”→“企业管理器”,如图 1.4 所示。

第2步 在出现的“SQL Server Enterprise Manager”窗口中,单击“控制台根目录”下“Microsoft SQL Servers”图标前面的“+”号,出现“SQL Server 组”。单击“SQL Server 组”图标前面的“+”号,可以看到服务器名称。单击打开服务器下的“数据库”文件夹,可以看到该服务器下的所有数据库。在数据库上单击右键,选择“所有任务”中的“还原数据库”,打开“还原数据库”对话框,如图 1.5 所示。

第3步 在“还原数据库”对话框的“常规”选项中,选择“从设备”还原。然



图 1.4 选择“企业管理器”

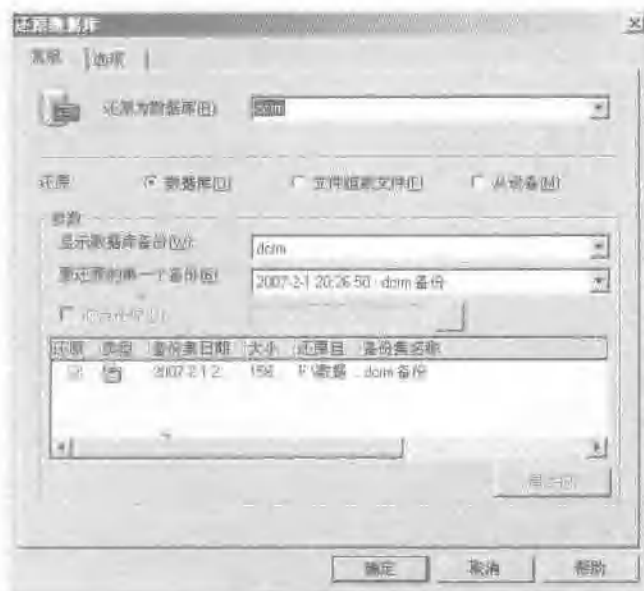


图 1.5 “还原数据库”对话框

后单击“选择设备”按钮,出现“选择还原设备”对话框。单击对话框中设备名旁边的“添加”按钮,出现“选择还原目的”对话框,如图 1.6 所示。

第 4 步 在“选择还原目的”对话框中,单击“文件名”框右边的“...”按钮,选择数据备份的位置,即文件名为“dcm_simple”的文件所在的位置。单击选中该文件名,单击“确定”按钮;再单击“选择还原目的”对话框中的“确定”按钮,单击

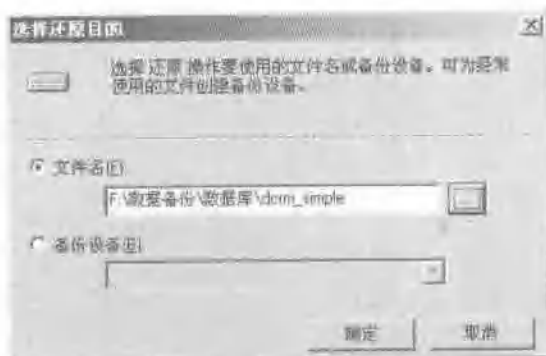


图 1.6 “选择还原目的”对话框

“选择还原设备”对话框中的“确定”按钮,回到“还原数据库”对话框。此时,在“常规”选项卡的“参数”区的“设备”文本框中,显示数据库文件名为“derm_simple”的相关路径,如图 1.7 所示。

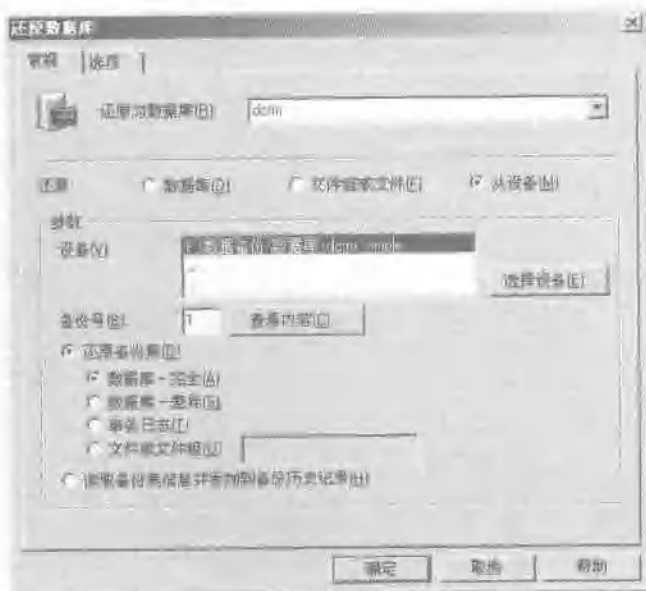


图 1.7 “常规”选项卡

第 5 步 在“还原数据库”对话框的“选项”选项卡中,选择“在现有数据库上强制还原”,修改“移至物理文件名”下的文件的物理路径。把文件移至“F:\数据备份\数据库\data”下,如图 1.8 所示。

第 6 步 单击“确定”按钮,系统进行数据库的还原工作,当“derm_simple”数据库还原成“derm”数据库后,出现如图 1.9 所示的“SQL Server 企业管理器”对话

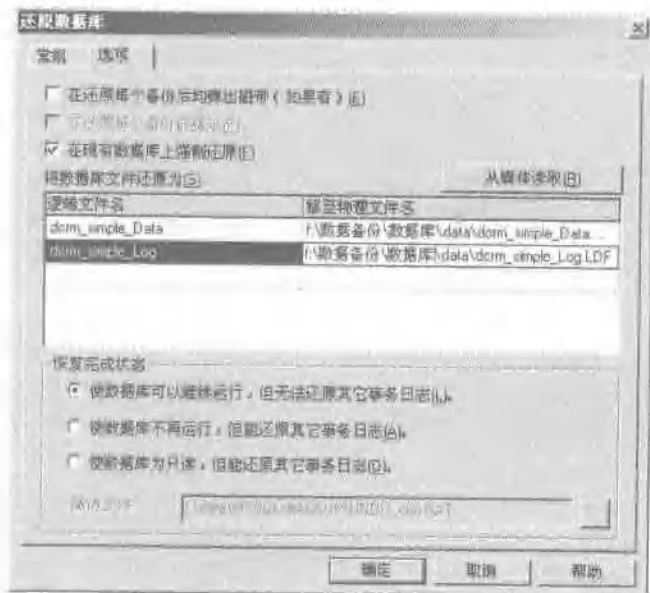


图 1.8 “选项”选项卡

框,单击“确定”按钮。

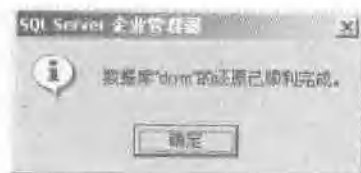


图 1.9 “SQL Server 企业管理器”对话框

处理这一任务需要一段时间,在实际操作时,需耐心等待。

1.4 实验报告

在完成本教材每一章节的实验后,均须提交相应的实验报告,该报告应包括实验概述、实验内容和小结三方面的内容。

1.4.1 实验概述

实验概述包括实验目的及要求、实验原理、实验环境三部分。

1. 实验目的及要求

实验目的应明确,应抓住重点。实验要求是每一章节的实验所要求的准则和实验要完成的任务。

2. 实验原理

简要说明本实验项目所涉及的理论知识。

3. 实验环境

说明实验用的软硬件环境(配置)。

1.4.2 实验内容

实验内容包括实验方案设计、实验过程以及结论或结果三部分。

1. 实验方案设计

实验方案设计是实验报告极其重要的内容,概括整个实验过程。要写明依据何种原理、操作方法进行实验,要写明需要经过哪几个步骤来实现其操作。必要时,还应该画出流程图,写出设计思路和设计方法,再配以相应的文字说明。

2. 实验过程

(1) 分别写出上机完成任务的主要内容及步骤,用到的多维数据集、表和维度。

(2) 主要实验步骤的截图。

(3) MDX 语言使用实验中,要写清楚使用的函数和表达式,给出基本的定义和查询语句。

(4) 分析实际操作中遇到的问题和提出解决办法。

3. 结论或结果

根据实验实际操作过程中所观察的现象和得到的数据或结果,得出合理的结论。

1.4.3 小结

本次实验的心得体会、思考和建议如下：

- (1) 实验中所用到的主要知识点的概括和总结。
- (2) 实验中所用维度、度量值、计算成员、多维数据集等的功能和意义。
- (3) MDX 语言的使用方法、常用函数的功能和使用、函数表达式的语法和书写技巧。
- (4) 维度、度量值、计算成员、多维数据集等的建立和编辑能力。
- (5) 对数据仓库强大的数据分析功能的理解。
- (6) 对利用哪些知识、哪些操作、哪些函数区解决实际生活中的哪些领域的数据分析的思考。

本章小结

以数据仓库为基础的商业智能系统强大的功能在实际应用中能带来高利润的回报。近年来,数据仓库在证券、银行、税务、控制金融风险、保险、客户管理等众多领域中得到了越来越广泛的应用。

本章从社会的发展需要为基本出发点,以数据仓库在保险行业的应用为实例,对整个实验进行了基本设计,大体介绍了实验的基本内容和基本要求,指出了扩展实验所涉及的主要内容,并且对实验报告的书写提出了具体要求。以下章节各实验的实验报告,请按本章的要求去完成。



第2章

源数据分析与 Analysis Services 配置

本章引言

本实验教材所用到的源数据,是以表的形式存放在 SQL 2000 数据库中。每张表有特定的结构,表内数据存储在各字段中,表和表之间存在着逻辑关系。

本章主要介绍了数据库源数据表的表结构、关系数据库表之间的关系,Analysis Services 的概念和功能,Analysis Servers、数据库和数据源的设置方法等,这些都为在后续章节中建立分析型数据库做好了准备。

本章重点

- 数据库及数据库管理系统的相关概念。
- 数据库管理系统的功能。
- 数据库的操作方法。
- dcrn 数据库的表结构与表关系。
- Analysis Services 的概念与功能。
- Analysis Servers、数据库、数据源的设置。

2.1 实验预备知识

数据库技术发展到今天已经是一门非常成熟的技术。它始于 20 世纪 60 ~ 70 年代,而今无论是数据库的技术水平还是数据库的应用水平,今天和过去都不可同日而语,无论是计算机用户还是从事计算机工作的技术人员,数据库都成了大家熟悉的数据管理工具。数据管理经历了从手工管理阶段、文件管理阶段到数据库管理阶段的变迁。进入 21 世纪这个信息时代后,数据库技术已经为广大用户所接受,并获得了广泛的应用。

2.1.1 数据库及数据库管理系统的相关概念

顾名思义,数据库就是存放数据的仓库,只不过这个仓库不是一般意义上的库房,它是存储在计算机存储设备上的相互关联的数据集合。

从中文字面上来看,数据库和数据仓库似乎是同一个概念,但这种理解是不准确的。数据库对应的英文单词是 Data Base,直译是数据基地的意思;而数据仓库对应的英文单词却是 Data Warehouse。数据库和数据仓库不是同义词,数据仓库是在数据库技术的基础上发展起来的一个新的应用领域。

随着计算机技术的不断进步,数据库技术也逐步走向成熟和完善。尽管近年来数据库技术有着日新月异的发展,但数据库的最基本特征没有变,概括起来应该包括如下几个方面:

- (1) 数据库是相关联的数据的集合。
- (2) 数据库用综合的方法组织数据,保证尽可能高的访问效率。
- (3) 数据库具有较高的数据独立性。
- (4) 数据库具有安全控制机制,能够保证数据的安全、可靠。
- (5) 数据库允许并发地使用,能有效、及时地处理数据,能保证数据的一致性、完整性。

2.1.2 数据库管理系统的功能

数据库的各种功能和特性,并不是数据库中的数据固有的,它是靠管理或支持数据库的系统软件——数据库管理系统(Data Base Management System,简称 DBMS)提供的。具体来说,一个数据库管理系统应该具备如下功能。

- (1) 数据库定义功能:可以定义数据库的结构和数据库的存储结构,可以定

义数据库中数据之间的联系,可以定义数据的完整性结束条件和保证完整性的触发机制等。

(2) 数据库操纵功能:可以完成对数据库中数据的操纵,可以装入、删除、修改数据,可以重新组织数据库的存储结构,可以完成数据库的备份和恢复等操作。

(3) 数据库查询功能:可以以各种方式提供灵活的查询功能,使用户可以方便地使用数据库中的数据。

(4) 数据库控制功能:可以完成对数据库的安全性控制、完整性控制、多用户环境下的并发控制等各方面的控制。

(5) 数据库通信功能:在分布式数据库或提供网络操作功能的数据库中还必须提供数据库的通信功能。

2.1.3 “Analysis Services”的概念和功能

Analysis Services 必须安装在 Microsoft Windows 2000 Server 上或搭配有 Service Pack 5 的 Microsoft Windows NT Server 4.0 的操作系统上。严格来说,Analysis Services 包含 Analysis Servers 及其所提供的服务、管理功能,具体有管理 Analysis Servers,建立数据库与指定数据源,建立与处理多维数据集,建立与处理数据挖掘模型,指定数据存储类型选项及基于使用的优化查询效率、安全性管理、浏览数据源、共享维度、安全性角色以及其他对象等服务、管理功能。

Analysis Services 预先计算 SQL 2000 数据仓库的汇总数据,并以事先定义的方式(也就是“维度”),组织并存储到多维数据集中,以便快速地响应决策分析者复杂而多维的数据查询。

Analysis Services 可以让用户从多维数据集或关系型数据库中建立数据挖掘模型,以进行数据的探索与预测。此外,Analysis Services 还搭配提供了 PivotTable (数据透视表)的功能,它与“OLE DB 提供者(OLE DB Provider)”兼容,以便使分析的数据可提供如 Excel 的数据透视表,或可以在其他应用程序中进行数据分析,甚至是建立脱机多维数据集的分析模式。

2.2 实验目的

通过本章内容的学习,读者可以大致了解数据库表的基本内容、表与表之间的联系以及数据分析的目标功能,重点掌握对数据库服务器、数据源的基本设置方法,为在后续章节中建立分析型数据库做好准备。

2.3 实验要求

希望读者通过本章内容的学习以及自己的思考,能够深入了解案例数据库的数据结构,理解和掌握数据分析的主要目标和任务,重点掌握如何设置 Analysis Servers 的方法,根据不同数据库系统的数据源,掌握它们之间数据格式的转换方法,并能熟练使用 ODBC (Open Data Base Connectivity, 开放数据库互连) 进行数据集成,了解并完成各项目标、任务的技术路线和具体要求。

2.4 实验内容与步骤

本实验的主要内容是数据分析的需求和目标,数据库的数据结构分析, Analysis Servers、数据库和数据源的设置。

本书中的案例是模拟某寿险公司的保单资源相关情况进行简单的数据分析,包括对投保人类型、保险销售人员业绩的分析等。该案例贯穿整个教学过程,在随后的几章中将会逐一介绍如何运用 OLAP (联机分析处理) 工具从多个角度审视保险行业的保单资源信息。

2.4.1 数据分析的需求和目标

本案例将针对不同的需求建立不同的多维数据集和实现不同的功能,例如针对保单资源分析建立的“保单资源分析”多维数据集;针对主管人员建立的“业务经理专用”多维数据集;针对助理人员建立的“保单资源公用分析”多维数据集;针对汇总功能实现的“汇总公式”功能;针对用户自定义功能实现的“自定义公式范例”功能;以及虚拟维度和虚拟多维数据集的实现。

本案例模拟实际情况,从不同角度对数据进行分析,进而建立多种维度:结构简单的单层标准维度,如“收付费类型”;多层次标准维度,如“险种_投保人类型分类”、跨越多个数据表的“区域_机构_业务员”;父子结构的维度,如“业务员等级”;时间维度,如“时间”;虚拟维度,如“险种分类_标准赔付率”。

一般来说,对保单资源,人们关心的主要是“新单保费”、“续期保费”、“退还保费”、“实付赔款”等指标。但是,我们依然希望通过这些显而易见的数据,通过一定的统计方法,来得到更多的信息,如“实际收入”、“实际收入环比”或部分业务人员的统计数据等。

2.4.2 数据库的结构和内容

本案例的数据源是 SQL Server 2000 数据文件, 名称为“derm”, 存有模拟一个寿险公司的各类业务数据。从“关系数据库”的观点来看, 该数据库包含以保单资源为主的“FACT_保单”。在建立保单的过程中, 涉及“机构”和“投保人员”等, 包括“机构区域”、“业务员”、“投保客户”等; 保单中涉及的不同险种根据其不同的性质, 又可以从销售渠道、险种类别等角度对其进行分类。此外, 在该数据库中还包括关于收付费类型的明细信息, 用于表示保单的收付费情况等; 还有一张“中间表”, 该表包含了保单、客户、业务员和险种类别等基本信息, 是对保险业务中的基本信息进行连接的桥梁和纽带。通过该表, 可以实现从法人客户角度对投保人数、保险金额、保单数量进行分析; 也可以实现从险种类别角度对投保人类型、投保人数、保险金额、保单数量进行分析; 另外, 还可以进行业务员业绩分析和法人客户单位性质与客户状态的分析。下面给出所用表的详细信息。

1) 表 1: “FACT_保单”

字段: “保单 ID”, “时间”, “预收保费”, “新单保费”, “续期保费”, “退还保费”, “实付赔款”, “实付退保金”, “实付生存金”, “收付费类型 ID”, “管理费比例”。

2) 表 2: “DIM_机构区域分类”

字段: “区域 ID”, “区域分类”。

3) 表 3: “DIM_机构细分”

字段: “机构 ID”, “机构名称”, “区域 ID”。

4) 表 4: “DIM_业务员”

字段: “业务员 ID”, “业务员名称”, “机构 ID”。

5) 表 5: “DIM_业务员等级关系”

字段: “业务员 ID”, “业务员名称”, “机构 ID”, “上级主管 ID”。

6) 表 6: “DIM_投保人类型险种分类”

字段: “投保人类型”, “险种代码”, “险种名称”。

7) 表 7: “DIM_销售渠道险种分类”

字段: “销售渠道类型”, “险种代码”, “险种名称”。

8) 表 8: “DIM_险种类别分类”

字段: “险种分类”, “险种代码”, “险种名称”, “险种标准赔付率”。

9) 表 9: “DIM_收付费类型明细”

字段: “收付费明细类型 ID”, “收付费明细类型表示”, “收付费归并 ID”, “收

付费归并类型”，“收付费”。

10) 表 10:“DIM_中间表”

字段:“保单 ID”,“业务员 ID”,“险种 ID”,“承保类型 ID”,“客户 ID”,“销售渠道 ID”,“保单状态 ID”,“投保人数”,“生效时间”,“保险金额”。

2.4.3 注册 Analysis Servers

安装了 SQL 2000 中的“Analysis Services”后,系统便会根据本机的计算机名称建立一个 Analysis Servers。本节以下的工作,都可以在此默认的 Analysis Servers 中建立,但也可以建立一个连接到其他计算机上的 Analysis Servers。

第 1 步 安装了 SQL 2000 中的“Analysis Services”之后,通过“开始”菜单,打开“Analysis Manager”窗口。

第 2 步 用鼠标右键单击“Analysis Servers”,在弹出的菜单中选择“注册服务器”命令,在出现的“注册分析服务器”对话框中写入使用的服务器名称,如图 2.1 所示。

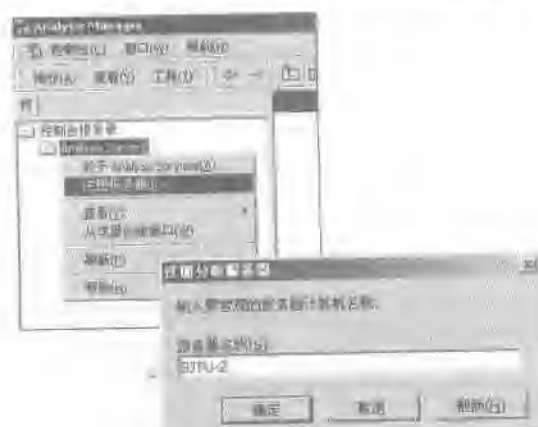


图 2.1 注册新的 Analysis Servers

2.4.4 设置数据库

设置新的 Analysis Servers 后,可以继续设置隶属于每一个 Analysis Servers 的数据库。

第 1 步 在 Analysis Servers 上单击右键,选择“新建数据库”,在出现的“数据库”对话框中输入数据库名称,如图 2.2 所示。

第2步 在“数据链接属性”对话框的“提供程序”选项卡中,选择“Microsoft OLE DB Provider for SQL Server”,如图2.4所示。



图2.4 “提供程序”选项卡

第3步 单击“下一步”按钮,在出现的“连接”选项卡中,选择“在服务器上选择数据库”下的“dcrm”,如图2.5所示。

第4步 单击“确定”按钮,完成设置,可以看到在数据源目录下多了一个“dcrm”数据源,如图2.6所示。

2.5 扩展实验

在学完本章主要内容之后,请读者自己完成以下练习:

(1) 这里用到的数据库也是名称为“dcrm”的SQL关系数据库文件,主要用到“DIM_中间表”、“DIM_业务员等级关系”、“dim_投保人类险种分类”、“dim_险种类别分类”、“dim_保单类型”、“Dim_法人客户表”、“dim_单位性质”、“dim_客户状态表”。其中,将“DIM_中间表”作为事实表建立了多维数据集“法人客户价值分析保单数量”、“法人客户价值——虚拟维度”和虚拟多维数据集“业务经理专

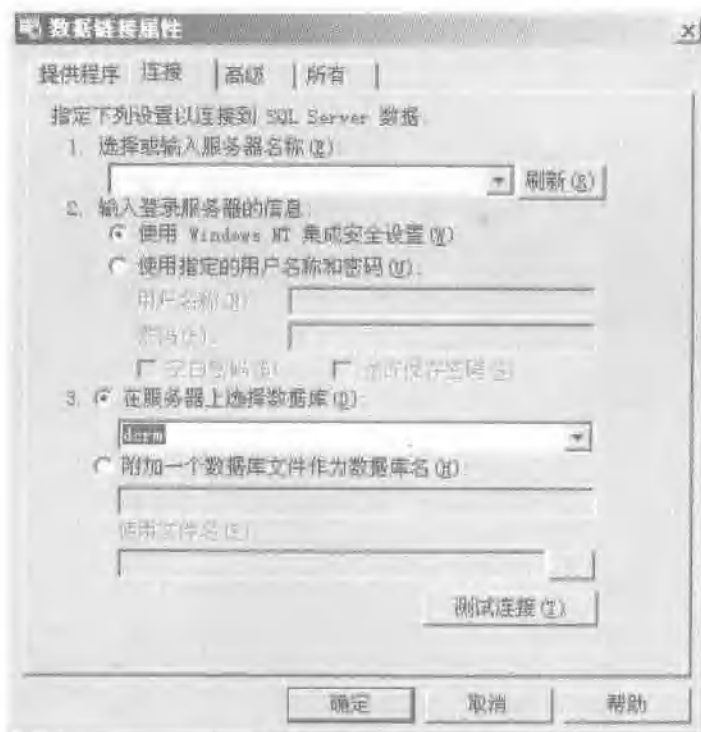


图 2.5 在服务器上选择数据库

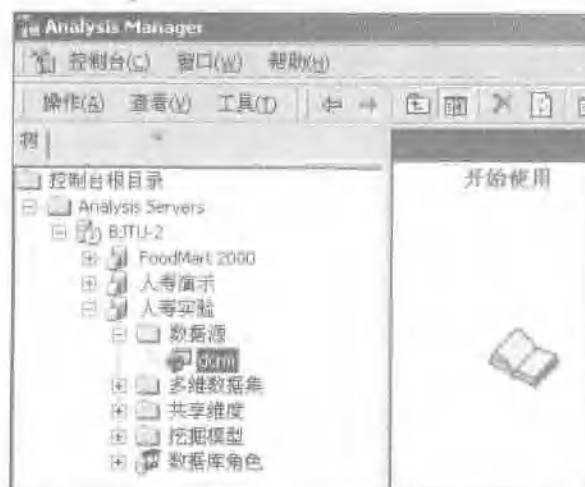


图 2.6 增加了“dorm”数据源

用”,来实现对投保人类型、投保人数、保险金额、保单数量、业务员业绩、法人客户单位性质与客户状态的分析。请对所用到的数据表进行详细介绍,主要是字段分析。

(2) 要求在 Analysis Manager 中的“Analysis Servers”下,注册名称为你自己的计算机名称的服务器,新建名称为“人寿实验”的数据库,选择服务器上的“dcrm”数据库作为数据源。

本章小结

本章简要介绍了案例所用数据的基本结构和基本关系,重点介绍了数据仓库的系统环境配置方法。这对读者深入理解书中的案例非常重要。通过本章内容的学习,读者应了解数据分析的目标和作用,明确数据库表的基本内容、表与表之间的联系,掌握分析服务器、数据库、数据源设置的基本方法。当然,对于不同的数据库,有不同的数据源设置方法,比如可以借助 ODBC 等工具来设置数据源,由于本书案例没有使用,此处不再赘述。



第3章

创建多维数据集

本章引言

本章将说明如何使用 Microsoft SQL Server 2000 的 Analysis Services 构造星型模型和雪花型模型,如何用这些模型创建多维数据集,以及在创建过程中可能遇到的技术问题及其解决方法。

Analysis Services 向导包括维度向导、多维数据集向导、虚拟维度向导、存储设计向导、基本应用优化向导、应用分析向导、维向导和虚拟立方体向导,用这些向导和编辑器可以方便、快捷地创建多维数据集和各种模型结构。

本章重点

- 时间维度的建立方法。
- 单一层次的标准维度的建立过程。
- 多层次标准维度的建立方法。
- 多重关系型标准维度的建立方法。
- 运用维度编辑器对维度进行编辑。
- 运用多维数据集编辑器对多维数据集进行编辑。
- 多维数据集的存储。

3.1 实验预备知识

在浏览并应用 Analysis Services 向导之前,先介绍几个 OLAP 的重要概念以及一些相关知识点。

3.1.1 共享维度

共享维度是指可以用在同一个数据库的不同多维数据集上的维度,与它相对的是“专用维度”。维度按结构可分为 5 种形式,即“星型架构:单个维度表”、“雪花架构:多个相关维度表”、“父子维度:单个维度表中相关的两列”,“虚拟维度:另一个维度的成员属性”、“挖掘模型:OLAP 挖掘模型的可预测列”。它们代表数据分析的不同需求和实体数据保存的不同方式。一般使用维度向导来建立各种维度。同时,从维度的意义上说,维度可以分为“时间维度”和“标准维度”。

3.1.2 多维数据集

一个多维数据集(Cube)是由很多数据维(Dimension,简称“维”)组成的,每一个维都表示某一类数据。维里的数据都限制在某一问题所界定的领域之内,在 Analysis Services 里的多维数据集可以包含 1~64 个数据维。

一个数据维可以包含一个或多个层次(Hierarchy),层次是多层结构中某一级别与下一级别之间的“父-子”关系,位于层次顶部的是“根级(Root Level)”,位于层次底部的是一个或多个“叶子(leaf)”。层次中的任何一级都可以看成是数据维里的一类数据。层次里某一级别的一个成员(Member),也称为级别值(Level Value),包含了由一个级别所表示的数据集合。层次的顶部(根级)容纳了粗略划分的数据集合,而在层次的底部(叶子)则包含了精细划分的数据集合。

3.2 实验目的

理解维度、多维数据集的基本概念,掌握时间维度、标准维度和基本多维数据集的建立与编辑过程。

3.3 实验要求

掌握共享维度建立的方法和要求,理解多维视图使用维和度量进行数据分析

的方法,学会根据不同的应用选择相应的多维数据存储模型。

3.4 实验内容与步骤

本次实验的主要内容是共享维度的建立与编辑,包括时间维度、单一层次的标准维度、多层次标准维度、多重关系型标准维度等,从而为后续的多维数据集的创建做准备。

3.4.1 建立时间维度

根据数据库中的“FACT_保单”表中的“时间”数据列,从“年、季度、月、日”的角度分析数据,并建立时间维度,名称为“时间”。

第1步 在“Analysis Manager”窗口中的“人寿实验”下的“共享维度”上,以右键单击打开快捷菜单,选择“新建维度”,再选择“向导”,如图3.1所示。

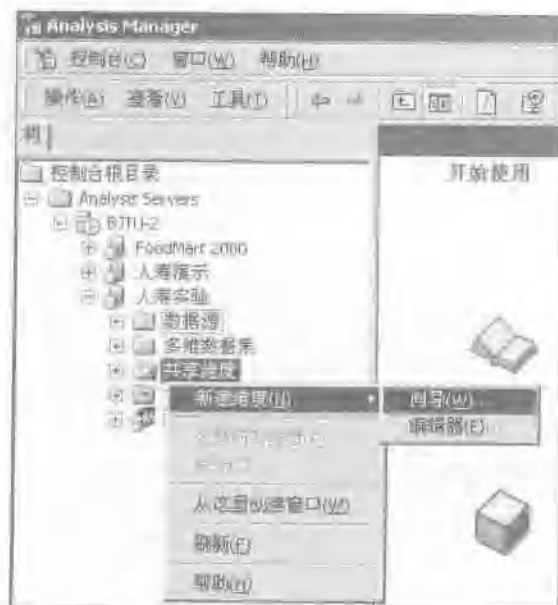


图 3.1 建立维度向导

第2步 这时,屏幕出现“欢迎”对话框,这个对话框不具有意义,可以选中“以后不再显示这个画面”。单击“下一步”按钮,出现“选择维度的创建方式”对话框。单击“下一步”按钮,出现“选择维度表”对话框,在左面的“可用的表”下选择“FACT_保单”,在右面的“详细信息”下会出现该表中的列,如图3.2所示。

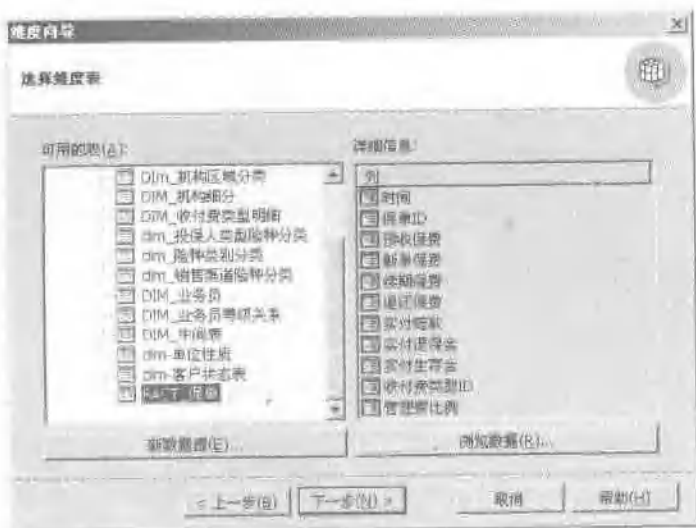


图 3.2 “选择维度表”对话框

第3步 单击“下一步”按钮,出现“选择维度类型”对话框,选择“时间维度”,并在“日期列”框中选择要作为时间维度的列,如图 3.3 所示。

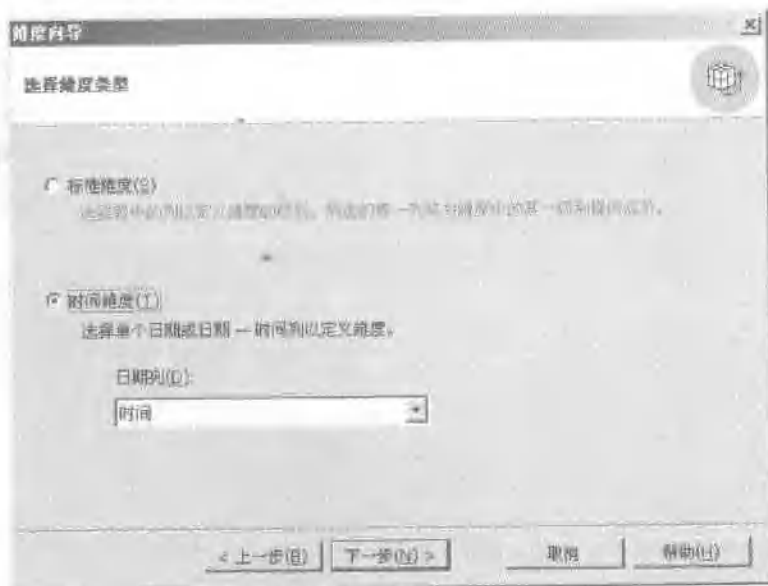


图 3.3 “选择维度类型”对话框

第4步 单击“下一步”按钮,出现“创建时间维度级别”对话框,选择不同的

时间结构,本例选择“年、季度、月、日”,如图 3.4 所示。

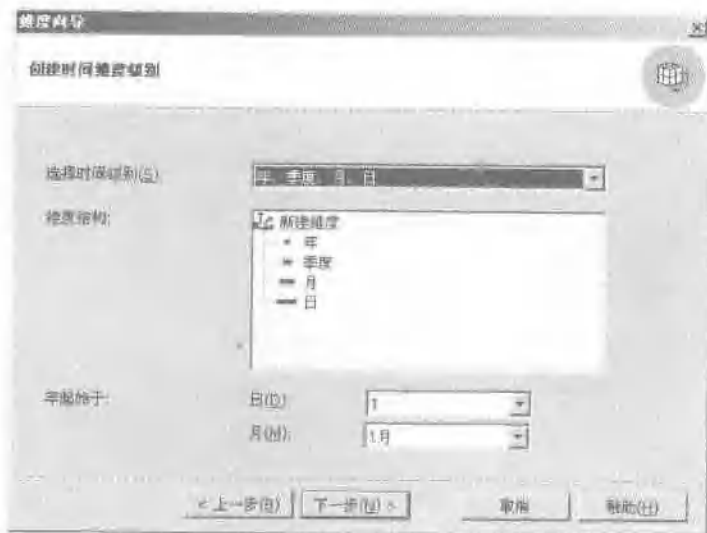


图 3.4 “创建时间维度级别”对话框

第 5 步 单击“下一步”按钮,出现“选择高级选项”对话框,如图 3.5 所示。

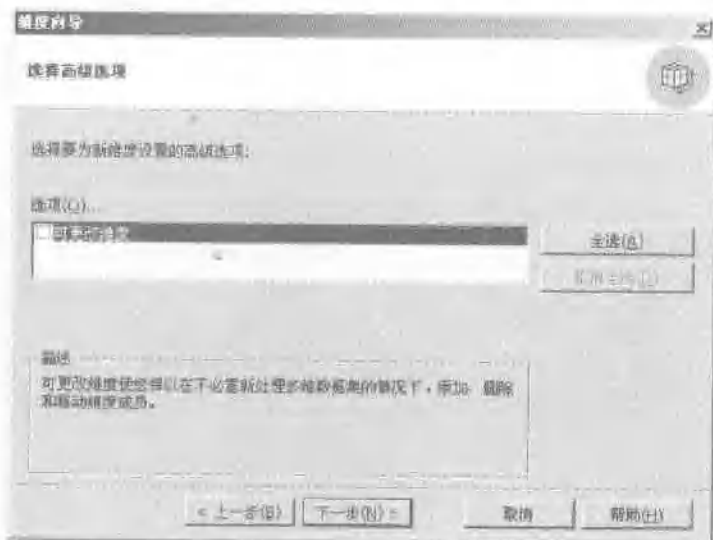


图 3.5 “选择高级选项”对话框

第 6 步 单击“下一步”按钮,出现“维度向导”对话框,在“维度名称”框中输入“时间”,同时可以在“预览”中查看维度的层次数据结构,如图 3.6 所示。

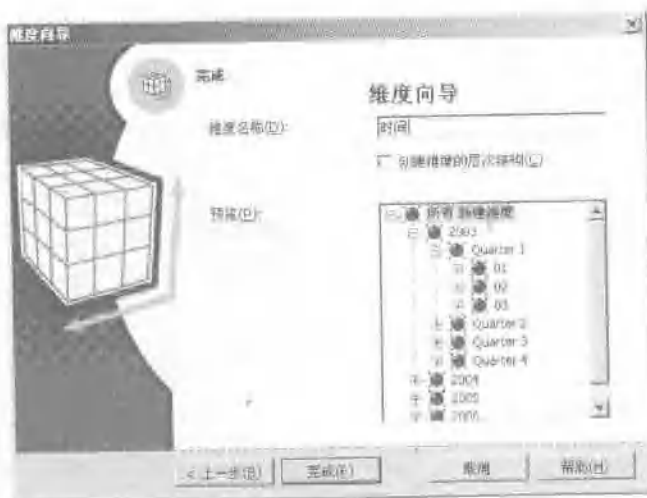


图 3.6 “完成”对话框

3.4.2 建立单一层次的标准维度

以“收付费类型”为标准,对保单资源进行分析。

第 1 步 单一层次的标准维度建立的过程与建立时间维度相同,将如图 3.2 所示的“选择维度表”对话框的“可用的表”下改选为“DIM_收付费类型明细”,这时出现“选择维度的级别”对话框,选择“收付费”,如图 3.7 所示。

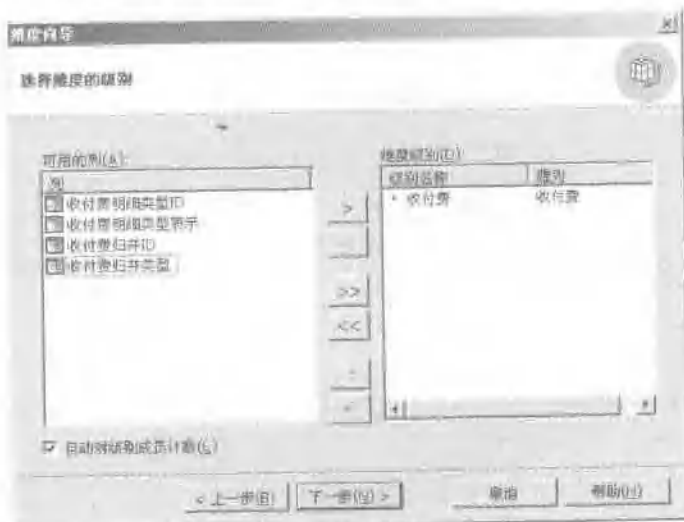


图 3.7 “选择维度的级别”对话框

第2步 单击“下一步”按钮,打开“指定成员键列”对话框。单击“下一步”按钮,出现“选择高级选项”对话框。单击“下一步”按钮,出现如图 3.6 所示的同样对话框,在“维度名称”框中输入“收付费类型”,单击“完成”按钮。

3.4.3 建立多层次标准维度

从投保人类型的角度,对保单资源进行分析。

第1步 建立多层次标准维度的过程与建立时间维度相同。在如图 3.2 所示的“选择维度表”对话框的“可用的表”下改选为“dim_投保人类型险种分类”,出现“选择维度的级别”对话框。在“维度级别”下依次选择“投保人类型”和“险种名称”构建出两个层次(注意选择层次的先后),如图 3.8 所示。

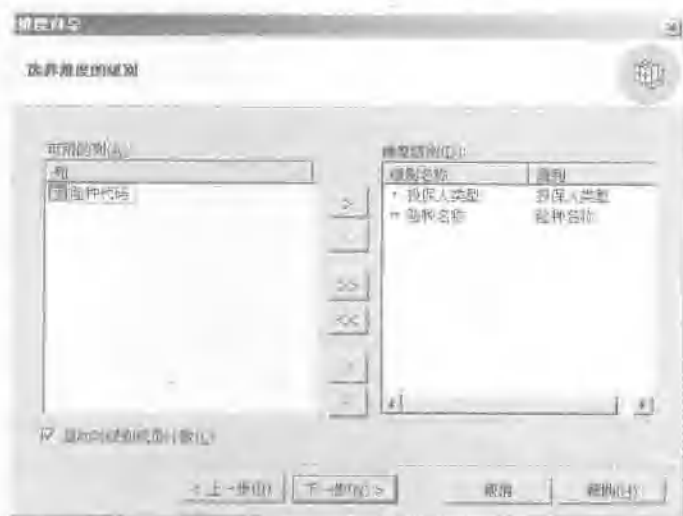


图 3.8 选择“投保人类型”和“险种名称”

第2步 单击“下一步”按钮,出现“指定成员键列”对话框。单击“下一步”按钮,出现“选择高级选项”对话框。单击“下一步”按钮,出现如图 3.6 所示的同样对话框,在“维度名称”框中输入“险种_投保人类型分类”,单击“完成”按钮。

3.4.4 建立多重关系型标准维度

从业务员及业务员所属机构的角度来分析保单资源,这时就要建立多重关系型维度——“区域_机构”维度。

第1步 从“维度向导”对话框进入到“选择维度的创建方式”对话框,并选择

“雪花架构:多个相关维度表”,如图 3.9 所示。

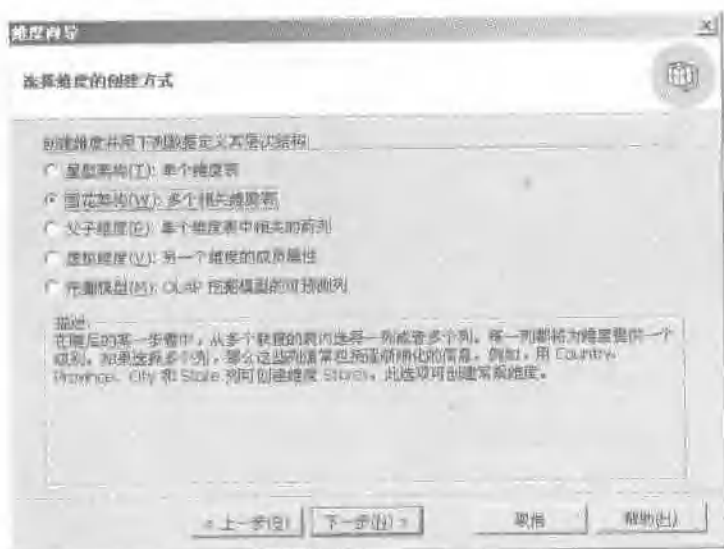


图 3.9 “选择维度的创建方式”对话框

第 2 步 单击“下一步”按钮,打开“选择维度表”对话框,在“选定的表”下选择“Dim_机构区域分类”、“DIM_机构细分”和“DIM_业务员”,如图 3.10 所示。

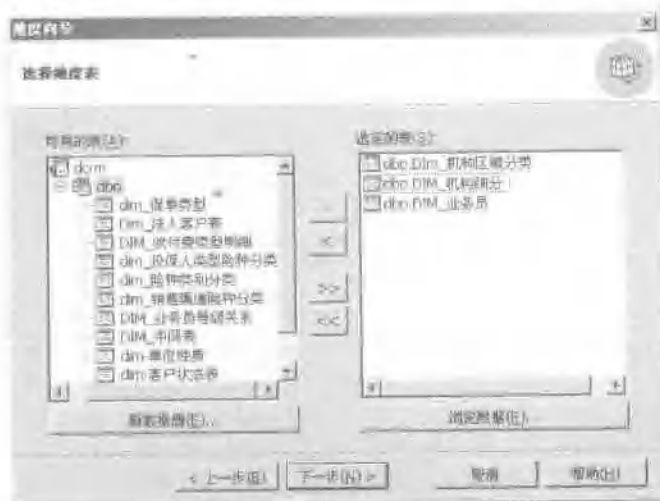


图 3.10 “选择维度表”对话框

第 3 步 单击“下一步”按钮,出现“创建和编辑联接”对话框,确认是由“区域 ID”和“机构 ID”连接 3 张表,如图 3.11 所示。

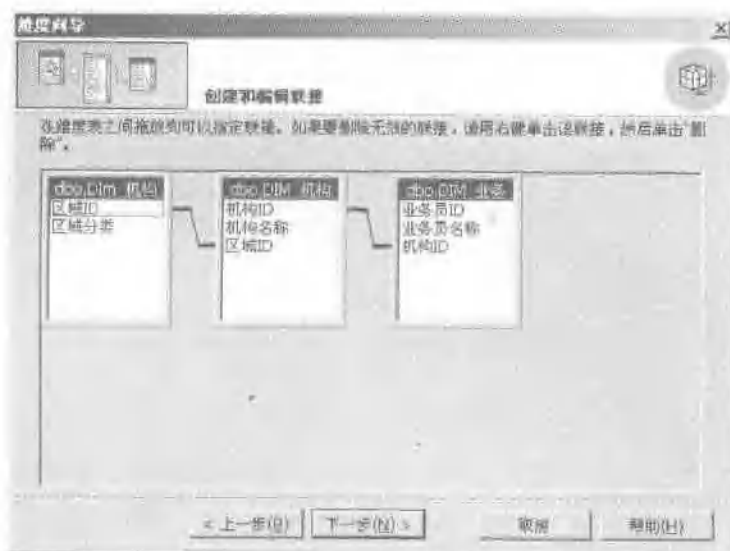


图 3.11 “创建和编辑联接”对话框

第4步 单击“下一步”按钮,出现“选择维度的级别”对话框,选择维度字段(注意级别的顺序),如图 3.12 所示。

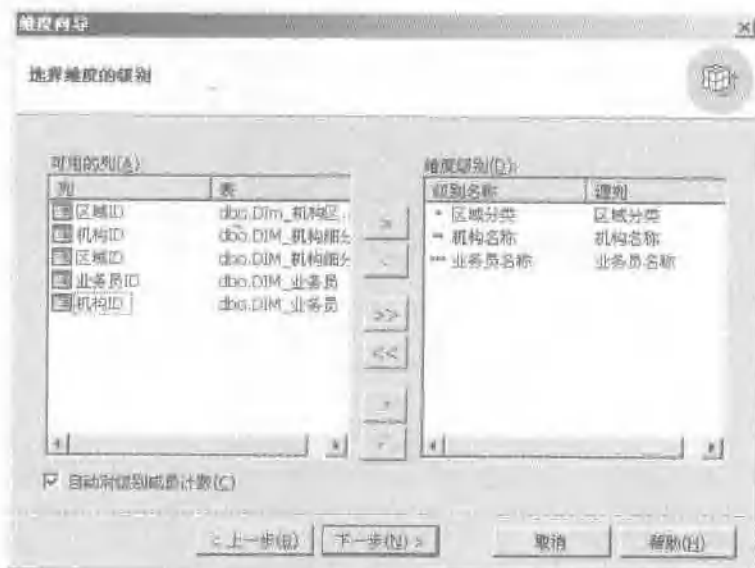


图 3.12 “选择维度的级别”对话框

第5步 其余步骤与前面建立“时间维度”的步骤相同,一直到最后的“完成”对话框。然后在“维度名称”框中,为此多层次维度命名;也可以选中“创建维度的层次结构”复选框,并在“层次结构名称”框中,为子级别的数据命名,如图 3.13 所示。

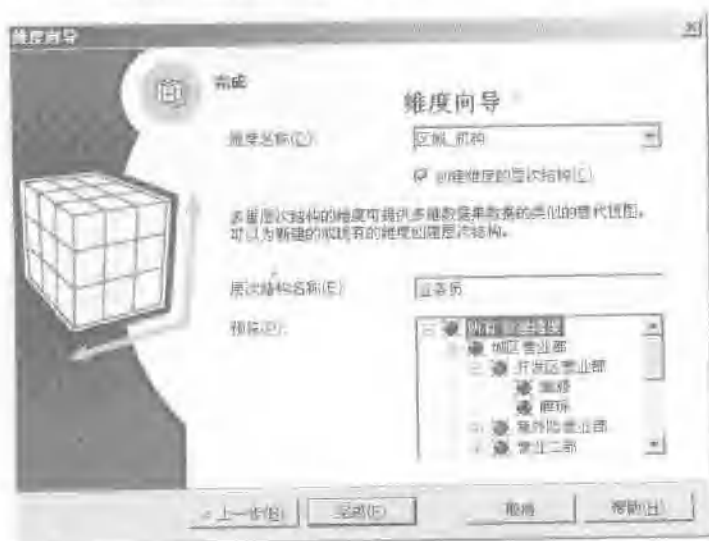


图 3.13 为多层次结构的维度命名

3.4.5 维度的编辑

更改时间维度,将季度的格式“Quarter N”更改为“第 N 季”,显示数据;在“收付费类型”维度中添加两个层次:“收付费归并类型”和“收付费明细类型表示”。

第1步 打开“时间”维度的“维度编辑器”窗口,选择“季度”项目,同时选择下方“属性”中的“基本”选项卡,将“Member Name Column”项目中的“'Quarter' + convert(CHAR, DatePart(quarter, 'dbo', 'FACT_保单', '时间'))”改为“'第' + convert(CHAR, DatePart(quarter, 'dbo', 'FACT_保单', '时间')) + '季'”,如图 3.14 所示。

第2步 打开“收付费类型”维度的“维度编辑器”窗口,在“收付费”上单击右键,选择“新建级别”,出现“插入级别”对话框。选择“收付费归并类型”,单击“确定”按钮,如图 3.15 所示。

第3步 以相同的方式插入“收付费明细类型表示”级别。

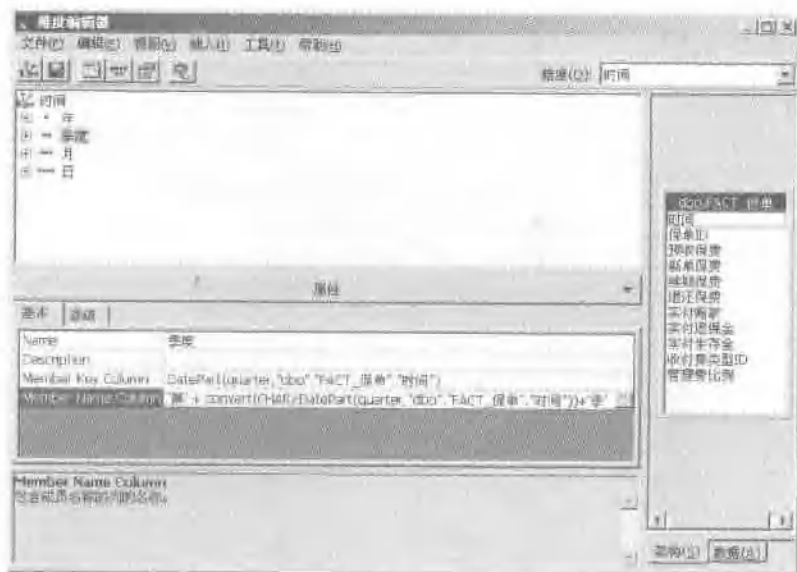


图 3.14 “维度编辑器”窗口

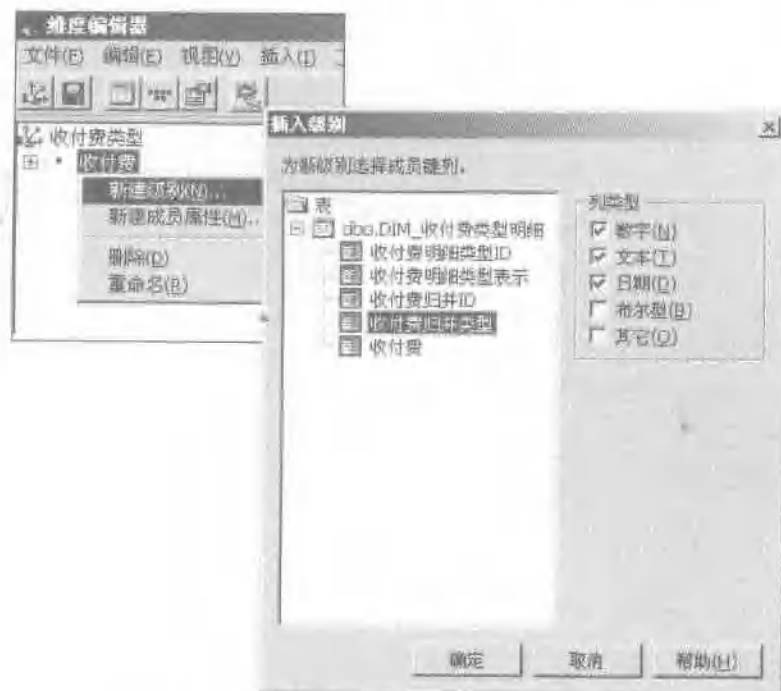


图 3.15 建立“收付费归并类型”级别

3.4.6 多维数据集的编辑

建立“保单资源分析”多维数据集,向多维数据集添加“收付费类型”、“险种_投保人类别分类”以及“区域_机构_业务员”维度;添加“时间”维度;新建“保单ID”和“管理费比例”度量值;将“管理费比例”的运算方式改为“计算最大值”,并将它的显示格式改为“百分比格式”;处理多维数据集。

第1步 在“人寿实验”下的“多维数据集”上单击右键,选择“新建多维数据集”→“向导”命令,如图3.16所示。



图 3.16 使用“向导”建立多维数据集

第2步 这时屏幕出现“欢迎”对话框,可以单击“以后不再显示这个画面”关闭该对话框。单击“下一步”按钮,出现“从数据源中选择事实数据表”对话框,在左面的“数据源和表”下选择“FACT_保单”,在右面的“详细信息”下出现该表的表结构,如图3.17所示。

第3步 单击“下一步”按钮,打开“选择用于定义度量值的数字列”对话框,将“预收保费”、“新单保费”、“续期保费”、“退还保费”、“实付赔款”、“实付退保金”、“实付生存金”选中,作为度量值,如图3.18所示。

第4步 单击“下一步”按钮,打开“选择多维数据集的维度”对话框,在“多维数据集维度”下将“区域_机构_业务员”、“收付费类型”和“险种_投保人类别分

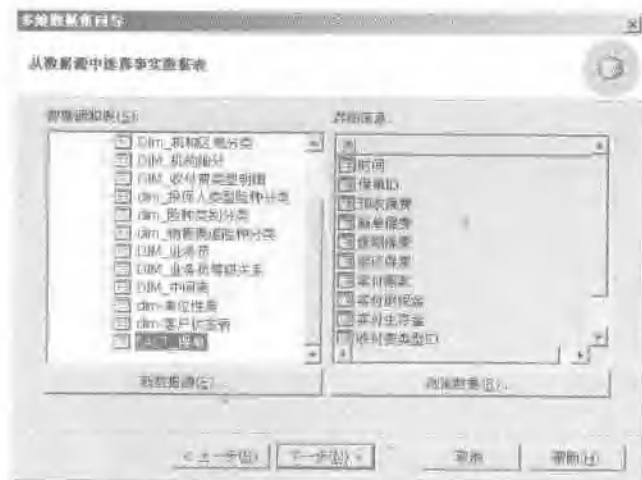


图 3.17 “从数据源中选择事实数据表”对话框

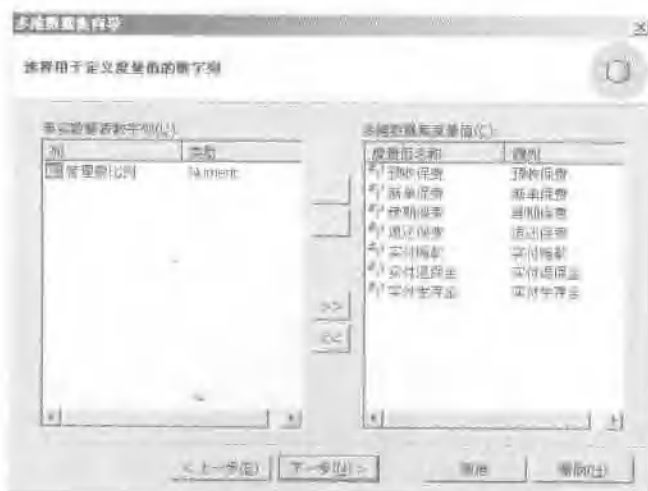


图 3.18 “选择用于定义度量值的数字列”对话框

类”三个维度选中,如图 3.19 所示,单击“下一步”按钮。

第5步 这时屏幕出现“事实数据表行数”提示框,单击“是”按钮,如图 3.20 所示。

第6步 屏幕出现如图 3.21 所示的提示信息:“无法找到多维数据集的事实数据表和下列维度表之间的自动连接”。单击“确定”按钮,在后面通过手工添加中间表来建立事实表与维度表的联系。

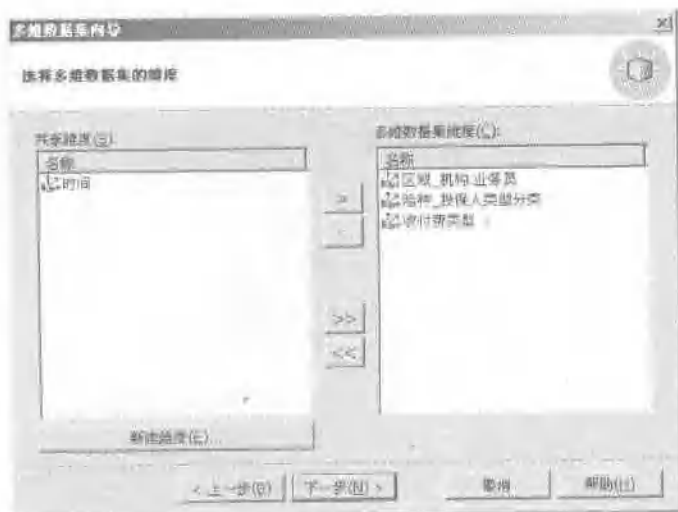


图 3.19 “选择多维数据集的维度”对话框

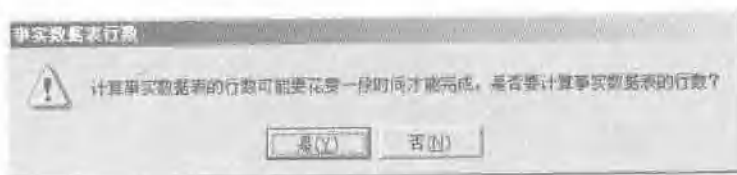


图 3.20 “事实数据表行数”提示框

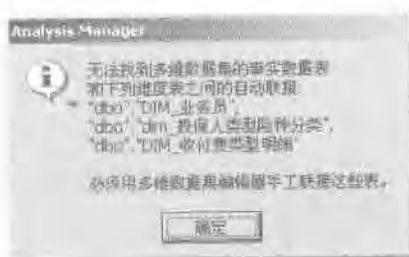


图 3.21 提示信息

第7步 在“完成”对话框中,在“多维数据集名称”框中输入“保单资源分析”,单击“完成”按钮,如图 3.22 所示。

第8步 进入“多维数据集编辑器”窗口,在右面的“结构”面板中,单击鼠标右键,选择“插入表”,如图 3.23 所示。



图 3.22 输入“保单资源分析”



图 3.23 插入表

第9步 这时,出现如图 3.24 所示的“选择表”对话框,在左面的“表”下选中“DIM_中间表”,单击“添加”按钮。

第10步 用鼠标左键选中“DIM_中间表”的“业务员 ID”,按住左键,将鼠标拖至“DIM_业务员”表中“业务员 ID”的上方,松开鼠标左键,这样就将两个表连接起来。用同样的方法将“DIM_中间表”中的“险种 ID”与“dim_投保人类型险种分类”表中的“险种代码”连接起来,再将“FACT_保单”表中的“收付费类型 ID”与“DIM_收付费类型”表中的“收付费明细类型 ID”连接起来,如图 3.25 所示。



图 3.24 “选择表”对话框



图 3.25 表的连接

第11步 打开“保单资源分析”的“多维数据集编辑器”窗口,在“维度”上单击鼠标右键,选择“现有的维度”,出现如图 3.26 所示的“维度管理器”对话框,选择“时间”维度。

第12步 打开“保单资源分析”的“多维数据集编辑器”窗口,在“度量值”上单击鼠标右键,选择“新建度量值”,出现“插入度量值”对话框。单击“文本”,在左面出现了“保单 ID”,将其选中,单击“确定”按钮,如图 3.27 所示。



图 3.26 增加维度



图 3.27 增加度量值

第13步 用同样的方法增加“管理费比例”度量值。在“多维数据集编辑器”窗口的“度量值”中选择“管理费比例”，在“属性”中选择“基本”选项卡，在“Aggregate Function”右侧下拉列表中选择“Max”；在“高级”选项卡的“Display Format”右侧下拉列表中选择“Percent”，如图 3.28 所示。



图 3.28 修改度量值属性

第14步 在“多维数据集编辑器”窗口中，选择“文件”→“退出”命令，出现“设计存储”对话框，如图 3.29 所示，单击“是”按钮。

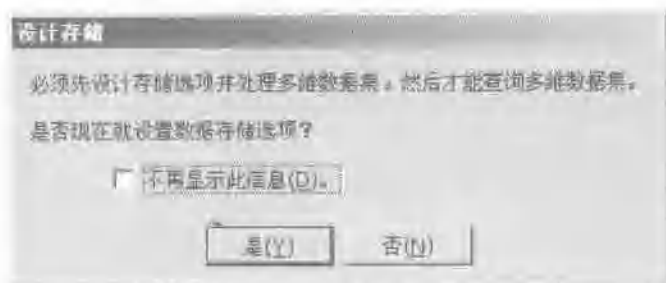


图 3.29 “设计存储”对话框

第15步 进入“存储设计向导”，在“选择数据存储类型”对话框中选择“MOLAP”，单击“下一步”按钮，如图 3.30 所示。

第16步 进入如图 3.31 所示的“设置聚合选项”对话框，进行聚合选项的测试，单击“下一步”按钮。

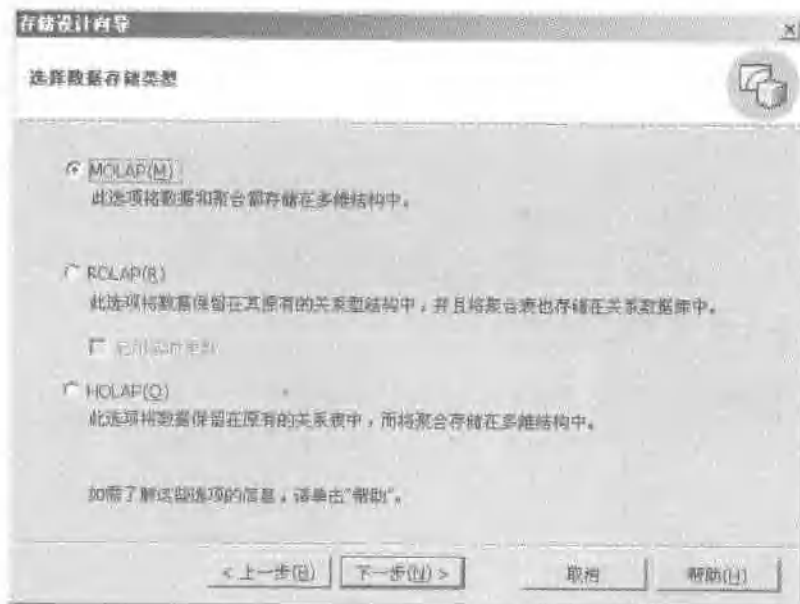


图 3.30 “选择数据存储类型”对话框

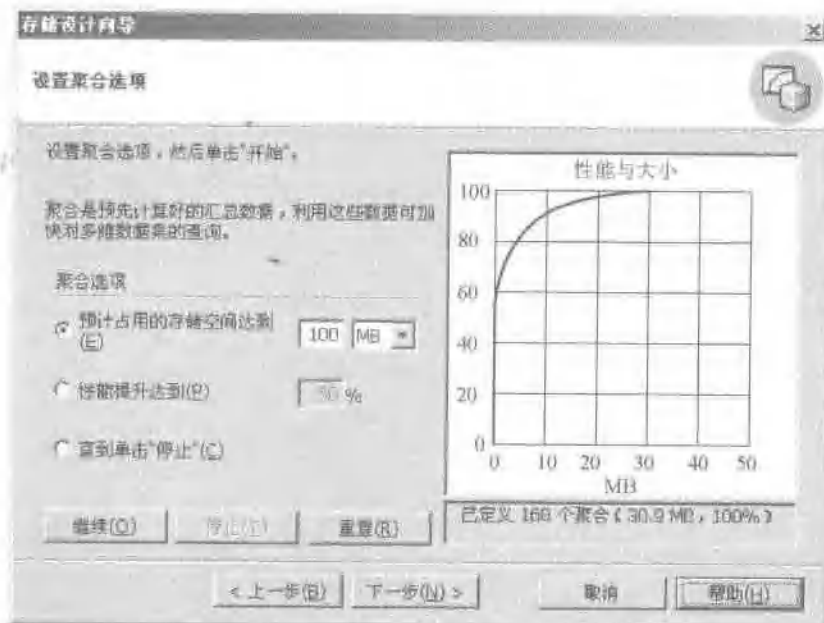


图 3.31 “设置聚合选项”对话框

第17步 在“完成”对话框中,选中“立即处理”,单击“完成”按钮,如图 3.32 所示。



图 3.32 立即处理多维数据集

第18步 已成功完成多维数据集处理的界面,如图 3.33 所示。

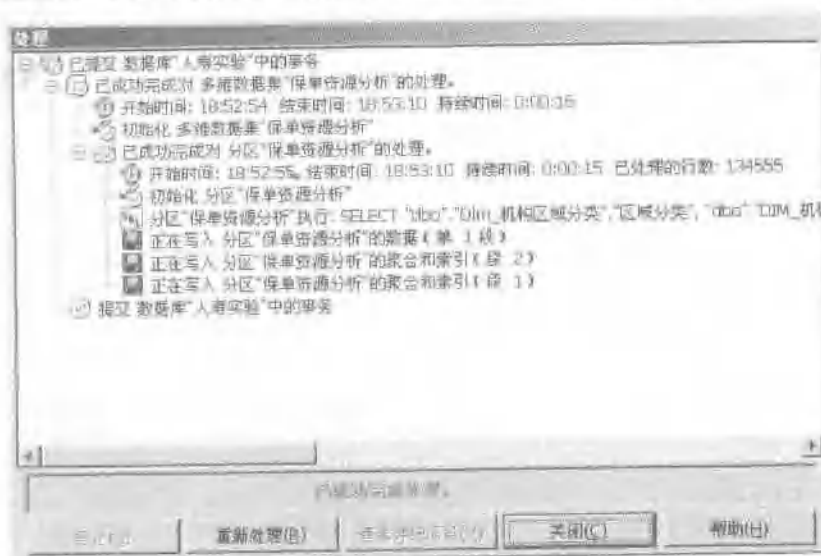


图 3.33 已成功完成多维数据集处理

3.5 扩展试验

下面是本章的扩展实验,可供有兴趣、有能力的读者进行巩固提高。

1. 建立时间维度

建立“星型架构”的“生效时间”维度,维度级别为“年、月、日”,这里用到“DIM_中间表”。

2. 建立单一层次的标准维度

通过“dim_保单类型”源表,建立“星型架构”的“保单类型”维度,其中只含有一个“保单类型”维度级别(注意维度级别的定义)。

3. 建立多层次标准维度

建立名称为“险种_险种类型分类”的有两个维度级别的“星型架构”的标准维度,选择“dim_险种类别分类”表,维度级别分别是“险种分类”和“险种名称”(注意维度级别的先后顺序)。

4. 建立多重关系型标准维度

建立具有两层维度级别的“星型架构”的标准维度,名称为“开户年.法人客户”,维度级别为“开户年”、“客户名称”(注意顺序),这里用到“DIM_法人客户表”。

更改“生效时间”时间维度,将“月”的格式属性“N”更改为“第 N 月”来显示数据;同时,在“保单类型”维度中添加“险种名称”和“险种代码”两个层次。

5. 编辑多维数据集

通过中间表,对法人客户的价值保单数量进行分析。建立“法人客户价值分析保单数量”多维数据集,度量值为“保险金额”和“投保人数”,在多维数据集中添加“保单类型”、“险种_险种类型分类”以及“开户年.法人客户”三个维度;添加现有的“生效时间”时间维度;通过“保单 id”新建“保单数量”度量值;将“保险金额”的运算方式改为“计算最大值”;将“保单数量”的显示格式改为“#, #”,将数据类型改为“integer”,将聚合函数改为“count”;进入“设计存储向导”,在“选择数

据存储类型”对话框中选择“MOLAP”,在“设置聚合选项”对话框中,选择“性能提升达到 50%”,最后成功完成对多维数据集的处理。

本章小结

Analysis Services 是 SQL Server 数据库中附加的一个分析管理器,提供了灵活的多维数据集分析功能和数据挖掘功能,简单易用。本章主要介绍了多维数据集的建立和编辑。

在建立多维数据集前,要选择合适的数据源,建立多维数据集的两个关键因素分别是度量值的确定和维度的建立。度量值是用户要分析的指标,一般为数字类型。维度是用户分析指标时所选用的角度,如时间维度等。多维数据集的建立可以使用多维数据集建立向导,先选择事实表和度量值,然后再建立维度。维度的结构有 5 种,即“星型架构:单个维度表”、“雪花架构:多个相关维度表”、“父子维度:单个维度表中相关的两列”、“虚拟维度:另一个维度的成员属性”、“挖掘模型:OLAP 挖掘模型的可预测列”。建立哪种维度要视具体需求和维度表、事实表的结构而定。建立好多维数据集后,通常要对其进行编辑,多维数据集的编辑在“多维数据集编辑器”对话框中进行,可以添加、删除和修改度量和维度。在该对话框中还可以浏览数据,对多维数据集进行保存和存储设计。

基于 Analysis Services 的数据仓库解决方案,具有成本低、方便灵活的特点,同时可以结合一些第三方提供的前台展示和查询工具,使之功能更加丰富。



第4章

复杂多维数据集的创建

本章引言

上一章对多维数据集的创建和编辑已经做了详细的介绍,主要包括时间维度、单一层次的标准维度、多层次标准维度、多重关系型标准维度的建立与编辑等。在实际需求中,基于 Analysis Services 的数据仓库解决方案往往需要从多种角度去考虑问题,从而做出正确的决策,因此要用到一些不同于前面介绍的复杂的维度和多维数据集。

本章的主要内容为复杂多维数据集的建立,包括父子维度、计算成员、成员属性、虚拟维度的建立与应用等。

本章重点

- ✦ 父子维度的建立。
- ✦ 计算成员的建立。
- ✦ 成员属性的建立与编辑。
- ✦ 虚拟维度的建立与应用。

4.1 实验预备知识

实际的操作设计离不开理论知识的支持,有了正确的理论知识的指导,后续的实验设计便会事半功倍。下面介绍有关复杂多维数据集的几个重要概念,如父子维度、计算成员等。

4.1.1 父子维度

在上一章中曾介绍过建立单一层次与关系型层次维度的做法,其中“层次”的概念,主要是表现在不同字段之间的“一对多”关系性。倘若这种一对多的关系,是发生在不同的数据表间,则在建立层次维度时,必须使用“雪花架构”的方式进行,但对于某些数据的结构,这些数据层次的关系发生在同一表中,如在本书实验中,有一个“业务员”数据,其中“业务 ID”为此数据表中的唯一键(主键),每一个业务也都有隶属的“主管”,因此以“主管 ID”字段为代表。从数据表来看,发生层次关系的项目来自于同一数据表的不同字段;再从“维度”角度来看,可能会分析“某一业务”的业绩,但也可能会分析“某一主管”管辖下的总业绩。要达到前一个目的,则必须使用 Analysis Manager 中所提供的父子维度来实现。

父子维度是由成员的父子关系定义的成员组织好的层次结构,这种数据关系发生在一个表内。通常,其各个分支并不具有对称数量的级别。

对称层次结构如图 4.1 所示。

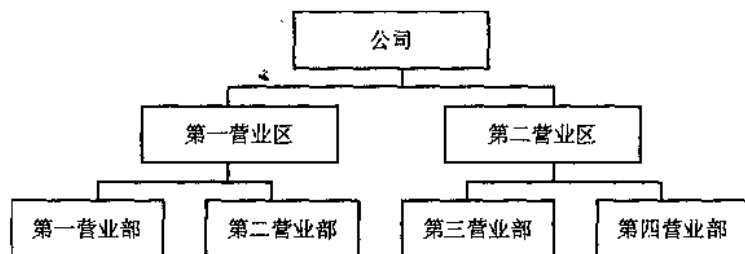


图 4.1 对称层次结构

不对称层次结构如图 4.2 所示。

4.1.2 计算成员

建立起来的维度和度量值有时不能满足分析的需要,根据已有的维度和度量值进行计算,建立新的维度和度量值,称为计算成员。计算成员通过在原始数据

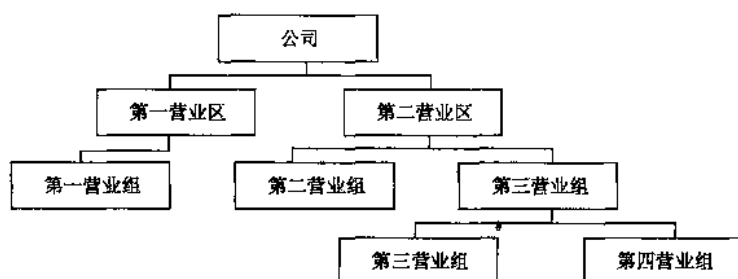


图 4.2 不对称层次结构

上建模,生成有意义的业务指标来增强分析。计算成员将增加分析的价值,它们可以描述趋势、行为和异常情况。

4.1.3 成员属性

从关系数据库的观点来说,对于逻辑数据库的定义,会以“实体”与“关系”来描述,每一个“实体”或“关系”都具有用来描述这些项目的“属性”。转换到实体数据库的设置,都可能以“表”来描述“实体”或“关系”,而属性则会转换成“表”中的“字段”。

同样,在“分析管理器”中,对于每一个“维度成员”,也可以使用隶属于该数据表的“字段”,作为描述维度成员的属性,称为“成员属性”。

成员属性是维度成员的一个特性。它为最终用户提供关于成员的其他信息。成员属性有多种用途,除提供关于成员的信息外,在分析多维数据集的数据时,还可在查询中使用成员属性为最终用户提供多个选项,成员属性还可作为虚拟维度中级别的基础。

4.1.4 虚拟维度

虚拟维度是基于物理维度内容的逻辑维度,这些内容可以是物理维度中的现有成员属性,也可以是物理维度的表中的列。

使用虚拟维度的优点是不占用磁盘空间,也不耗费处理时间。

4.2 实验目的

练习复杂维度、指标的建立和应用过程,掌握父子维度、计算成员、成员属性、虚拟维度的建立和应用过程。

4.3 实验要求

熟练掌握父子维度的建立方法和特点,理解和掌握计算成员在维度指标中的应用,了解成员属性的意义,掌握成员属性的建立方法,掌握通过成员属性来建立虚拟维度的方法。

4.4 实验内容与步骤

本节的实验内容主要是父子维度、计算成员、成员属性、虚拟维度的建立和应用,熟悉这一环节的内容之后,读者便可以按照实际需要建立符合要求的维度和数据集,以供决策分析之用。

4.4.1 父子维度的建立

新建一个名为“主管”的共享父子维度;再以“DIM_业务员等级关系”表为例,以“上级主管 ID”为父层,以“业务员 ID”为子层建立私有父子维度“业务员等级”,加入“保单资源分析”多维数据集,分析每一位成员(主管和部属)及其隶属成员的保单业绩。

第一步 在“Analysis Manager”窗口的“人寿实验”下的“共享维度”上,单击鼠标右键打开快捷菜单,选择“新建维度”,再选择“向导”,出现“选择维度的创建方式”对话框,选择“父子维度”,如图 4.3 所示。

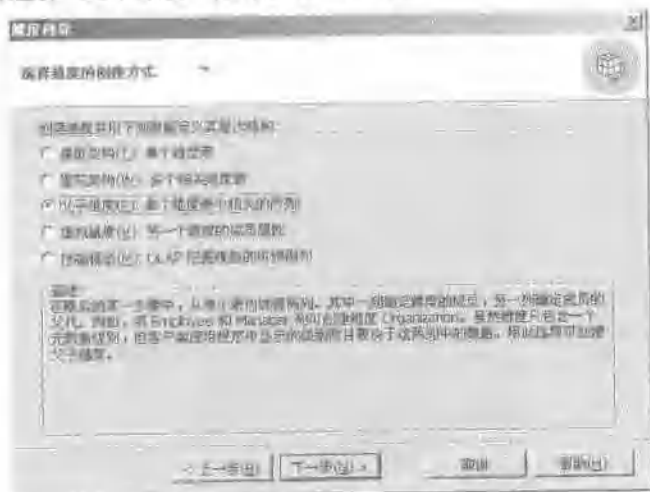


图 4.3 “选择维度的创建方式”对话框

第2步 单击“下一步”按钮,出现“选择维度表”对话框,在左面的“可用的表”下选择“DIM_业务员等级关系”,如图4.4所示。

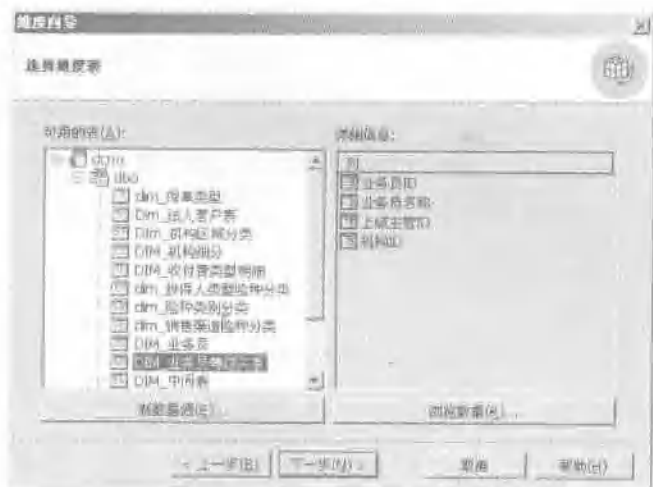


图 4.4 “选择维度表”对话框

第3步 单击“下一步”按钮,出现“选择定义父子数据层次结构的列”对话框,在“成员键”框中选择“业务员ID”,在“父键”框中选择“上级主管ID”,在“成员名称”框中选择“业务员名称”,如图4.5所示。

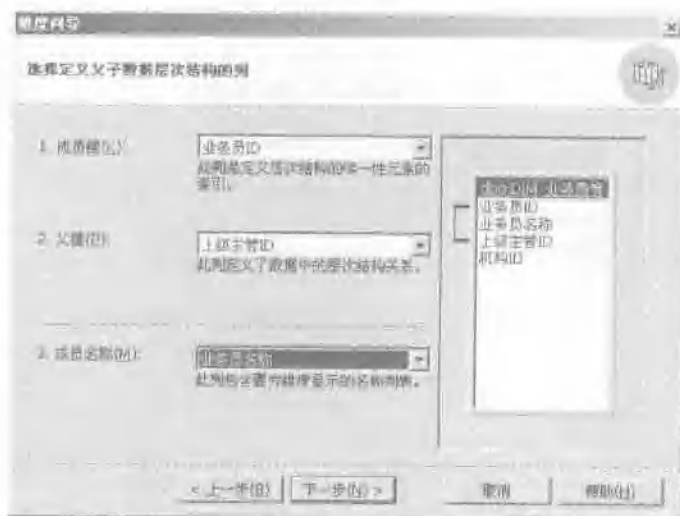


图 4.5 “选择定义父子数据层次结构的列”对话框

第4步 单击“下一步”按钮,出现“选择高级选项”对话框,在该对话框中选择“有数据的成员”,单击“下一步”按钮,如图4.6所示。

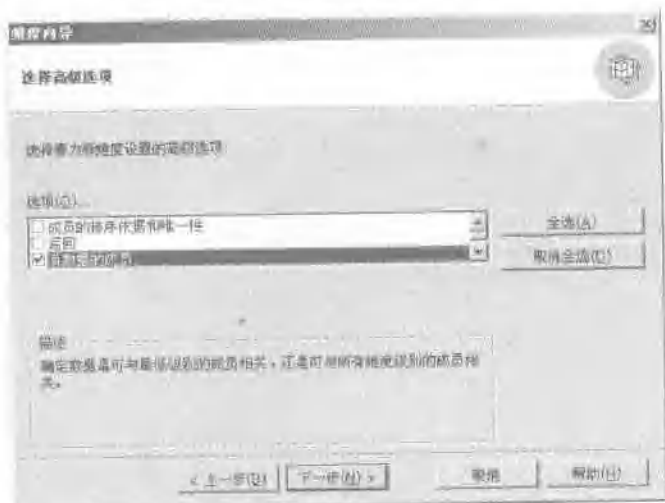


图 4.6 “选择高级选项”对话框

第5步 出现如图4.7所示的“设置带有数据属性的成员”对话框,选中“非叶成员有相关的数据”,并选择“数据成员可见”,单击“下一步”按钮。

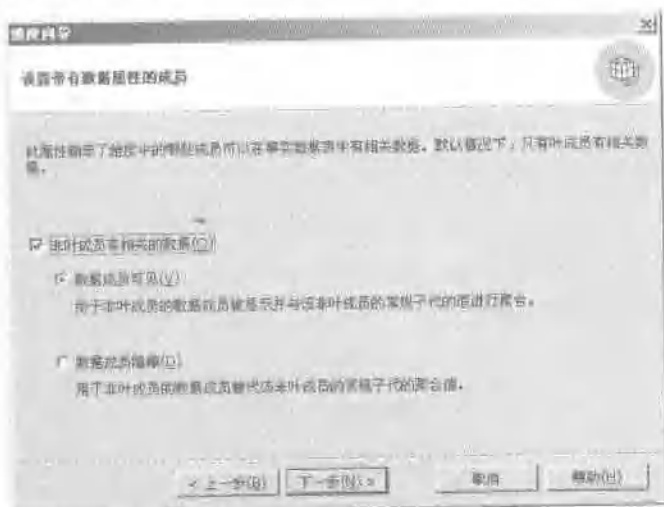


图 4.7 “设置带有数据属性的成员”对话框

第6步 进入“完成”对话框,在“维度名称”框中输入“业务员等级”,如图4.8所示。



图 4.8 输入“业务员等级”

第7步 在“多维数据集编辑器”窗口中,将“业务员等级”维度添加到“保单资源分析”多维数据集中,再将“DIM_业务员等级关系”表的“业务员ID”字段与“DIM_中间表”的“业务员ID”字段连接起来,如图4.9所示。



图 4.9 连接表

第8步 处理“保单资源分析”多维数据集,方法与之前的处理方法相同。

注意:“选择高级选项”对话框中“有数据的成员”的意义是指非叶(Non-Leaf)成员包含要分析的数据,而非叶成员是指层次结构中非最底层的成员,具有扮演“汇总结点”角色的成员,在本书中,就是指业务员等级关系中的那些上级主管。案例中,上级主管也是有保单业绩的,因此要选中“有数据的成员”这一选项。在“设置带有

数据属性的成员”对话框中选择“数据成员可见”的含义是:与非叶成员有关的数据参与聚合分析。在本书案例中,就是指对上级主管的保单信息也要进行统计和分析。如果非叶成员不包含要分析的数据,则不需要选中这两项。

4.4.2 计算成员的建立

在“保单资源分析”多维数据集中,新建度量值计算成员:“实际收入”=“新单保费”+“续期保费”-“退还保费”;“实际收入环比”=“当前单位时间实际收入”/“上个单位时间实际收入”,并将“实际收入”的格式设为“小数点后的一位数值格式”,将“实际收入环比”的格式设为“百分比”。

建立一个维度计算成员,位于“区域_机构_业务员”维度中,名称为“专业业务人员”,代表业务员中专门从事业务的人员,去除了其他工号、团体代理、银行保险、营销等部分。

第1步 打开“保单资源分析”的“多维数据集编辑器”窗口,在“计算成员”上单击鼠标右键,选择“新建计算成员”,出现如图4.10所示的“计算成员生成器”对话框。



图 4.10 “计算成员生成器”对话框

第2步 在“成员名称”框中输入“实际收入”。在“值表达式”框中输入计算公式,或从下面列表中选取相应的度量值和运算符: [Measures]. [新单保费] + [Measures]. [续期保费] - [Measures]. [退还保费]。单击“检查”按钮,如果提示“语法正确”,则单击“确定”按钮。

第3步 用同样方法设置计算成员“实际收入环比”,计算“实际收入环比”的计算公式为: $\text{if}([\text{时间}]. \text{CurrentMember} . \text{PrevMember} \text{ is null, null, } ([\text{时间}]. \text{CurrentMember} , [\text{Measures}]. [\text{实际收入}]) / ([\text{时间}]. \text{CurrentMember} . \text{PrevMember} , [\text{Measures}]. [\text{实际收入}]))$ 。

第4步 在建立计算成员后,回到“多维数据集编辑器”窗口,选择“实际收入”,在下方“属性”区域中选择“高级”选项卡,在“Format String”框中选择显示格式,并用同样方法设置“实际收入环比”,如图4.11所示。



图4.11 “高级”选项卡

第5步 建立维度计算成员,在“计算成员”上单击鼠标右键,进入“计算成员生成器”对话框,在“父维度”框中选择“区域_机构_业务员”,在“成员名称”框中输入“专业业务人员”,另一个维度计算成员的建立过程相同,如图4.12所示。

“专业业务人员”的计算公式: [区域_机构]. [业务员]. [区域分类]. & [城区营业部] + [区域_机构]. [业务员]. [区域分类]. & [大客户业务部]。

“专业业务人员”的计算成员公式也可以写成: [区域_机构]. [业务员]. [所有区域_机构] - [区域_机构]. [业务员]. [区域分类]. & [郊区团体部]。



图 4.12 构建维度计算成员

4.4.3 成员属性的建立

为“险种_投保人类型分类”维度中的“险种名称”成员设置“成员属性”——“险种代码”，然后将“dim_险种类别分类”表中的“险种分类”和“险种标准赔付率”设为“险种名称”成员的“成员属性”。

第1步 在“共享维度”下的“险种_投保人类型分类”上单击鼠标右键,进入如图 4.13 所示的“维度编辑器”窗口,选择“险种名称”,在“成员属性”上单击鼠标右键,选择“新建成员属性”,出现如图 4.14 所示的“插入成员属性”对话框。

第2步 选择“险种代码”,单击“确定”按钮。

第3步 回到“险种_投保人类型分类”的“维度编辑器”窗口,选择“插入”→“表”菜单命令,出现“选择表”对话框。在左面的“表”下选择“dim_险种类别分类”,单击“添加”按钮,如图4.15所示。单击“关闭”按钮回到“维度编辑器”窗口。

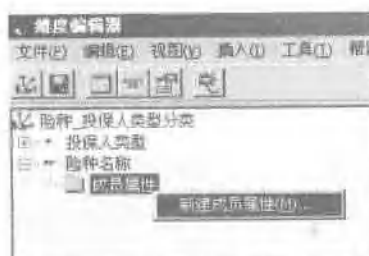


图 4.13 “维度编辑器”窗口



图 4.14 “插入成员属性”对话框

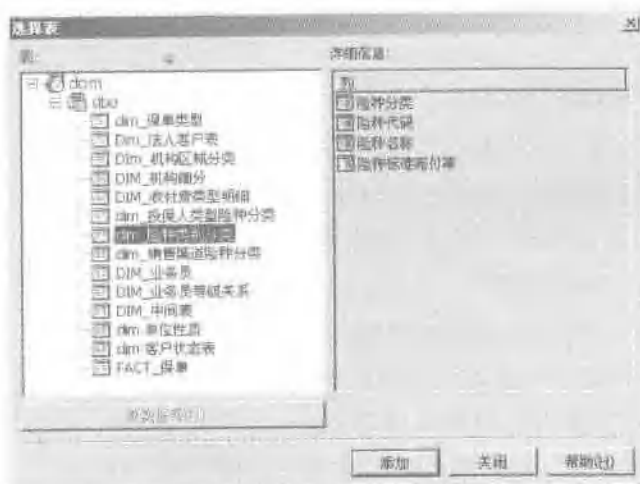


图 4.15 “选择表”对话框

第4步 系统自动连接两个表,如果关系不正确,可以删除连线,此处通过“险种代码”将两个表连接起来,可以用拖动方式将“险种分类”和“险种标准赔付率”拖入到左边“险种名称”下的“成员属性”中,如图4.16所示。



图 4.16 添加成员属性

第5步 用同样的方法,再在“险种_投保人类别分类”的“维度编辑器”窗口中插入表“dim_销售渠道险种分类”,将“销售渠道类型”也加入到“险种_投保人类别分类”维度中“险种名称”级别的“成员属性”中。

4.4.4 虚拟维度的建立

以“险种_投保人类别分类”的“险种名称”成员属性为基础,建立名为“险种分类_标准赔付率”的虚拟维度。以“dim_销售渠道险种分类”为列的提供者,建立“销售渠道类型虚拟维度”。建立“虚拟维度分析”的多维数据集,分析保单。

第1步 在“共享维度”上单击鼠标右键,选择“新建维度向导”,出现“选择维度的创建方式”对话框,选择“虚拟维度”,如图4.17所示。

第2步 单击“下一步”按钮,出现“选择带有属性成员的维度”对话框,选择“险种_投保人类别分类”,如图4.18所示。

第3步 单击“下一步”按钮,出现“选择虚拟维度的级别”对话框,选择“险

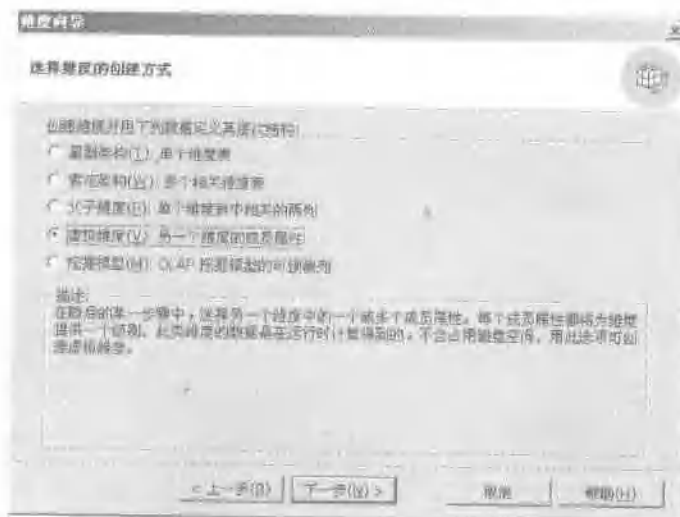


图 4.17 “选择维度的创建方式”对话框

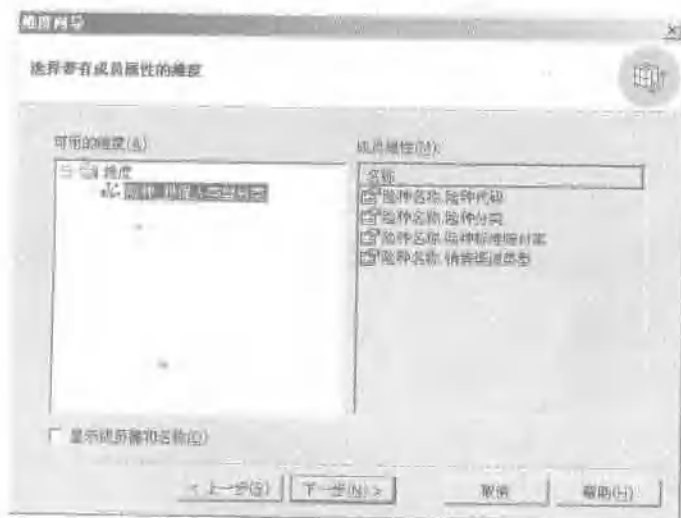


图 4.18 “选择带有属性成员的维度”对话框

种分类”和“险种标准赔付率”，如图 4.19 所示。

第4步 单击“下一步”按钮，出现“选择高级选项”对话框；再单击“下一步”按钮，出现如图 4.20 所示的“移动级别”对话框。单击“否”按钮，出现“完成”对话框，如图 4.21 所示。

第5步 在“维度名称”框中输入“险种类别”，在“层次结构名称”框中输入“标准赔付率”，单击“完成”按钮。

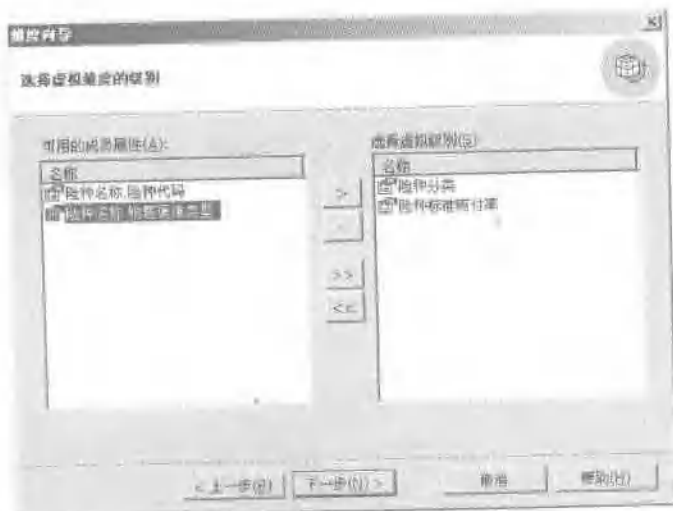


图 4.19 “选择虚拟维度的级别”对话框

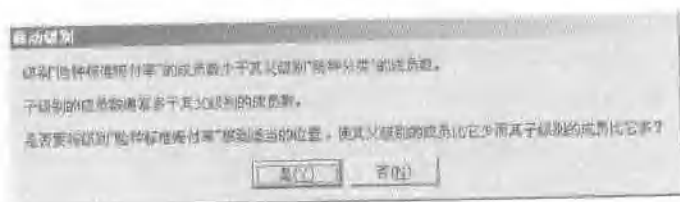


图 4.20 “移动级别”对话框



图 4.21 虚拟维度创建完成

第6步 在“共享维度”上单击鼠标右键,选择“新建维度向导”进入“选择维度的创建方式”对话框,因为本例需要的列在一个表中,所以选择“星型架构:单个维度表”,单击“下一步”按钮。

第7步 接下来的步骤与建立一般维度相同,选择“dim_销售渠道险种分类”作为维度表,选择“销售渠道类型”列作为维度的层次。一直到进入“选择高级选项”对话框,选择“可更改维度”,如图4.22所示。

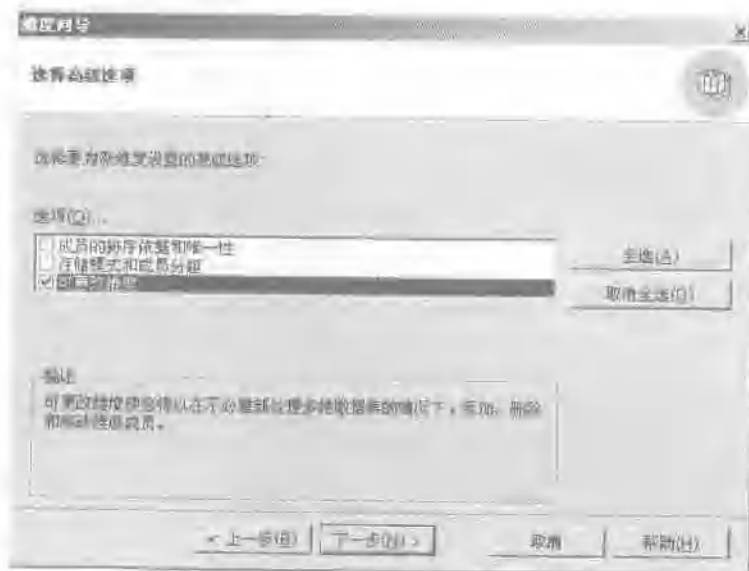


图4.22 “选择高级选项”对话框

第8步 单击“下一步”按钮,出现“设置可更改属性”对话框,选择“是,新建的维度可更改”,如图4.23所示。

第9步 单击“下一步”按钮,出现“完成”对话框,在“维度名称”框中输入“销售渠道类型虚拟维度”,单击“完成”按钮。

第10步 在“销售渠道类型虚拟维度”的“属性”下的“高级”选项卡中,选择“Depend On Dimension”,选择提供列的“险种_投保人类型分类”维度。选择“Virtual”属性,选择“True”,如图4.24所示,保存维度后退出。

第11步 建立“虚拟维度分析”多维数据集的步骤与前面介绍的相同,这里不再赘述。该多维数据集用到的事实表为“FACT_保单”,选择适当的度量值:维度有“时间”、“险种_投保人类型分类”、“险种分类_标准赔付率”、“销售渠道类型虚拟维度”;将“DIM_中间表”中的“险种_ID”与“DIM_投保人类型险种分类”表的

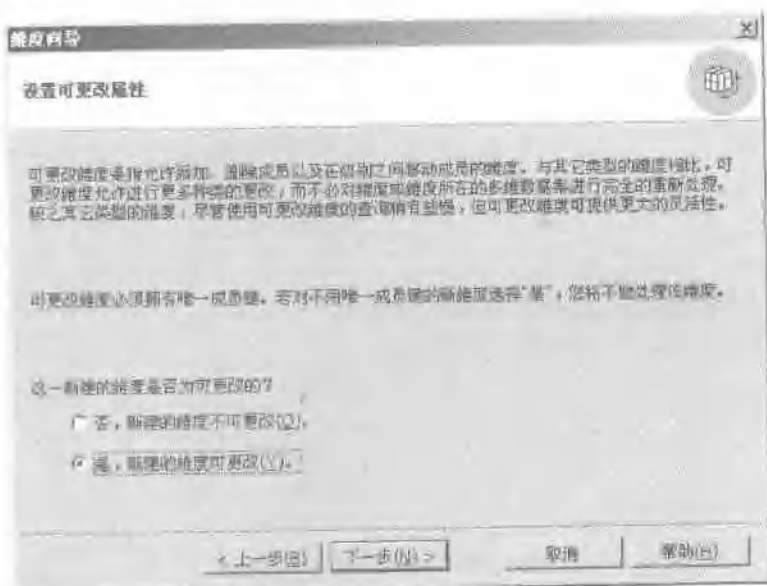


图 4.23 “设置可更改属性”对话框



图 4.24 修改虚拟维度属性

“险种代码”相连。

4.5 扩展试验

下面是本章的扩展实验,以供有兴趣、有能力的读者进行巩固提高。

1. 建立父子维度

利用“DIM_业务员等级关系”表,新建一个名为“管理层次”的共享父子维度,其中,成员键为“业务员 ID”,父键为“上级主管 ID”,成员名称为“业务员名称”;再以“DIM_业务员等级关系”表为例,建立以“上级主管 ID”为父层,“业务员 ID”为子层,选择“有数据的成员”和“非叶成员有关的数据”中的“数据成员可见”,建立私有父子维度“业务员等级”,将该私有父子维度加入“法人客户价值分析保单数量”多维数据集,并对多维数据集进行处理,分析每一位成员(主管和部属)及其隶属的成员的保单业绩。

2. 建立计算成员

在“法人客户价值分析保单数量”多维数据集中,分别通过“DIM_法人客户表”和“DIM_单位性质”表,新建私有维度“客户”和“单位性质”,其层次关系依次为“开户年”、“客户名称”和“客户 ID”,“单位性质”维度,仅有一个层次为“单位性质名称”。此时,应注意两表的连接(“单位性质代码”),进而对多维数据集进行处理。然后,新建度量值计算成员“客户数量”,这里用到 distinctcount 函数,建立父维度为“单位性质”的计算成员“非公有制”,它是四个单位(个人、个体、私营、港澳台地区三资和外国三资)的简单求和。

3. 建立成员属性

为“险种_险种类型分类”维度中的“险种名称”成员设置“成员属性”——“险种代码”和“险种分类”。

再将“DIM_投保人类型险种分类”表中的“投保人类型”加入到“险种_险种类型分类”的“险种名称”的“成员属性”中。

4. 新建虚拟维度

以“险种_险种类型分类”的“险种名称”成员属性为基础,建立名为“险种.类型”的虚拟维度,其维度级别依次为“险种代码”和“险种分类”。以“DIM_投保人

类型险种分类”为列的提供者,建立“雪花架构”的“投保人类型”的标准维度,修改其属性,使之成为一个“虚拟维度”。

5. 创建复杂多维数据集

用数据源表“DIM_法人客户表”、“DIM_中间表”、“DIM_险种类别分类”、“DIM_投保人类型险种分类”、“DIM_客户状态表”,建立名称为“法人客户价值——虚拟维度分析”的多维数据集。其中,把“DIM_中间表”作为事实表,维度有“险种_险种类别分类”、“投保人类型”、“险种_类型”、“开户年_法人客户”和“生效时间”,有“投保人数”和“保险金额”度量值,来对法人客户价值中虚拟维度进行分析。

本章小结

父子维度、虚拟维度等复杂维度的建立也可以通过维度建立向导来创建,并将其设为共享维度。在建立多维数据集的时候,可以应用这些现成的共享维度。注意:在建立父子维度的时候,父成员与子成员之间要有一定的逻辑关系,比如本实验中的上级主管与员工,是一种隶属关系。

本章要掌握的知识点如下:

- ◆ 父子维度、虚拟维度的建立过程。
- ◆ 计算成员的建立过程。
- ◆ 计算成员既可以作为维度,也可以作为度量值。
- ◆ 成员属性的建立。
- ◆ 使用多维数据集编辑器对多维数据集进行编辑和管理,可以对维度和度量值进行新建、删除和更改属性。



第5章

多维数据集的分区、 虚拟多维数据集

本章引言

通过前4章的学习,读者可以建立和编辑简单的星型架构、雪花架构、父子维度和虚拟维度,编辑维度成员属性,增加、删除与修改度量值,新建和编辑简单的计算成员,进而完成对简单多维数据集和复杂多维数据集的建立和编辑,并完成对所建立的多维数据集进行存储设计和成功处理。

本章从多维数据集的处理效率方面考虑,介绍多维数据集的三种不同处理方法,介绍分区与合并的概念、原理、实现方法和意义。另外,在简单多维数据集的基础上,介绍虚拟多维数据集的概念、建立、编辑和功能,使读者对于多维数据集有一个全面而清晰的认识。

本章重点

- 多维数据集的分区。
- 多维数据集分区的合并。
- 虚拟多维数据集的概念。
- 虚拟多维数据集的建立。

5.1 实验预备知识

为了提高多维数据集的处理效率,除了要善于运用前面章节所介绍的 OLAP 数据存储方式技术以及后面的章节将介绍的“多维数据集使用分析与基于使用的优化”方法外,Analysis Manager 还可以把多维数据集分为多个数据块,并且每一个分区可以使用不同的模式存储,采取不同的汇总策略。

5.1.1 多维数据集分区

对于已经建立的多维数据集,至少包含一个分区,并且该分区中包含原有多维数据集的数据。当用户为多维数据集建立一个新的分区后,新的分区会加进该多维数据集现有的分区集中。因此,在为多维数据集建立分区时,应保证所有的分区的数据区域没有重复。

对多维数据集分区后,可以使用汇总百分比较高的 MOLAP 结构存储,以产生快速的响应速度;使用 HOLAP 存储方式来存储不常被查询的数据,可以减少存储的需求;使用汇总百分比较低的 ROLAP 模式来分区可以节省存储空间。

应用多维数据集时,不管是否分区,都不影响产生的结果。不论查询常用的或不常用的数据,整个多维数据集的有效查询都将被逐一执行,但是返回不常用数据的查询比返回常用的数据查询花费更多的时间。

每个分区都有一个数据源,这与一般的多维数据集相同,这个数据源可以与分区的多维数据集的数据源相同,也可以不同。若数据源不同,新数据源的结构应该与多维数据集中使用的结构相同,也就是说,该分区的事实数据表和维度表包含的列应该与多维数据集中原有的列相同。

在数据存储部分,每个分区都可以将汇总数据存储在设计分区所在的服务器上,也可以存储在另一台服务器上,即分区有“本地”和“远程”的区别。

5.1.2 多维数据集分区的合并

对于已经被分区的多维数据集,还可以对分区进行合并。为了确保分区以后可以成功合并,保证合并后数据的正确性,分区应该遵循以下准则:

分区数据必须位于相同的多维数据集中,具有相同的事实数据表或相同的结构,以相同的模式存储,有相同的汇总设计。注意:远程的分区数据只能与定义在同一远程服务器上的其他数据进行合并。

5.1.3 虚拟多维数据集

在前面的章节中,已对“虚拟维度”做了介绍,它是以维度数据表中字段成员的“属性”来建立的维度。但是,虚拟多维数据集并不是因为虚拟维度而成为虚拟多维数据集。它的概念类似于关系数据库中“视图”(View)的概念。视图只是一个数据结构的表达,本身并没有实际数据,当使用视图时,根据数据结构从原数据表中产生所需要的数据。

虚拟多维数据集,是一个可以获取现有单一或多重多维数据集而组成的一个新的多维数据集逻辑定义结构,通过现有的多维数据集的维度和度量值来建立虚拟多维数据集。虚拟多维数据集不需要实体存储空间,因为它只保存逻辑定义结构而不真正保存多维数据集的组件和数据。虚拟多维数据集主要应用于多维数据集的安全性管理。Analysis Manager 可以通过设定角色类别来限定对多维数据集的访问权限,同时,如果某些多维数据集中包含具有绝密性质的数据,不适合所有用户的访问,可以在现有的多维数据集的基础上,建立虚拟多维数据集,删除机密信息,并配合角色类别的安全性设置,授权给不同的用户使用,从而保证多维数据集中数据的安全。

5.2 实验目的

通过本实验,读者应了解多维数据集分区的作用,理解虚拟多维数据集的概念与作用,掌握多维数据集分区、虚拟多维数据集的建立。

5.3 实验要求

掌握多维数据集分区的建立和编辑方法,掌握虚拟多维数据集的建立方法。

下面将介绍多维数据集的分区、合并以及虚拟多维数据集的建立等相关操作的具体实现步骤。

5.4 实验内容与步骤

实验的主要内容是练习多维数据集分区的建立、编辑方法,以及虚拟多维数据集的建立方法。具体实验设置是:为多维数据集“保单资源分析”中2003年的数据建立一个单独的区块,对该区块进行存储设计;再将这个区块加入到“保单资

源分析”区中。然后,以多维数据集“保单资源分析”和“虚拟维度分析”为基础,建立一个虚拟多维数据集“保单资源公用分析”,使其度量值只包含“预收保费”、“新单保费”、“续期保费”和“退还保费”,仅有“时间”和“区域_机构_业务员”两个维度。下面介绍具体的实验步骤。

5.4.1 多维数据集分区的建立

对多维数据集“保单资源分析”中 2003 年的数据建立分区,并将其存储方式设计为 ROLAP,汇总百分比为 50%,名称为“历史”。编辑“保单资源分析”多维数据集,使它不包含“历史”数据。最后,将“历史”分区合并到“保单资源分析”多维数据集中。

第 1 步 如图 5.1 所示,在“Analysis Manager”窗口中,在“多维数据集”下选择“保单资源分析”,在“分区”上单击鼠标右键,选择“新建分区”,出现“分区向导”对话框。

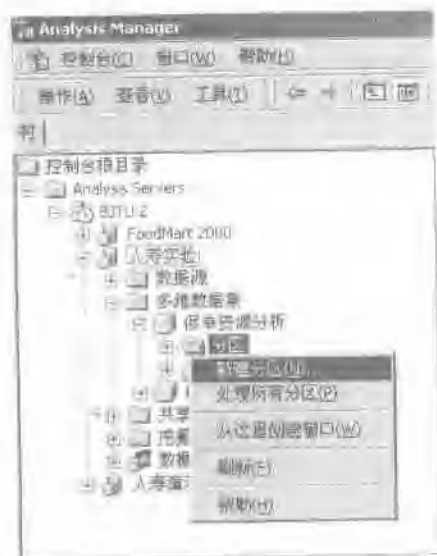


图 5.1 新建分区

第 2 步 单击“下一步”按钮,出现如图 5.2 所示的“指定数据源和事实数据表”对话框。如果系统默认的数据源和事实表不正确,也可以单击“更改”按钮,进入“选择事实数据表”对话框进行修改。本范例不修改。

第 3 步 单击“下一步”按钮,出现“选择数据切片”对话框,在“成员”下的“所有时间”下选择“2003”,如图 5.3 所示。

第 4 步 单击“下一步”按钮,出现“指定分区类型”对话框,可以根据需要选

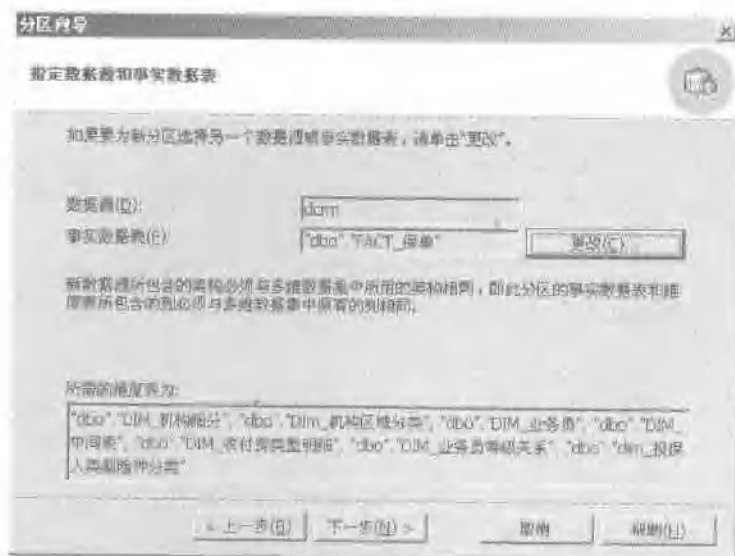


图 5.2 “指定数据源和事实数据表”对话框

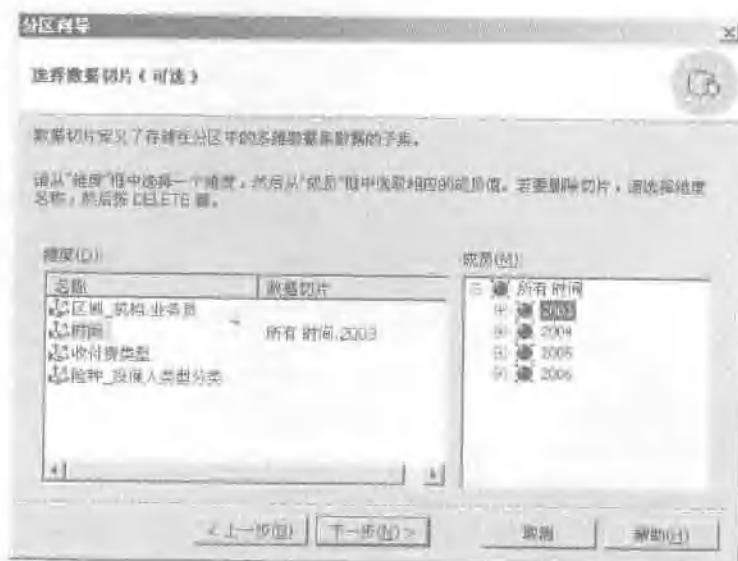


图 5.3 “选择数据切片”对话框

择“本地”或“远程”, 本例中选择“本地”, 如图 5.4 所示。

第5步 单击“下一步”按钮, 出现“完成分区向导”对话框。在“分区名称”框中输入“历史”, 并选择“现在就设计分区的聚合”, 如图 5.5 所示。

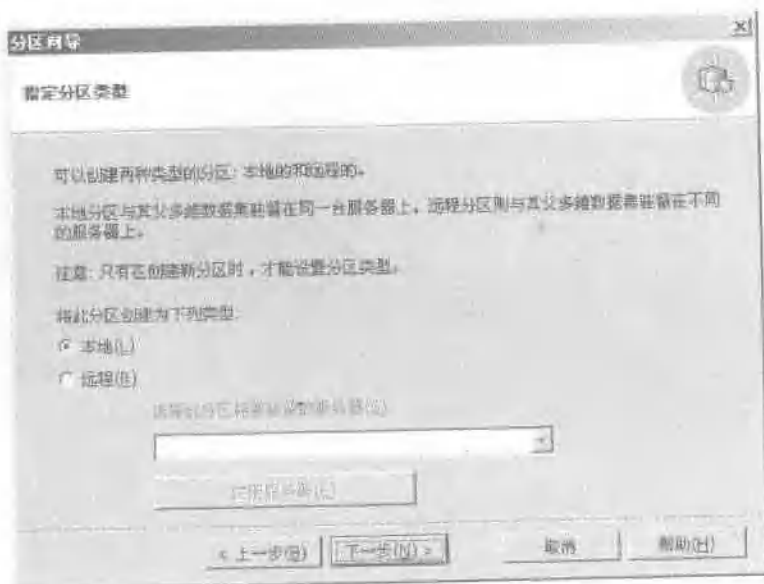


图 5.4 “指定分区类型”对话框

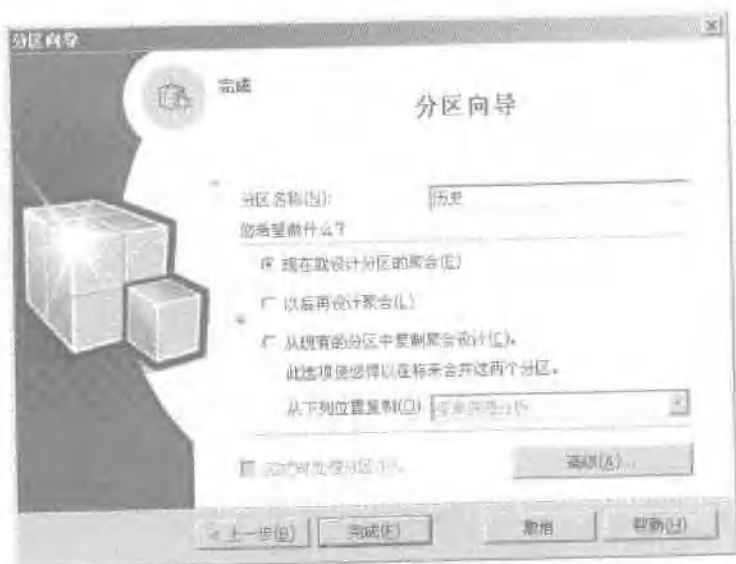


图 5.5 “完成分区向导”对话框

第 6 步 单击“完成”按钮,出现“存储设计向导”对话框,如图 5.6 所示。

第 7 步 单击“下一步”按钮,出现“选择数据存储类型”对话框,选择“ROLAP”存储模式,如图 5.7 所示。

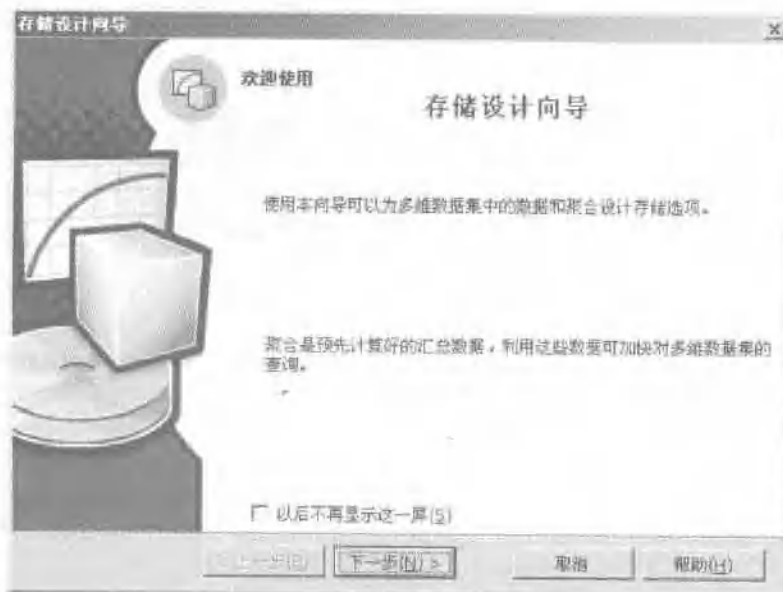


图 5.6 “存储设计向导”对话框

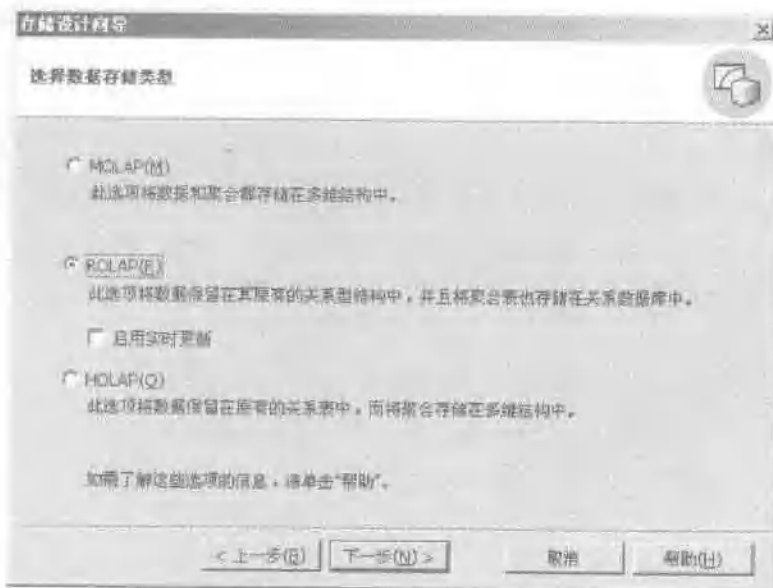


图 5.7 “选择数据存储类型”对话框

第8步 单击“下一步”按钮，出现“设置聚合选项”对话框，选择“性能提升达到”“50%”，如图 5.8 所示。

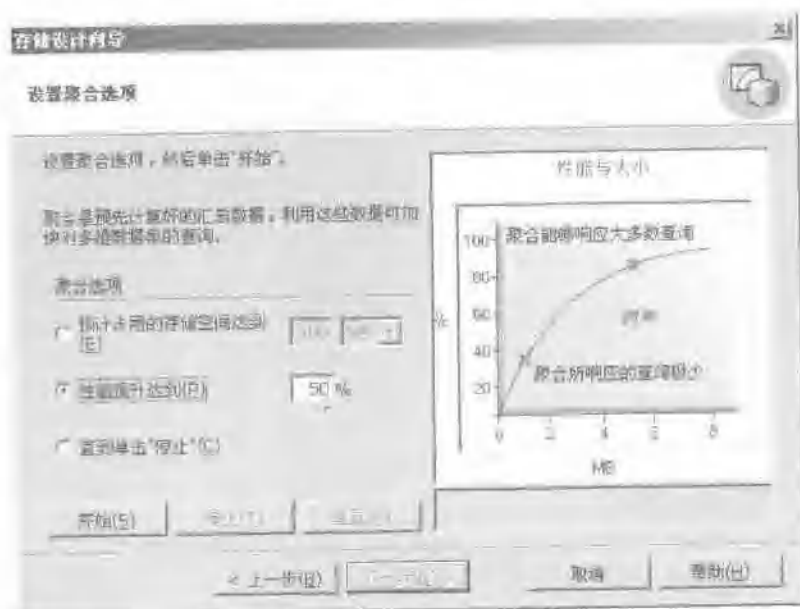


图 5.8 “设置聚合选项”对话框

第9步 单击“开始”按钮，系统处理完后，单击“下一步”按钮，出现“完成”对话框，单击“完成”按钮。

5.4.2 多维数据集分区的应用

进行分区的时候，必须正确地建立分区并加以管理，以避免数据的不一致性与不正确性。当用户从分区中获取希望的摘要数据时，若同一数据出现在不同的分区，则系统会当成不同的数据来获取，而造成不正确的数据，也将反馈给用户过多重复的数据。

在建立分区时，用户必须建立合理的筛选条件，以确保多维数据集的分区包含不重复的正确的数据。对于前面所涉及的“历史”的分区，在原有的“保单资源分析”分区中，就不能包含“历史”分区。此时，需要对“保单资源分析”的分区进行编辑。

第1步 在“保单资源分析”中删除“历史”的数据。在“保单资源分析”多维数据集中，选择“分区”中的“保单资源分析”分区，单击鼠标右键，选择“编辑”，出现“指定数据源和选择事实数据表”对话框。单击“下一步”按钮，进入“选择数据切片”对话框，不选择任何维度。单击“下一步”按钮，选择“本地”按钮。单击“下

一步”按钮,直到出现“完成”对话框。单击“高级”按钮,出现“高级设置”对话框。在“筛选语句”区中加入“时间”<>'2003',如图 5.9 所示。

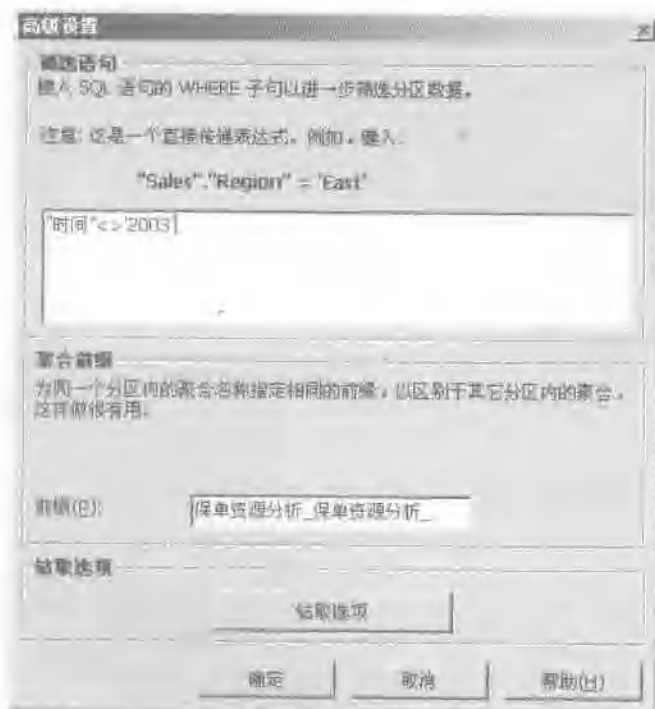


图 5.9 “高级设置”对话框

第2步 单击“确定”按钮,回到“完成”对话框。单击“完成”按钮,随后的处理与前面相同。

5.4.3 多维数据集分区的合并

第1步 合并分区。在“Analysis Manager”窗口中,在“历史”分区上单击鼠标右键,选择“编辑”,出现对话框,连续单击“下一步”按钮,直到出现“完成分区间导”对话框,选择“从现有分区中复制聚合设计”,在“从下列位置中复制”框中选择“保单资源分析”,选中“完成时处理分区”,如图 5.10 所示。

第2步 单击“完成”按钮,回到“Analysis Manager”窗口,再次选择“历史”,单击鼠标右键,选择“合并”,出现如图 5.11 所示的“合并分区”对话框,单击“合并”按钮,完成合并。(注意:一定要在“历史”分区上合并,如果在“保单资源分析”上合并就会将“保单资源分析”分区合并到“历史”分区里。)



图 5.10 修改分区聚合设计

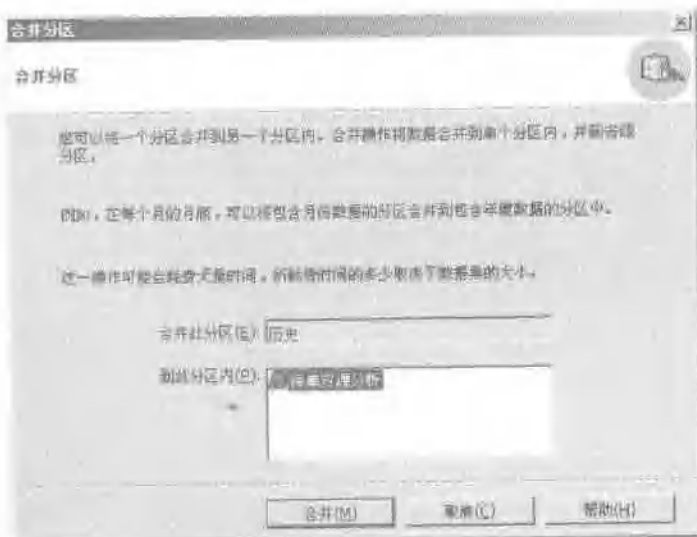


图 5.11 “合并分区”对话框

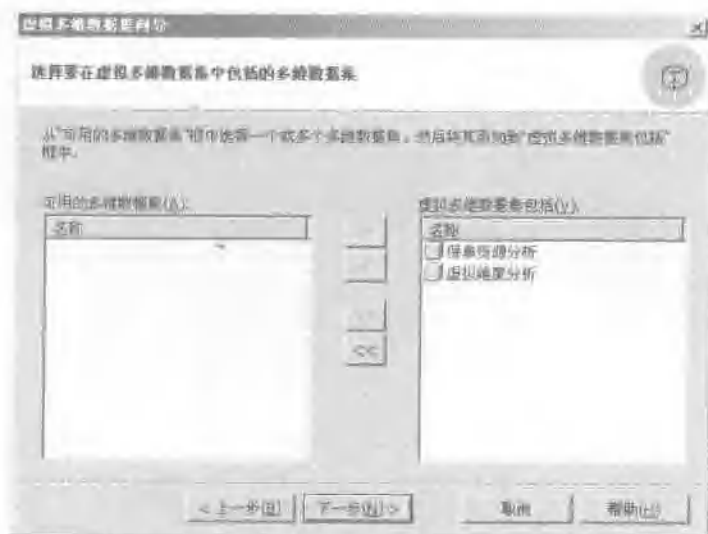
5.4.4 虚拟多维数据集的建立

以“保单资源分析”多维数据集为基础，建立一个“保单资源公用分析”虚拟多维数据集，只包含“预收保费”、“新单保费”、“续期保费”和“退还保费”等维度。

第1步 在“Analysis Manager”窗口中，在“多维数据集”上单击鼠标右键，选择“新建虚拟多维数据集”，出现“虚拟多维数据集向导”对话框，如图 5.12 所示。



第2步 单击“下一步”按钮,出现“选择要在虚拟多维数据集中包括的多维数据集”对话框,在“虚拟多维数据集包括”下选择“保单资源分析”和“虚拟维度分析”,如图 5.13 所示。



第3步 单击“下一步”按钮,出现“选择虚拟多维数据集的度量值”对话框,在“所选的度量值”下选择“预收保费”、“新单保费”、“续期保费”和“退还保费”。

为度量值,如图 5.14 所示。

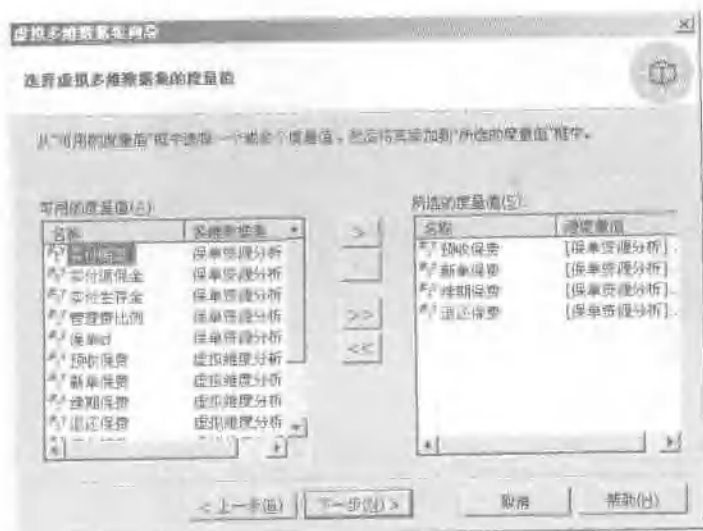


图 5.14 “选择虚拟多维数据集的度量值”对话框

第 4 步 单击“下一步”按钮,出现“选择虚拟多维数据集的维度”对话框,在“所选的维度”下选择“时间”和“区域_机构_业务员”,如图 5.15 所示。

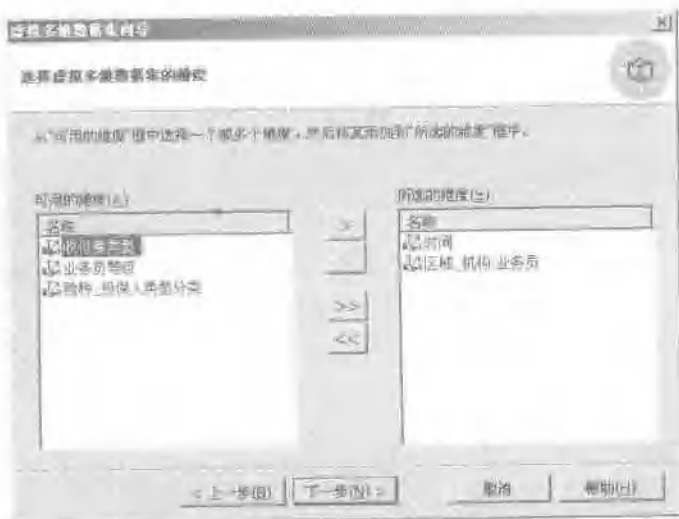


图 5.15 “选择虚拟多维数据集的维度”对话框

第 5 步 单击“下一步”按钮,出现如图 5.16 所示的“完成虚拟多维数据集向

导”对话框,在“虚拟多维数据名称”框中输入“保单资源公用分析”,单击“完成”按钮。



图 5.16 “完成虚拟多维数据集向导”对话框

5.5 扩展实验

为了加深对所学知识和技术的掌握,本章专门设计了如下 4 个扩展实验,以供有能力的读者得到进一步的提高。

1. 多维数据集的分区

对“法人客户价值分析保单数量”多维数据集中的“保单类型”表中的类型为“两可”的数据,建立分区,设计其存储方式为 ROW,性能提高到 50%。然后,进行处理,名称为“保单类型两可”。编辑“法人客户价值分析保单数量”多维数据集,使它不包含“保单类型两可”。

2. 编辑分区与筛选条件设置

对于前面所涉及的“保单类型两可”的分区,在原有的“法人客户价值分析保单数量”分区中,不能包含“保单类型两可”分区。此时,需要对“法人客户价值分析保单数量”的分区进行编辑,书写筛选语句以删除“法人客户价值分析保单数量”中的“保单类型两可”的数据。

3. 分区的合并

将“保单类型两可”分区合并到“法人客户价值分析保单数量”分区中。

4. 建立虚拟多维数据集

在“法人客户价值分析保单数量”多维数据集的基础上,建立一个“业务经理专用”虚拟多维数据集,选择“法人客户价值分析保单数量”和“法人客户价值——虚拟维度分析”多维数据集,使之只包含“投保人数”和“保单数量”两个度量值,维度有“生效时间”、“保单类型”、“客户”和“业务员等级”。

本章小结

通过本章内容的学习,可以使读者掌握多维数据集分区与合并的基本概念、基本原理和注意事项,独立完成简单的多维数据集分区与合并工作,能够对不同的分区进行不同的存储设计和编辑处理,设置筛选条件并完成分区的合并,同时,了解虚拟多维数据集的概念和意义,可以根据本章内容在已有的多维数据集基础上根据安全性需求建立合理的虚拟多维数据集,以保证数据的安全性。另外,读者可以通过扩展实验,深化对上述知识点的理解和掌握,提高实际操作水平。



第6章

MDX 语言应用

本章引言

在前面章节中,曾经多次运用到 MDX 语法,如“计算成员”、“维度安全性自定义规则”、“命名集”、“对策”、“计算单元”、“自定义汇总公式与自定义成员公式”等。在运用的过程中,通常会使用“MDX 生成器”,以帮助我们建立 MDX 公式,但 MDX 的内涵究竟是什么?在运用“Analysis Manager”的高级过程中,很多时候都必须使用到 MDX 公式,因此本章将为读者详细介绍 MDX 语言的应用。

本章重点

- MDX 语言的概念与语法规则。
- MDX 语言与 SQL 语言的比较。
- MDX 的处理目标表示方式。
- MDX 查询语句的使用。
- MDX 函数的使用。

6.1 实验预备知识

顾名思义,MDX(Multidimensional Expressions,多维表达式)是支持多维度对象的数据定义与运用的一种语言。MDX在许多方面与SQL(Structured Query Language,结构化查询语言)语言相类似,但MDX并不是SQL语言的延伸。简单地说,SQL所处理的对象,经常是数据库中的原始表与其中的数据;而MDX所处理的对象则是经过OLAP处理的多维数据集。两者最大的相似之处是,多用于处理“查询”。

在语法的结构上,MDX与SQL在查询时,两者经常都包含“筛选对象”(select)、“数据源”(from子句)以及必要的“筛选条件”(where子句),甚至也可以包含“排序”(asc或desc)。这些关键字结合了多维数据集的表达式,可用来从分析的多维数据集中挖掘指定数据;而也如SQL语法一样,MDX语法也包含功能强大的函数,以协助数据处理与挖掘,并提供用户自定义函数功能来扩充MDX的运用范畴。

与SQL一样,MDX也提供“数据定义语言”(DDL)来管理数据结构,包含建立、删除多维数据集、维度、度量值以及其他附属对象的命令。

6.2 实验目的

本章通过对MDX语言的模拟和运行,使读者初步掌握MDX这种支持多维度对象的数据定义和应用语言。读者应掌握MDX函数的主要处理目标及其表达式,主要包括成员的表达式、坐标轴与子多维数据集的表达式、注释的表达式以及建立计算成员等。通过本章实验,读者最终应掌握运用MDX进行查询的基本语法和函数。

6.3 实验要求

本章要求读者学会使用“MDX示例应用程序”,理解和使用成员表达式,理解和使用子多维数据集表达式,学会使用注释表达式,学习空数据集的处理,学会建立计算成员,能够建立命名集,学习使用crossjoin等函数。

6.4 实验内容与步骤

本章实验的主要内容包括成员的表达式、坐标轴与子多维数据集的表达式、注释的表达式以及建立计算成员等。

6.4.1 使用“MDX 示例应用程序”建立连接

本节主要建立使用 MDX 表达式的环境,这部分工作在“MDX 示例应用程序”中进行。

第1步 了解 MDX 的基本语法结构。MDX 与标准 SQL 语句有本质的差别,它不是针对传统关系型数据库中二维表的查询语句,而是针对多维数据集的数据定义和操作语言。但是在语法结构上,又与 SQL 语句有很多的相似之处。二者都包含“筛选对象”(select 子句)、“数据源”(from 子句)、可选的筛选条件(where 子句)。与 SQL 语句不同,MDX 的筛选对象是多维数据集子集。多维数据集子集由维度中的成员表达式构成,一个 MDX 查询中最多可以指定 128 个坐标轴。MDX 的数据源是多维数据集。MDX 查询中的条件也是针对成员表达式的。

第2步 打开“MDX 示例应用程序”。如图 6.1 所示,单击“开始”→“Microsoft SQL Server”→“Analysis Services”→“MDX 示例应用程序”菜单选项,进入“MDX Sample Application (MDX 示例应用程序)”窗口。



图 6.1 选择菜单命令进入“MDX Sample Application”窗口

第3步 连接服务器。进入“MDX Sample Application”窗口之后,首先弹出如图 6.2 所示的“Connect”对话框,在该对话框中建立服务器的连接。在“server”框

中输入安装 Analysis Services 的计算机名(完成本实验的计算机名为“BJTU-2”,读者需输入自己的计算机名),在“Provider”框中输入“MSOLAP”,单击“OK”按钮。



图 6.2 “Connect”对话框

第 4 步 连接数据源和多维数据集。连接好服务器后,进入“MDX Sample Application”窗口。在该窗口中,进行数据源和多维数据集的连接。在“DB(数据源名称)”框中选择要连接的数据源,此处建立的数据源名称为“人寿实验”,如图 6.3 所示”。在“Cube(多维数据集立方体)”框中选择要连接的立方体,此处为“保单资源分析”,如图 6.4 所示。

6.4.2 使用成员表达式

进入“MDX Sample Application”窗口并建立好各种连接以后,就可以在此窗口中输入 MDX 语句。该窗口共分为 4 个主要部分:上部是工具栏,中部是 MDX 语句的输入界面,下部左侧是多维数据集的树形结构,右侧是各种 MDX 函数的树形结构。

第 1 步 使用成员表达式。“成员”(member)是“维度”中的一个项目,当我们在 MDX 中描述多维数据集的单元格时,“成员”是最低的参照层次。

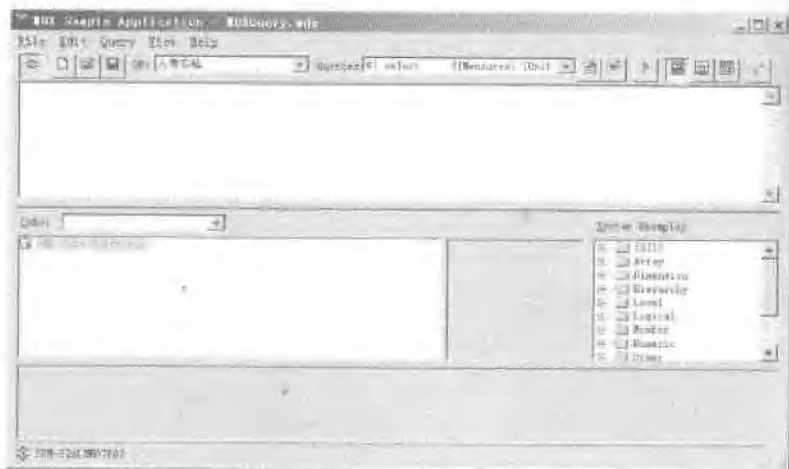


图 6.3 “MDX Sample Application”窗口



图 6.4 选择“保单资源分析”

例如,我们已经建立的时间维度——“时间”,该维度共有 4 个层次:年、季度、月、日。可以使用如下表达式来代表“年”与“季度”中的成员:

[时间].[年]

[时间].[季度]

成员的具体的指定方法有两种,例如在“时间”中,针对年层次为 2003 年的成员,可以使用如下表示:

合。查询结果如图 6.6 所示。

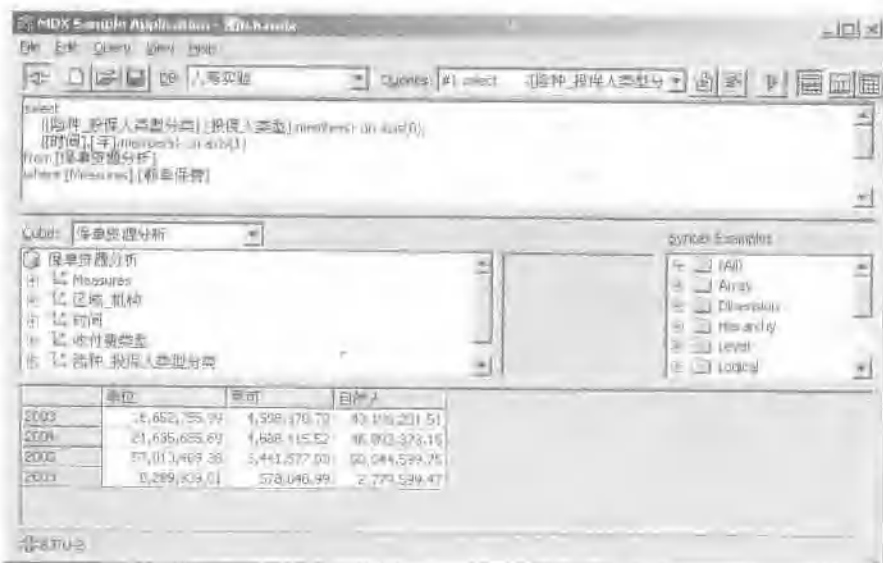


图 6.6 members 函数示例的查询结果

6.4.3 坐标轴与子多维度数据集

如上文所述,MDX 的查询语句与 SQL 的表达形式类似,都是 select - from - where 的形式。事实上,在这种格式的查询语句中,涉及到的是两个子数据集。

第 1 步 坐标轴维度与多维度数据集。进行 MDX 查询时,实际上是对被查询的多维数据集进行划分,在这个划分的过程中,涉及到下列两个子集。

① 坐标轴维度:也就是 select 后面的子集,它可以由多个成员组成,一个 MDX 查询最多可以涉及到 128 个坐标轴。

② 多维度数据集:也就是 where 后面的子集,通常为单一成员所组成。

在一般情况下,使用一般维度作为“坐标轴维度”,使用度量值作为“多维度数据集”。如果没有进行 where 语句的设置,也就是说没有选择“多维度数据集”,那么 MDX 查询会把度量值中的第一个指标默认为 where 子句中的“多维度数据集”。

第 2 步 无“多维度数据集”的查询。在没有“多维度数据集”设置的查询中,系统会默认维度度量值中的第 1 个指标为 where 子句中的“多维度数据集”。下面以一个最简单的多维查询为例,MDX 查询语句如下:

```
select
```

```
[[时间].[年].members] on axis(0)
```

```
from [保单资源分析]
```

在该语句中,只有一个“坐标轴维度”,即[时间]维度中[年]层次的4个成员作为一个坐标轴。数据源为[保单资源分析]这个多维数据集,此处没有设置“多维度数据集”,故系统默认“多维度数据集”为度量值中的第1个指标,在这个多维数据集中为“预收保单”。在输入框中输入查询语句以后,单击工具栏中的三角形符号,即输出查询结果,如图6.7所示。

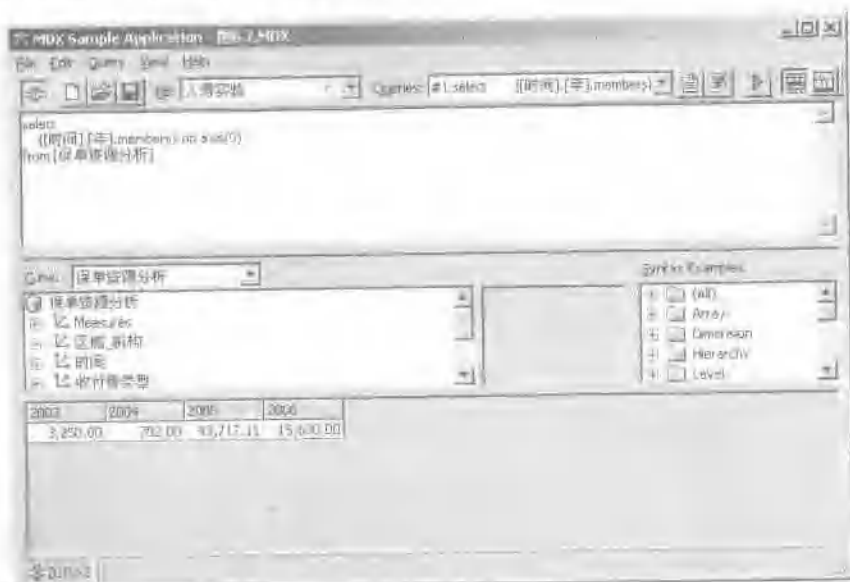


图 6.7 无“多维度数据集”的查询结果

查询结果出现在窗口最下方的输出框中。在业务分析模式的树形结构中,可以看到,measures 中的第1个指标为“预收保单”。在输入框中,输入如下语句:

```
select
```

```
[[时间].[年].members] on axis(0)
```

```
from [保单资源分析]
```

```
where [measures].[预收保费]
```

在此,选择“预收保单”作为“多维度数据集”,将其查询结果与图6.7中的结果相比较会发现,二者的结果完全一致,如图6.8所示。

第3步 有“多维度数据集”的查询。在MDX查询中,通常带有“多维度数据集”选项。例如,想查询每个年份每个区域的实付赔款情况,可以使用如下语句:



图 6.8 输出查询结果

select

{[时间].[年].members} on axis(0),

{[区域_机构, 业务员].[区域分类].members} on axis(1)

from [保单资源分析]

where [measures].[实付赔款]

查询结果如图 6.9 所示。

同时,为了显示各个区域类别的各个度量值指标,也可以使用如下语句:

select

{[measures].members} on axis(0),

{[区域_机构, 业务员].[区域分类].members} on axis(1)

from [保单资源分析]

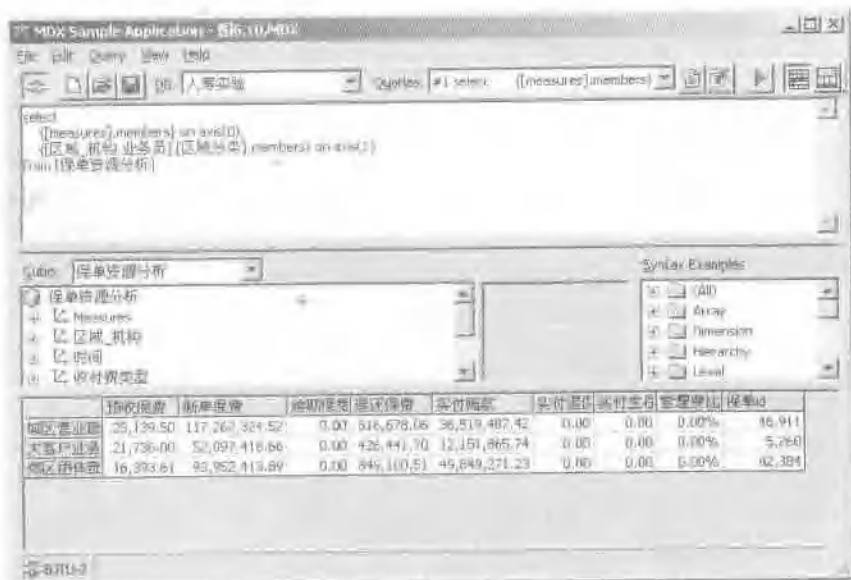
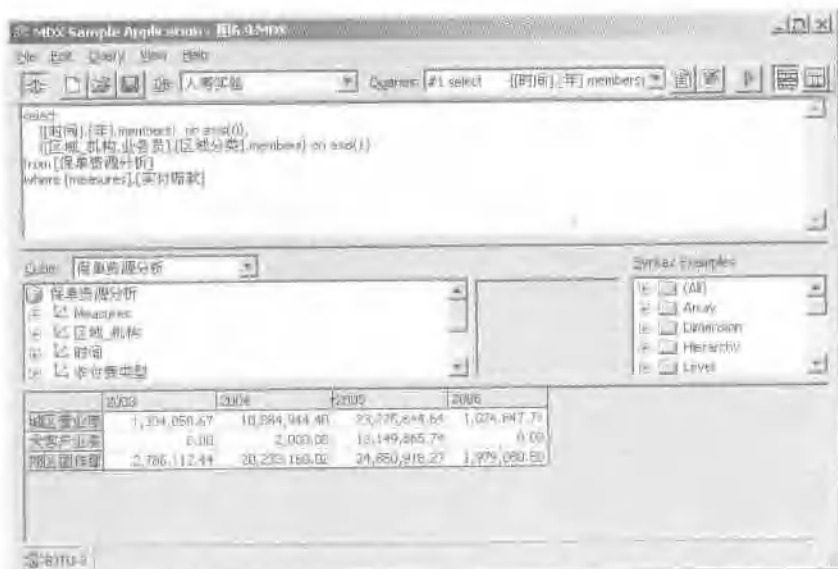
查询结果如图 6.10 所示。

第4步 使用 crossjoin 函数。在多维查询语句中,有时候希望按照嵌套的层次来展示查询的结果。这时,可以使用 crossjoin 函数。例如,在输入框中输入如下语句:

select

{[时间].[年].members} on axis(0),

crossjoin([区域_机构, 业务员].[区域分类].members,[险种_投保人类型分



类].[投保人类型].members) on axis(1)
from [保单资源分析]

where [Measures].[新单保费]

其中,纵坐标使用了 crossjoin 函数,crossjoin([区域_机构_业务员],[区域分类].members,[险种_投保人类型分类].[投保人类型].Members)将区域的类别和三种投保人类型嵌套起来共同作为纵坐标轴。查询结果如图 6.11 所示。



图 6.11 使用 crossjoin 函数输出查询结果

事实上,对于 crossjoin 函数,还有另外一种更为直接的表达方式: {set1} * {set2}。对于上面的查询语句,可以直接用下面的语句加以代替:

```
select
  ([时间].[年].members) on axis(0),
  ([区域_机构_业务员].[区域分类].members * [险种_投保人类型分类].[投保人类型].members) on axis(1)
from [保单资源分析]
where [measures].[新单保费]
```

该语句的查询结果与图 6.11 中的输出结果完全一致。

6.4.4 注释表达式与空数据集的处理

第 1 步 注释表达式的使用。为了表达清晰,不管在何种程序语言中,通常

会使用注释。注释是良好的程序设计风格中不可缺少的一部分。同样,在 MDX 查询中,也存在着注释的使用,在 MDX 中有下列 3 种注释方式。

① //:正斜线注释,与正斜线同一行间的所有文字均被注释掉。

② --:短线注释,与短线同一行间的所有文字均被注释掉。

③ /* */:在起始于“/*”与结束于“*/”之间的所有文字均被注释掉。

例如,对于上面的 crossjoin 函数的语句,可以使用如下注释方法:

```
select
  {[时间].[年], members} on axis(0),
  crossjoin([区域_机构_业务员].[区域分类].members,[险种_投保人类型分
    类].[投保人类型].members) on axis(1)
```

/*

采用时间作为横坐标,

采用 crossjoin 函数将系列与销售形态进行组合,并将其作为纵坐标。

*/

from[保单资源分析]

--数据源维度业务分析模式

where[measures].[新单保费]

//多维数据集维度为销货数量

将上述语句输入到应用程序窗口的输入框中,其查询结果与图 6.11 完全一致,因为注释中的文字是被忽视掉的。

第 2 步 空数据集的处理。在 MDX 查询结果中,不可避免地会出现返回空值的情况。无论是在数值的显示中,还是在字符串的显示中,对返回空值都可以进行一定的处理,比如可以使用 non empty 关键字对空值加以过滤,例如,输入如下查询语句:

```
select
  {[时间].[年].members} on columns,
  {[区域_机构_业务员].[业务员名称].members} on rows
from[保单资源分析]
where[measures].[新单保费]
```

即可查询 4 个季度中所有图书的销货数量,其查询结果如图 6.12 所示。

从查询结果中可以发现,对于很多业务员,其 4 个年度的新单保费业绩均为

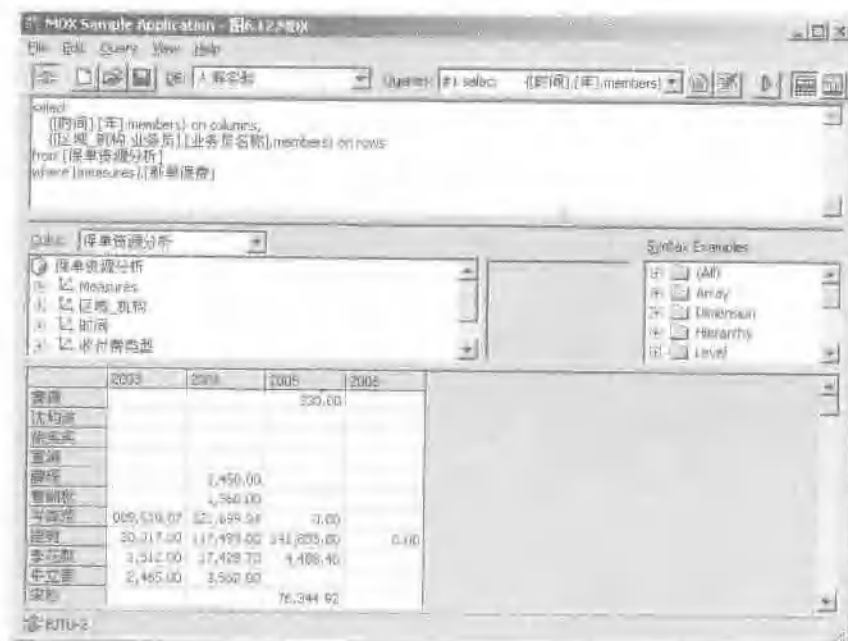


图 6.12 输出查询结果

空。这样,可能希望将这样的空白行过滤掉,可使用 non empty 函数对上面的查询语句做如下修改:

```
select
    ([时间].[年].members) on columns,
    non empty ([区域_机构_业务员].[业务员名称].members) on rows
from [保单资源分析]
where [measure].[新单保费]
```

该语句对返回空值加以过滤,从而使返回结果中的空白行被过滤掉,其返回结果如图 6.13 所示。

6.4.5 建立计算成员和命名集

第 1 步 建立计算成员。在第 5 章中已经介绍了计算成员的概念,在 MDX 应用程序窗口中,同样可以定义和使用计算成员。

建立计算成员,使用如下的定义格式:

```
with member [计算成员的名称] as [计算公式]
```

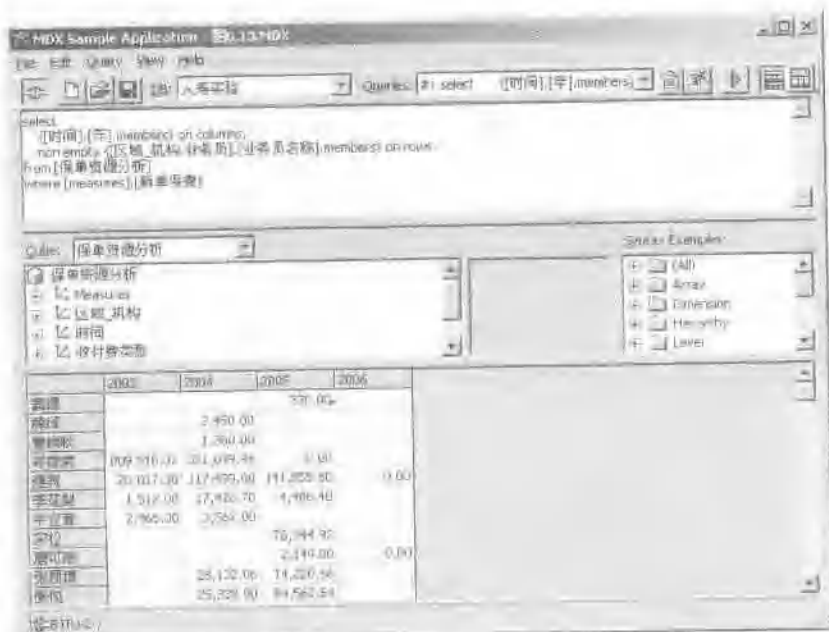


图 6.13 使用 non-empty 函数返回查询结果

在该业务分析模式多维数据集中,我们想计算个人保单的总支出,可以使用“投保人类型”为“个人”的保单的“退还保费”+“实付赔款”+“实付退保金”+“实付生存金”,其定义语言如下:

with member[measures].[个人保单总支出] as'([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[退还保费])+([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付赔款])+([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付退保金])+([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付生存金])'

其中, $\{[\text{险种_投保人类型分类}], [\text{所有险种_投保人类型分类}], [\text{自然人}], [\text{measures}], [\text{退还保费}]\}$ 为一个集合, 为个人保单的退还保费金额, 其余几个集合依次表示个人保单的实付赔款金额、个人保单的实付退保金金额、个人保单的实付生存金金额。

为了显示该计算成员的数值,输入如下 MDX 子句:

with member[measures].[个人保单总支出] as'([险种_投保人类别分类],
[所有险种_投保人类别分类],[自然人],[measures].[退还保费]))+([险种_投保

保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付赔款])+([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付退保金])+([险种_投保人类型分类].[所有险种_投保人类型分类].[自然人],[measures].[实付生存金])

select

 {[measures].[个人保单总支出]} on axis(0),

 {[时间].[年].members} on axis(1)

from 保单资源分析

该语句首先定义了计算成员“个人保单总支出”,然后计算各个年份个人保单的总支出金额。查询结果如图 6.14 所示。

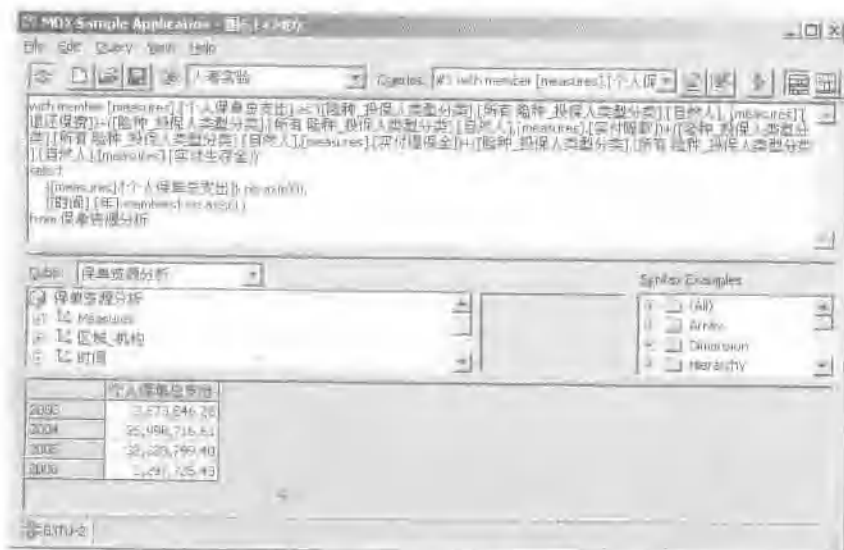


图 6.14 “个人保单总支出”查询结果

第2步 建立命名集。命名集是“维度成员”或“表达式”的集合,主要用于多维表达式查询中的重复使用。

在“MDX 示例应用程序”窗口中也可以定义命名集,其语法格式为:

with set [命名集的名称] as [表达式]

在保单资源分析中,建立一个命名集,为新单保费最多的一个区域的集合,即“冠军销售区域”。其定义语句如下:

with set[冠军销售区域] as* topcount([区域_机构].[业务员].[区域分类]

```
(members,1,[measures].[新单保费]))
```

其中, `topcount` 函数为 MDX 提供的集合函数来排序, 在此例中, 按照“新单保费”对“区域分类”进行排序, 选出第 1 名的区域, 定义为“冠军销售区域”。

在输入框输入如下语句:

```
with set[冠军销售区域] as 'topcount([区域_机构].[业务员].[区域分类],  
members,1,[measures].[新单保费])'
```

```
select
```

```
[冠军销售区域] on axis(0),
```

```
[measures].[新单保费] on axis(1)
```

```
from 保单资源分析
```

该语句首先定义了“冠军销售区域”的命名集, 然后输出这个高业绩区域的新单保费金额, 其查询结果如图 6.15 所示。

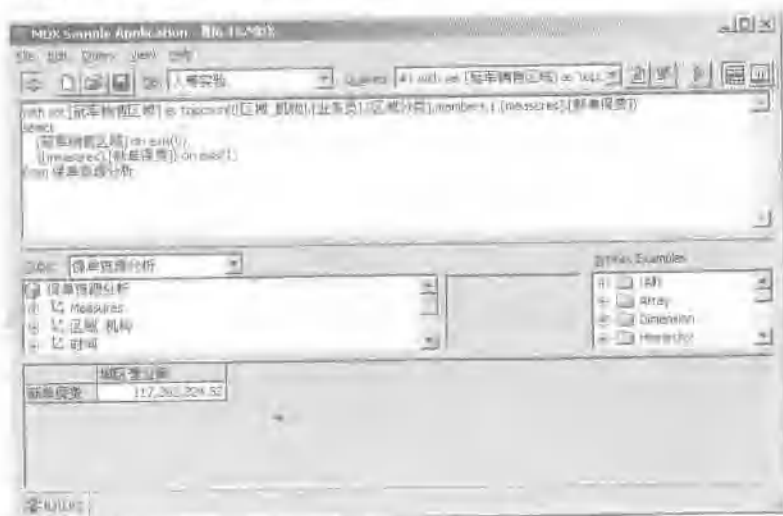


图 6.15 输出查询结果

第 3 步 使用函数表达式。在 MDX 查询中, MDX 语言本身提供了大量的函数, 其中主要包括数值函数和集合函数。这些函数全部包含在“MDX 示例应用程序”窗口中的“Syntax Examples”的树形结构中。

数值函数主要是指用来执行各种统计或汇总计算的函数, 这类函数包括 `aggregate`、`if`、`avg`、`count`、`max`、`min`、`median`、`sum` 等。这里仅就 `aggregate` 和 `count` 的用法给出实例, 对其他函数不再一一列举。

`aggregate` 函数将维度成员进行汇总, 用来返回汇总成员的相关汇总数值。在

保单资源分析中,利用该函数建立新的计算成员。将“时间”维度中的 2003 年、2004 年、2006 年的平均数据定义为“历史年份平均”,将 2006 年定义为“当前年份”,其定义语句如下:

```
with member[时间].[历史年份平均] as 'aggregate([时间].[所有时间].[2003],[时间].[所有时间].[2004],[时间].[所有时间].[2005])/3'
```

```
member[时间].[当前年份] as '[时间].[所有时间].[2006]'
```

在应用程序界面中输入如下的语句:

```
with member[时间].[历史年份平均] as 'aggregate([时间].[所有时间].[2003],[时间].[所有时间].[2004],[时间].[所有时间].[2005])/3'
```

```
member[时间].[当前年份] as '[时间].[所有时间].[2006]'
```

```
select
```

```
{[measures].[members] on axis(0),
```

```
{[时间].[历史年份平均],[时间].[当前年份]} on axis(1)
```

```
from 保单资源分析
```

在该语句中,首先定义了两个计算成员“历史年份平均”和“当前年份”,然后,查询所有历史年份的平均度量值数据和当前年份的度量值数据,其查询结果如图 6.16 所示。

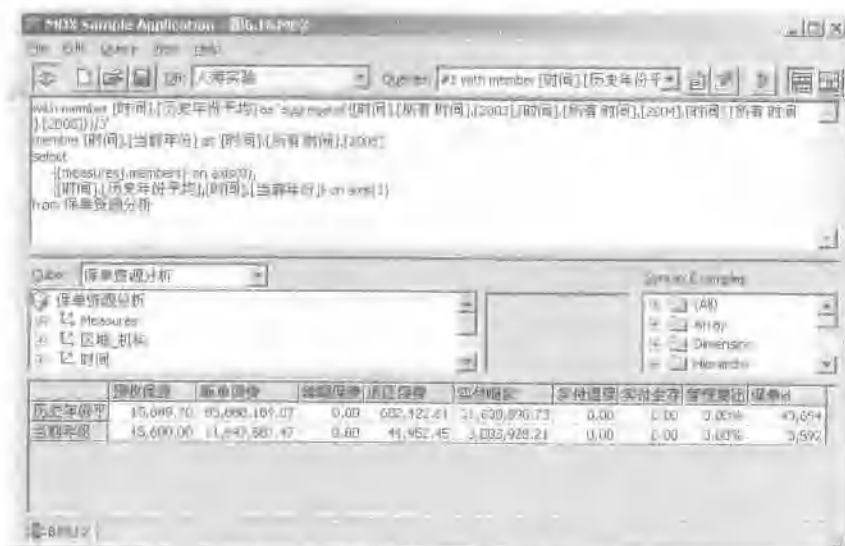


图 6.16 使用 aggregate 函数查询的结果

count 函数用来计算维度或数据集中成员的个数,结合保单资源分析,我们是

用该函数来计算业务员的数量,建立新的计算成员“业务员个数”,来统计业务员的数量。在“MDX 示例应用程序”窗口中输入以下语句:

```
with member [区域_机构].[业务员].[业务员个数] as 'count([区域_机构].[业务员].[业务员名称].members,excludeempty)'
```

```
select
```

```
    | [时间].[年].members | on axis(1),
```

```
    | [区域_机构].[业务员].[业务员个数] | on axis(0)
```

```
from 保单资源分析
```

在该语句中,首先定义了新的计算成员“业务员个数”,用来计算业务员的数量。然后,查询出每个年份业务员的数量。其中,excludeempty 参数排除了空集合,返回了实际的业务员数量。查询结果如图 6.17 所示。

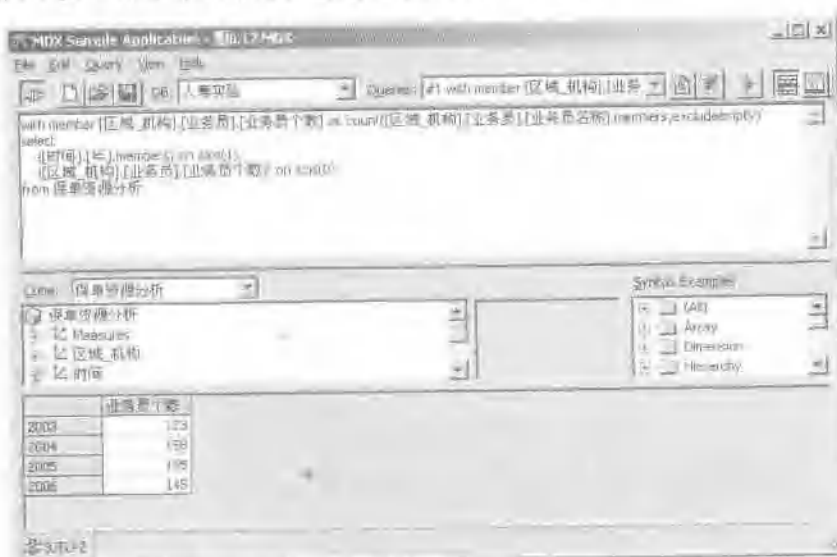


图 6.17 使用 count 函数查询的结果

集合函数包括 bottomcount、bottompercent、children、filter、head、intersect、lastperiods、order、topcount 等。在此,以 children 为例,简要说明集合函数的用法。

children 函数用来返回一个成员的所有子成员,在输入框中输入以下语句:

```
select
```

```
    | [measures].[新单保费] | on axis(0),
```

```
    | [险种_投保人类型分类].[所有险种_投保人类型分类].[单位]
```

```
, children | on axis(1)
```


3. 坐标轴与多维度数据集

1) 无“多维度数据集”的查询

查询“法人客户价值分析保单数量”中每年的“投保人数”,请用 MDX 查询语句实现。

2) 有“多维度数据集”的查询

查询每年每个客户的保险金额,请用 MDX 查询语句实现。

3) 多维数据的嵌套查询

查询不同单位每年在 3 种不同险种类别中的保险金额,提示:使用 crossjoin 函数,实现单位性质与保单类型两个维度数据的嵌套。

4) 空数据集的处理

在查询每个生效年份中所有险种的投保人数时,请实现对空值的过滤。

5) 建立计算成员和命名集

在“MDX 示例应用程序”窗口中,可以定义和使用计算成员。在法人客户价值分析保单数量的多维数据集中,计算所有企业不同险种类型的投保金额,可以使用“单位性质”为“企业”的保险名称的“保险金额”,其计算成员名称为“企业保险金额”。

在“MDX 示例应用程序”窗口中,可以定义并建立一个命名集,在法人客户价值分析保单数量中,定义并建立保险金额最多的 5 个机构的集合,即“五大保险机构”。

6) 使用函数表达式

在法人客户价值分析保单数量中,利用 aggregate 函数建立新计算成员,对维度成员进行汇总,并返回汇总成员的相关汇总数值。将“生效时间”维度中的 2002 年、2003 年、2004 年和 2005 年的数据定义为“历史年份”,2006 年定义为“当前年份”。

利用 count 函数来计算维度或数据集中成员的个数,结合法人客户价值分析保单数量,计算单位的数量。建立新的计算成员“单位数量”,来统计单位的个数。

练习使用集函数,例如,用 children 函数查询所有投意外险的法人客户的保险金额。

本章小结

本章的主要内容是数据仓库中 MDX 语言的使用,首先,介绍了 MDX 中基本的成员表达方式以及一些基本函数;随后,详细介绍了坐标轴与子多维数据集的使用;描述了 MDX 中注释语句的表达方式以及对空数据集的处理方法;最后,举例详述了 MDX 查询的具体办法,包括 select 语句的使用、计算成员的建立、cross-join 函数的使用、命名集的建立;数值函数的使用方法、集合函数的使用方法、判断函数的使用方法等。

MDX 与 SQL 既有相似的地方,又有很大的差别,希望读者可以通过示例和实际的练习细细体会。



第7章

建立 OLAP 的 Web 动态访问

本章引言

通过前面章节的学习,可以熟练地进行多维数据集的建立、编辑、处理和分析。但是,这些功能的实现仅仅是单机、单用户的。在如今这样一个网络信息化时代,信息日益受到人们的重视,是人类社会生产与生活必不可少的一大资源,如何实现数据资源的共享,即多维数据集的 Web 网络访问已经是社会的必然需求。

本章将从直接使用 Analysis Manager 进行数据浏览、使用 Excel 作为前端分析工具和使用 OLAP 的 Web 动态数据透视方面,介绍数据的共享使用技术。其中,Web 动态访问技术是本章的重点,也是难点。

本章重点

- 直接使用 Analysis Manager 进行数据展示。
- 维度的替换和嵌套。
- 使用 Excel 作为前端分析工具。
- 使用 OLAP 的 Web 动态数据进行访问。
- FrontPage 的使用与数据展示。

7.1 实验预备知识

建立 OLAP 分析模式,是让决策分析者从不同角度进行对信息的分析。对于已经建立的多维数据集,可以通过 Analysis Manager、Excel 与 Web 进行 OLAP 的访问和分析。

1. 直接使用 Analysis Manager 进行数据浏览

当完成整个多维数据集的处理之后,可以进行多维数据集中分析数据的浏览。可以看到在所定义的度量值下,不同维度、不同层次上的相关数据;对不同维度可以进行替换和嵌套,以得到所需的数据信息。

2. 使用 Excel 作为前端分析工具

建立多维数据集分析之后,不仅可以通过 Analysis Manager 直接进行数据浏览,还可以在 Excel 中,使用数据透视技术对数据进行分析,并产生统计图表。

3. 使用 OLAP 的 Web 动态数据透视

同样,用户也可以通过 IE 浏览器实现对 OLAP 分析结果的访问。但是,对 Web 服务器部分需要进行网站建立和管理等相关工作,这可以通过网站管理器打开另一个网站、虚拟目录或子目录,使 IIS(Internet Information System, 互联网信息系统)可以顺利识别已建立的 OLAP 动态网页。同时,还应当考虑浏览者的使用权限等因素。即使把 OLAP 分析模式建立于 Web 上,用户仍然可以在 Web 浏览器上调整字段列表、数据属性、显示格式,以便以最适合分析者角度的方式进行分析。

7.2 实验目的

实验的目的是,使读者可以通过本实验的模拟,理解 OLAP 的数据分析模式,学习建立 OLAP 分析数据的 Web 动态访问方式,从而最终在 Web 上建立 OLAP 数据透视,对多维立方体进行展示分析。

7.3 实验要求

本实验要求理解几种主要的 OLAP 数据分析模式,熟练使用 Analysis Manager

来展示数据,掌握使用 Web 进行数据透视的方法。

7.4 实验内容与步骤

本实验将具体讲述如何使用 Analysis Manager 展示数据和如何在 Web 上使用 OLAP 数据的动态访问技术,包括不同维度层次上数据的浏览,维度替换和嵌套的结果分析;如何使用 FrontPage 建立与分析数据库的连接;如何使用数据透视表工具箱进行多维数据集数据的选择、筛选和展示;如何进行网页的保存和发布。本实验主要完成了对“人寿实验”数据库中“保单资源分析”多维数据集的“新单保费”按照“区域分类”进行的筛选、展示和保存。

7.4.1 使用 Analysis Manager 展示数据

实验目的 理解多维数据集的分析模式。通过前面的实验,读者已经系统学习了连接数据源和建立多维模型的操作。对于多维分析而言,最终目的是将分析结果展示给用户。因此,在建立好多维立方体的基础之上,至关重要的就是数据展示,也就是多维数据集的分析模式。在多维数据集的分析模式下,可以通过程序语言来使用数据分析结果。但是,对于一般的使用者而言,简单地说有下列 3 种方式可供使用:

- (1) 使用 Analysis Manager。
- (2) 使用 Web。
- (3) 使用 Excel。

这 3 种方式使用起来简单、便捷,同时能够满足进行多维立方体展示的要求。

实验步骤 使用 Analysis Manager。首先进入到“Analysis Manager”窗口中,选择好数据库,展开树形结构,然后进入到多维数据集中,选择要浏览的多维数据集,单击鼠标右键,选择“浏览数据”,如图 7.1 所示。

实验结果 使用“多维数据集浏览器”。进入到“多维数据集浏览器”对话框中。在“多维数据集浏览器”对话框中,有两个主要部分。如图 7.2 所示,上半部分展示了可选择的维度,其下拉框中包含了维度的层次和所有成员;下半部分是展示结果,按照所选择的维度对指标加以展示。

在“多维数据集浏览器”对话框中,可以进行各种操作。首先,可以将维度展开。单击维度上的“+”符号,可以将维度展开到下一层次。在此例中,将“区域_

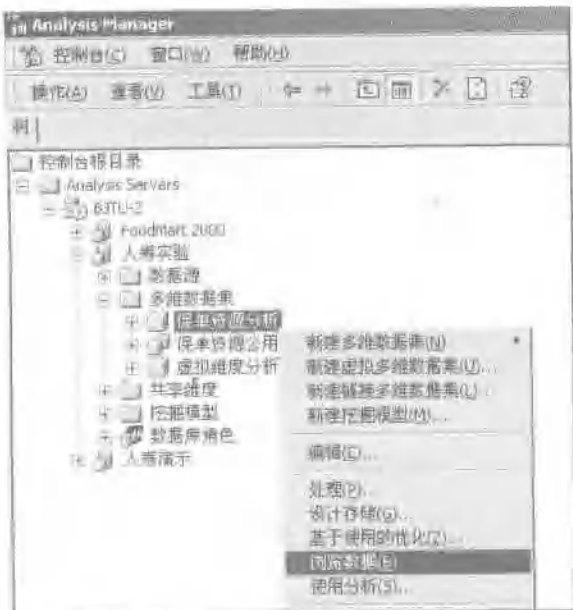


图 7.1 “Analysis Manager”窗口

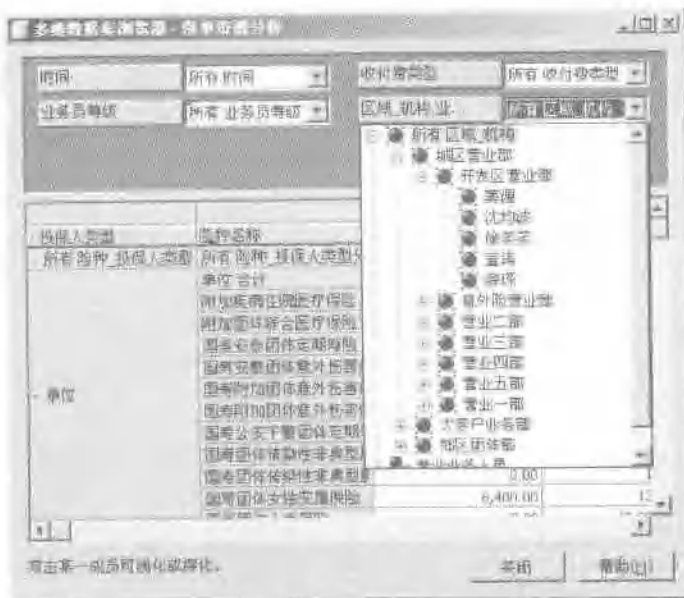


图 7.2 “多维数据集浏览器”对话框

机构、业务员”维度加以展开,进入到“机构”的层次,如图 7.3 所示。

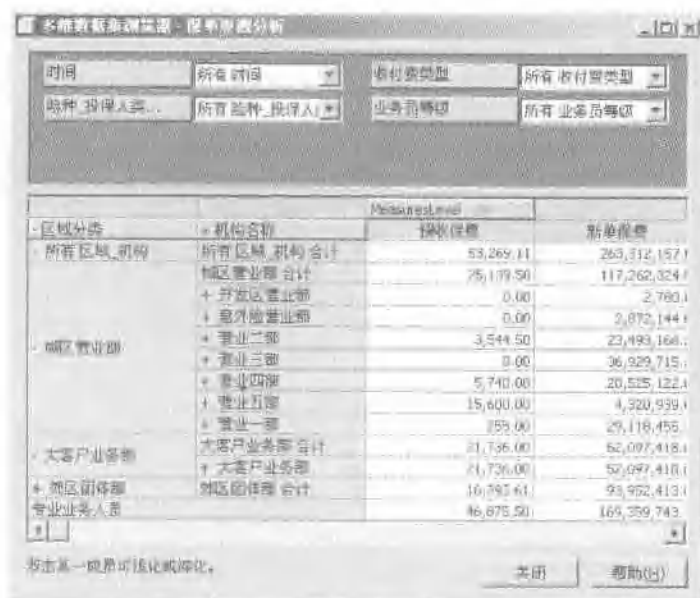


图 7.3 对数据集进行操作

其次,可以进行维度的替换。例如,按照“时间”方式进行数据展示。可以用鼠标单击左方的“时间”,将其拖动到结果展示的维度部分,从而将按照“时间”对指标加以展示,如图 7.4 所示。



图 7.4 维度的替换

同时,可以进行维度的嵌套。例如,将“时间”维度和“投保人类型”两个维度嵌套起来,先将“时间”维度拖动到维度位置,再将“险种_投保人类型”维度拖动到其后面。展示结果如图 7.5 所示。

险种	投保人类型	保费	保额	保费/保额
+ 险种	险种_险种_投保人类型	65,725.11	252,312,157.07	0.00
	• 单位	44,777.01	181,391,000.07	0.00
	• 个人	0.00	17,338,513.17	0.00
	• 其他	20,948.10	53,612,723.88	0.00
+ 2005	险种_险种_投保人类型	3,250.00	31,397,428.28	0.00
	• 单位	3,250.00	16,850,755.99	0.00
	• 个人	0.00	1,548,100.00	0.00
	• 其他	0.00	13,000,572.29	0.00
+ 2006	险种_险种_投保人类型	762.00	7,117,814.38	0.00
	• 单位	0.00	11,635,605.65	0.00
	• 个人	0.00	4,588,415.52	0.00
	• 其他	762.00	46,899,373.15	0.00
+ 2007	险种_险种_投保人类型	41,717.11	119,098,546.98	0.00
	• 单位	34,817.01	91,313,467.30	0.00
	• 个人	0.00	3,141,577.89	0.00
	• 其他	6,899.10	23,644,501.79	0.00
+ 2008	险种_险种_投保人类型	5,600.00	11,641,587.41	0.00
	• 单位	5,600.00	3,299,239.01	0.00
	• 个人	0.00	577,188.22	0.00
	• 其他	0.00	7,774,549.47	0.00

图 7.5 维度的嵌套

总之,在多维数据集浏览器中,主要是通过鼠标的拖动和释放来达到各种分析效果。但是,对于一般的用户而言,该方式只能在“Analysis Manager”中进行,十分不方便。所以,用户通常使用 Web 或 Excel 进行数据透视。

7.4.2 使用 Web 进行数据展示

使用 Web 进行数据展示是使用 Web 进行多维数据集的动态访问,也就是在网页中建立与数据库服务器的连接,从而将分析结果发布到页面上加以展示。

第 1 步 建立与数据库服务器的连接。首先,进入到“Microsoft FrontPage”窗口,进行页面设计,在这里,主要是针对数据库服务器加以介绍。打开“Microsoft FrontPage”窗口之后,在“插入”菜单中选择“Web 组件”,然后在如图 7.6 所示的“插入 Web 组件”对话框的“组件类型”框中选择“电子表格和图表”,再在“选择一个控件”框中选择“Office 数据透视表”。



图 7.6 “插入 Web 组件”对话框

进入到如图 7.7 所示的数据透视界面,可以看到在该界面中有一行工具栏,工具栏的右侧有一个“命令和选项”图标。

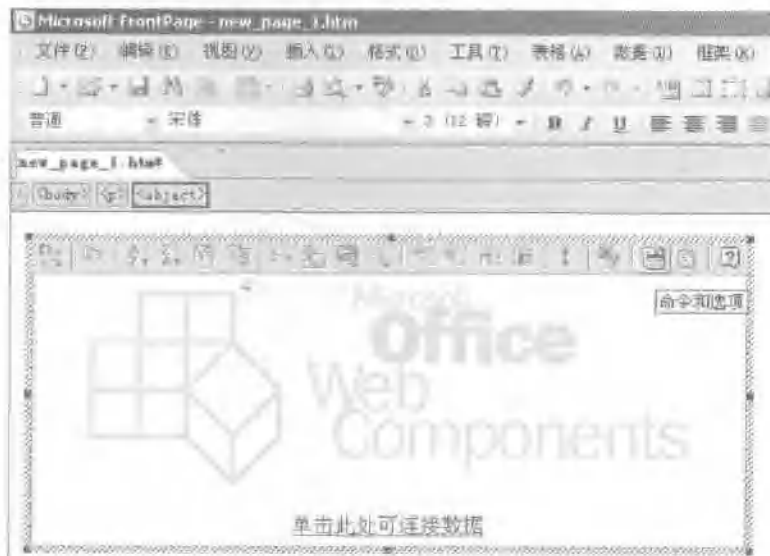


图 7.7 数据透视界面

单击“命令和选项”图标,屏幕上会出现“命令和选项”对话框,如图 7.8 所示。在“命令和选项”对话框中有“数据源”、“特性”和“保护”3 个选项卡。在“数

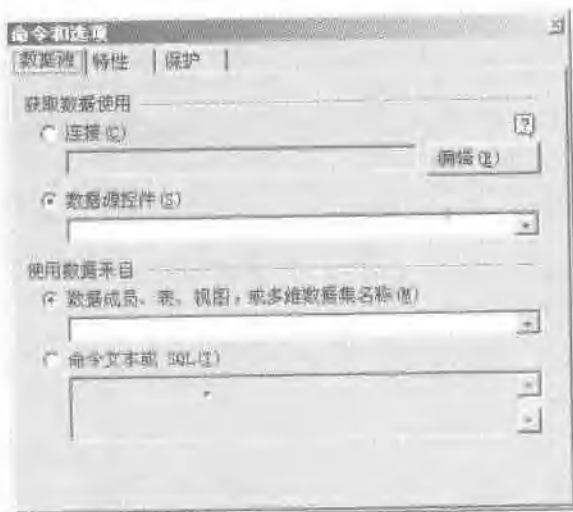


图 7.8 “命令和选项”对话框

据源”选项卡中,选择“连接”,然后单击“连接编辑器”,进入到如图 7.9 所示的“选取数据源”对话框。在该对话框中单击“新建源”按钮。

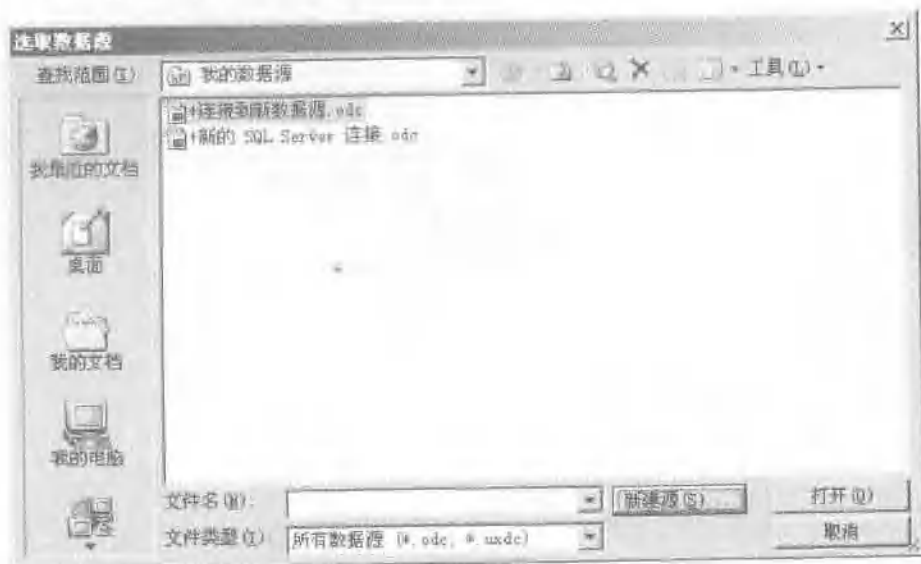


图 7.9 “选取数据源”对话框

进入到如图 7.10 所示的“数据连接向导”对话框后,选择“Microsoft SQL Server OLAP 服务”作为数据源,单击“下一步”按钮。

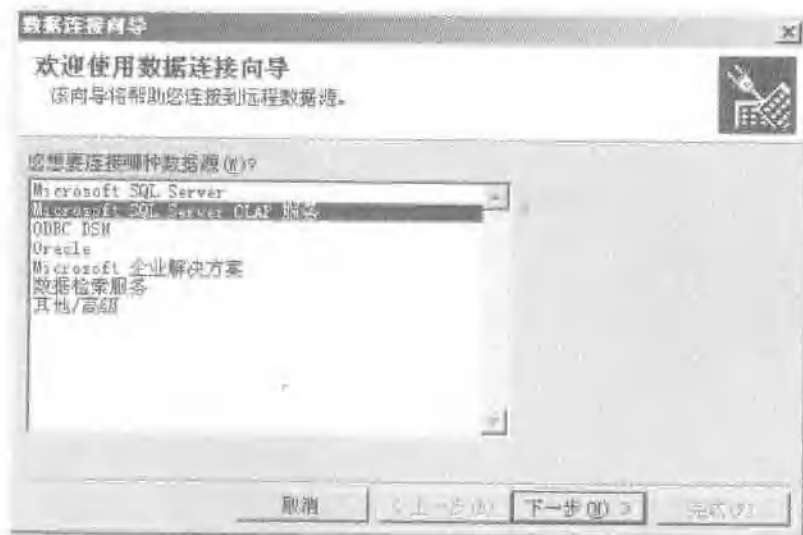


图 7.10 “数据连接向导”对话框

屏幕出现如图 7.11 所示的“连接数据库服务器”对话框,按需求填写对应的服务器名称,并且选择合适的登录凭据,单击“下一步”按钮。

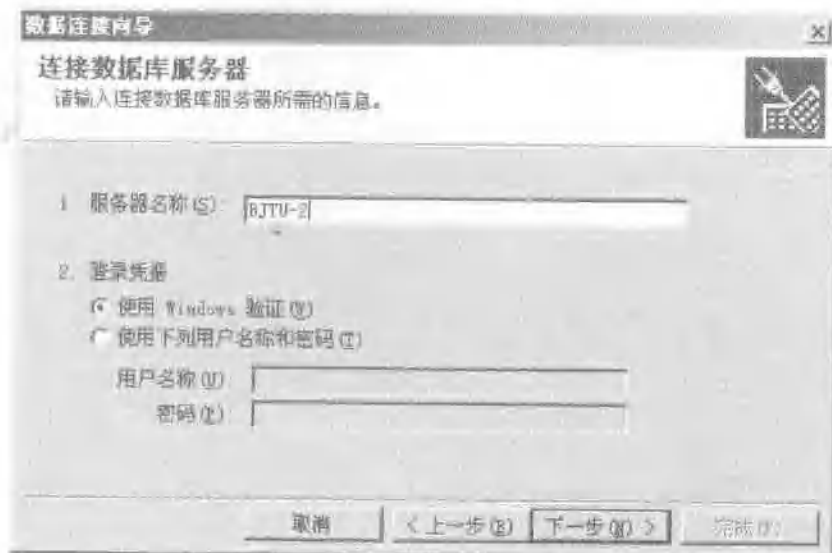


图 7.11 “连接数据库服务器”对话框

这时,屏幕显示如图 7.12 所示的“选择数据库和表”对话框,根据需要选择数

数据库为“人寿实验”，再选择“保单资源分析”多维数据集，单击“下一步”按钮。

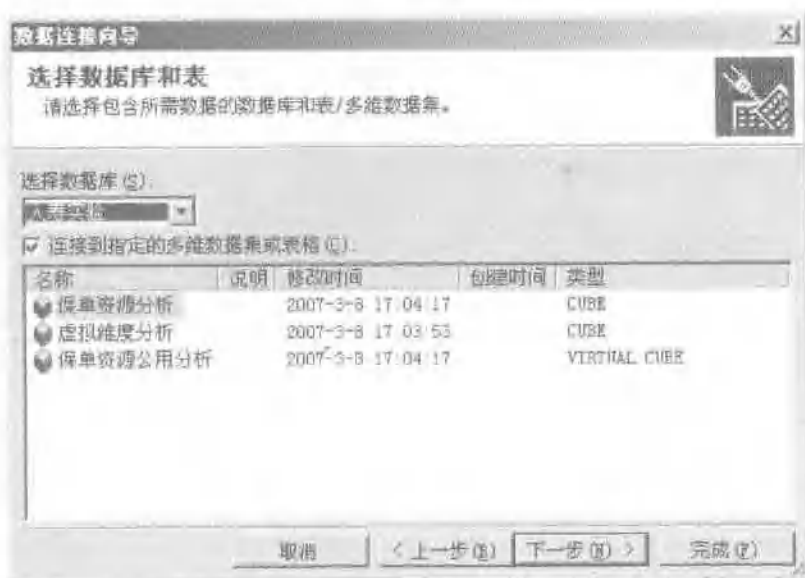


图 7.12 “选择数据库和表”对话框

进入“保存数据连接文件并完成”对话框后，单击“完成”按钮，完成数据连接向导的设置，如图 7.13 所示。

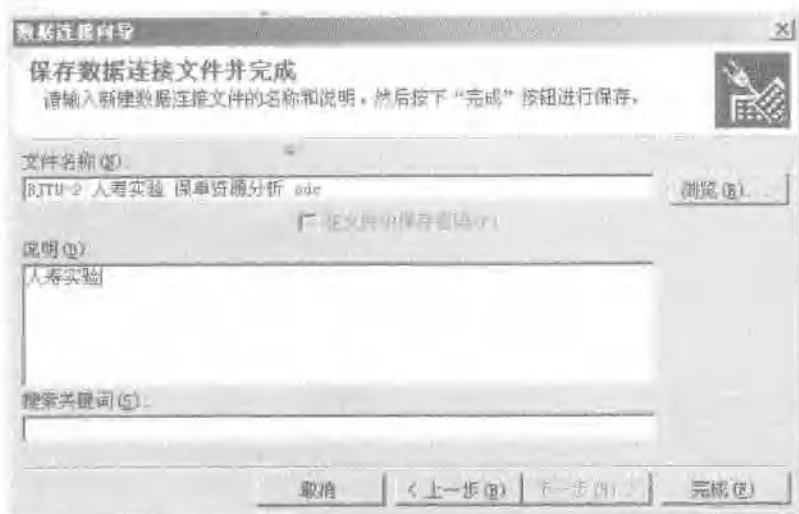


图 7.13 “保存数据连接文件并完成”对话框

新建数据源之后,又回到“选取数据源”的对话框,选取新建立的“BJTU-2 人寿实验 保单资源分析. odc”数据源,单击“打开”按钮,如图 7.14 所示。

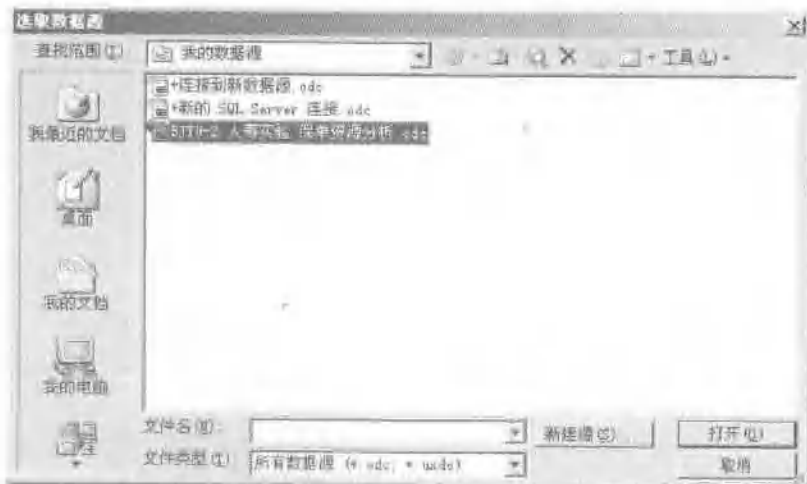


图 7.14 选择“BJTU-2 人寿实验 保单资源分析. odc”数据源

至此,多维数据集的连接已经完成,如图 7.15 所示。

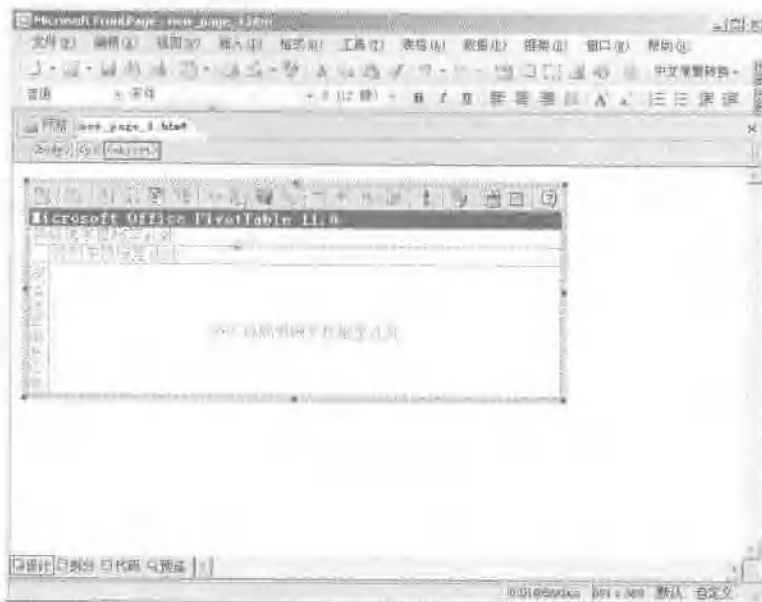


图 7.15 多维数据集的连接已经完成

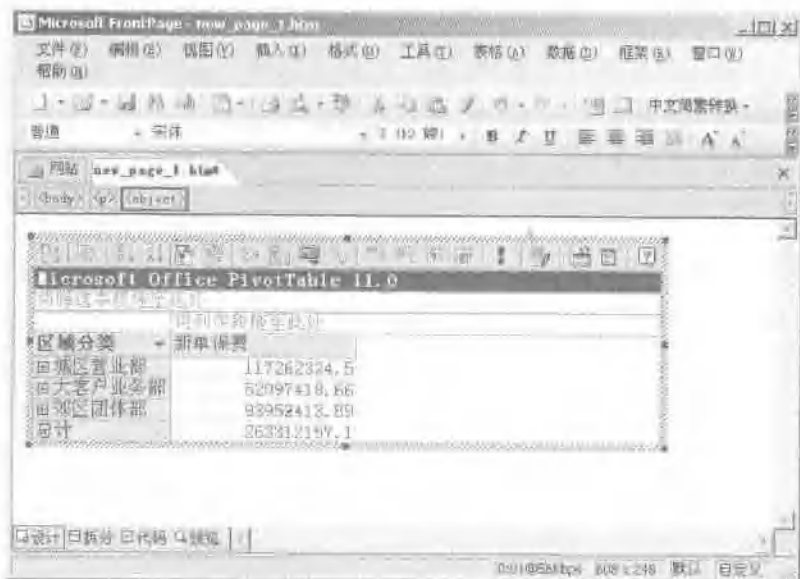


图 7.17 将各个字段拖入到左边的区域内

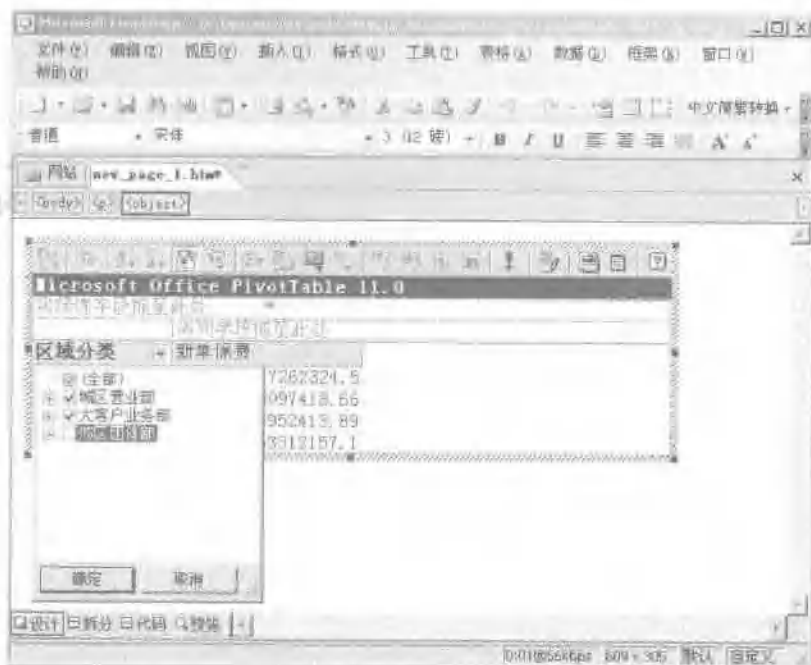


图 7.18 对维度的成员加以选择

在如图 7.19 所示的界面中,只展示了“大客户业务部”和“郊区团体部”两个维度分析数据的界面。然后,可以将该网页保存。最后,直接发布到 Web 上,如图 7.20 所示。

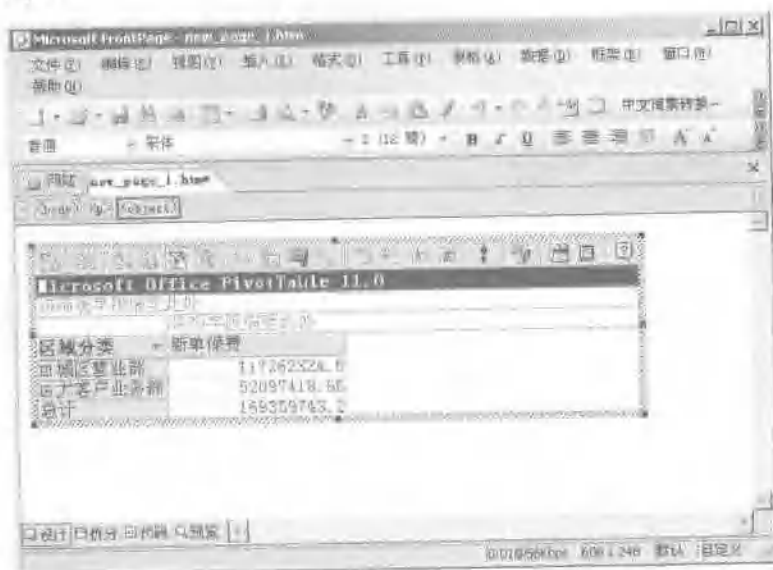


图 7.19 按照区域分类加以筛选的分析结果

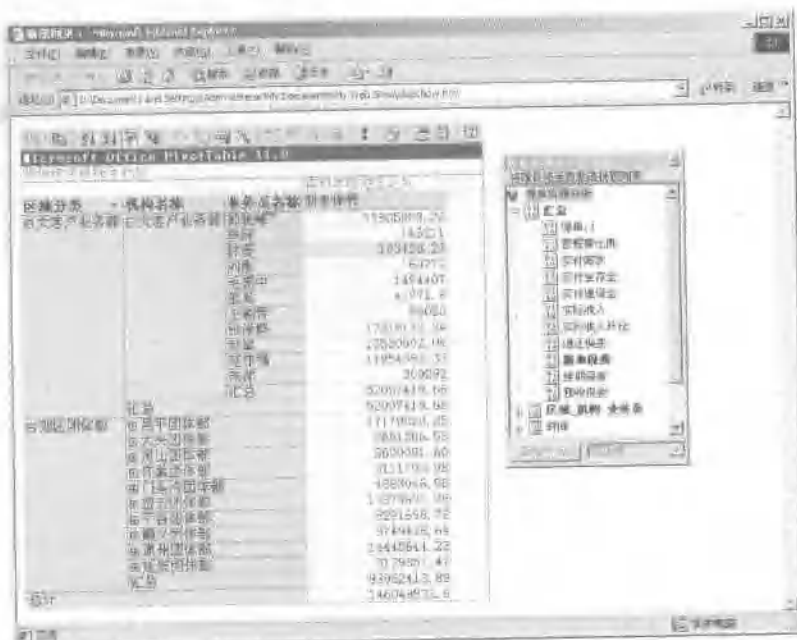


图 7.20 网页保存并发布到 Web 上

在 Web 上,仍然可以对维度和指标进行各种操作,十分灵活、便捷。可以说,Web 的动态访问为多维数据集的展示提供了良好的途径。同时,也可以在网站上进行发布,读者自己可以加以尝试。

另外,还可以通过 Excel 进行多维数据集的展示,其操作过程与 Web 方式类似,这里不再赘述。

7.5 扩展实验

下面介绍关于该章节内容的扩展实验的具体要求,以供有能力的同学得到进一步的提高。

1. 使用 Analysis Manager 直接实现数据浏览

通过 Analysis Manager,使用“多维数据集浏览器”来显示多维数据集“法人客户价值分析保单数量”中的数据,可以把维度展开到该维度下的各个层次。

2. 练习维度的替换和嵌套

练习实现“保单类型”和“客户”的替换,以及在“保单类型”维度中嵌套“险种_险种类别分类”维度,从而分析多维数据集中的数据。

3. 使用 Web 进行数据集的动态访问

可以通过 FrontPage 使用 Web 进行数据集的动态访问,建立与数据库服务器的连接。不仅要完成“人寿实验”数据库的连接,还要完成“法人客户价值分析保单数量”数据源的连接。

在练习对“人寿实验 法人客户价值分析保单数量.odc”文件中的“多维数据集”分析数据的浏览时,实现“单位性质”和“保险金额”在数据区域中的显示,进行“险种_险种类型”和“单位性质”维度的替换、“险种_险种类型”和“开户年_法人客户”维度的嵌套。除此之外,还要对“单位性质”中的“学校”字段进行筛选;同时,实现网页的保存和发布。

本章小结

本章主要介绍了多维数据集分析模式的建立与使用,使用该模式进行数据分析结果展示的3种技术,使用 Analysis Manager 直接实现数据浏览,使用 Excel 作为前端分析工具和使用 OLAP 的 Web 动态数据透视的方式,以及对各技术优缺点的比较与具体使用方法。同时,本章着重介绍了使用 Analysis Manager 直接实现数据浏览、维度的替换和嵌套,以及用 Web 进行多维数据集的动态访问技术。需要注意的是,按照本章的做法,在 Web 上建立 OLAP 分析模式的应用,只能在局域网中进行,即属于 Intranet 层次。若实现在 Internet 上的访问与查询显示功能,应使用其他方式进行处理。此外,对于分析数据的保护,必须注意安全性设置,以避免让无使用权限的用户获得不当的数据。



第8章

系统维护与管理

本章引言

完成数据库系统的建立后,还要对系统进行维护和管理。这部分的内容包括系统的安全性控制与权限控制,数据库的备份和恢复,多维数据集的优化以及系统环境的设置等。对系统的维护与管理,可以提高数据库系统的安全性,控制用户的使用权限,保证数据的存储安全,提高多维数据集的访问效率等。

本章重点

- 数据库角色管理。
- 多维数据集角色管理。
- 数据库的存档与恢复。
- 多维数据集的优化。
- 系统环境的设置。

8.1 实验预备知识

OLAP 数据分析中所用到的数据,主要来自企业中的数据库,因此在设置数据源时,必须具有数据库的访问权限,然后才能以数据库中的数据为基础,产生 OLAP 分析数据。

模式建立者建立了 OLAP 并产生“汇总”数据后,从数据“安全性”角度来说,并不允许所有人都可以挖掘每一个多维数据集的数据,也不是每一个用户都可以对多维数据集模式进行修改,因此,应该进行 Analysis Servers 的管理、维护与使用权限设置。

当建立好一个多维数据集模式后,除了选择“实体存储模式”外,还要进行汇总项目的性能提升测试,主要的目的是,在存储空间与“汇总项目”计算的总项数之间取得一个平衡点。

很显然,系统事先计算完成的“汇总项目”越多,用户查询结果的响应速度将越快,但所占的空间将越大。从处理效率的观点来说,当然希望能有足够的空间放置所有的“汇总项目”和相应的数据,但是,在实际的应用中,可能会有困难。

从实际的应用来说,当第一次建立好分析模式后,进行汇总项目的性能提升测试,并没有区分哪些维度、哪些度量值是比较常用的,也没有分析哪些用户在哪些时段常使用哪些分析组件,但随着系统逐渐的使用,这些使用状况的分析数据便会被逐一地记录到系统上。此时,Analysis Manager 提供两个重要的工具,帮助用户取得使用状况的分析数据,并根据这些数据进行系统使用优化的调整。

8.2 实验目的

通过本章的学习与实践,读者应掌握使用权限的设置和角色管理,能够进行数据库的备份与恢复,同时能够了解对多维数据集进行优化的基本操作以及系统环境的一些常规设置。

8.3 实验要求

读者通过对本章的学习,最后应掌握数据库的角色管理的主要操作过程,能对数据库进行存档与恢复,能够进行多维数据集的优化工作,可以进行系统环境

的常规配置。

8.4 实验内容与步骤

安装 Analysis Servers 的时候,系统会自动建立一个“OLAP Administrator”用户组,该组中的成员可以访问 Analysis Servers 中的数据库,相当于“系统管理员”。同时,可以在“Analysis Servers”窗口中建立角色,向角色中添加用户,并进行权限的设置。

在“Analysis Servers”窗口中,进行角色管理共有3种模式:数据库角色管理、多维数据集角色管理和挖掘模型的角色管理。本实验中将以以前两者为主。

8.4.1 数据库角色管理

不管是何种模式的角色管理,向角色中添加的用户都是基于系统的用户或组。因此,在进行角色管理之前,有必要先搞清楚系统用户管理中用户和组的概念。

在“Analysis Servers”窗口中,可以进行对数据库角色的添加、修改和删除等操作。

第1步 数据库角色的添加。在“Analysis Servers”窗口中,鼠标右键单击“数据库角色”,然后选择“管理角色”,打开“数据库角色管理器”对话框,如图8.1所示。

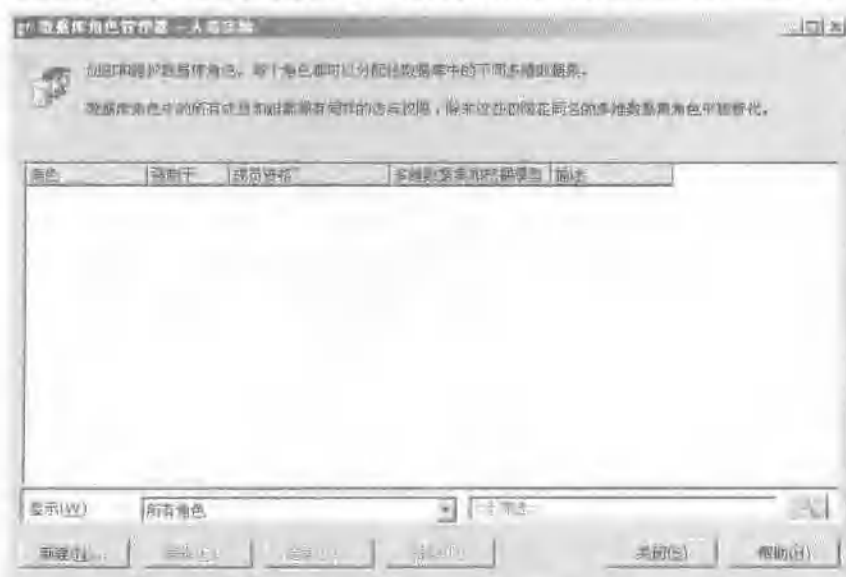


图 8.1 “数据库角色管理器”对话框

在“数据库角色管理器”对话框中,可以列出数据库中现有的角色。目前,在“人寿实验”数据库中不存在角色。

在“数据库角色管理器”对话框中,可以添加角色。例如,我们需要建立一个名称为“一级主管”的新角色,该角色包含 3 个系统用户: user6、user7 和 user8,下面以此为例,介绍如何添加数据库角色。

在“数据库角色管理器”对话框中,单击“新建”按钮,屏幕出现“创建数据库角色”对话框,在该对话框的“角色名”框中输入要建立的角色名称“一级主管”,如图 8.2 所示。

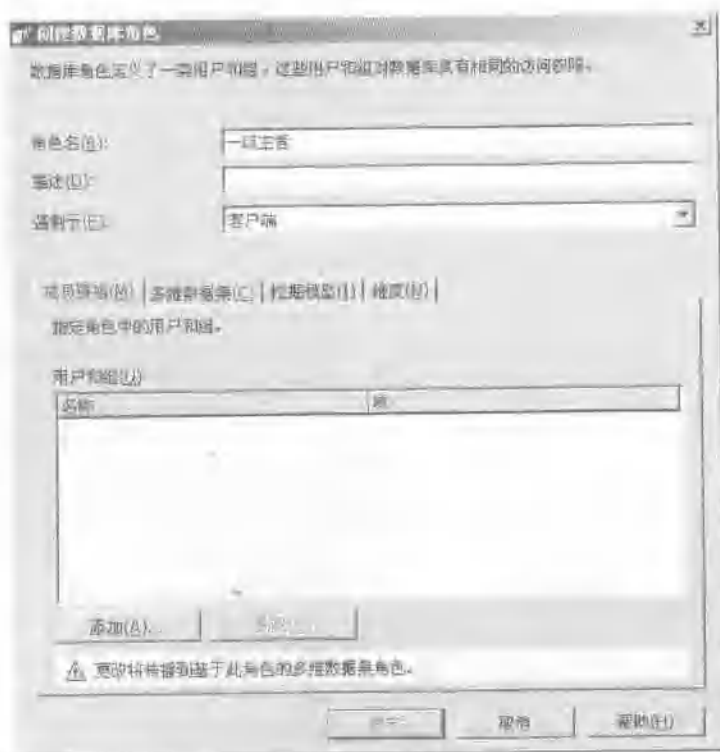


图 8.2 “创建数据库角色”对话框

在建立角色的同时,需要向角色中添加系统用户或组。在“创建数据库角色”对话框中单击“添加”按钮,开始向角色中添加用户。在如图 8.3 所示的“添加用户和组”对话框中,单击“显示用户”按钮后,将在上部的“名称”列表中显示出系统中的所有用户。将“user6”、“user7”和“user8”添加到下部的列表中,即将这些用户添加到“一级主管”的角色中。

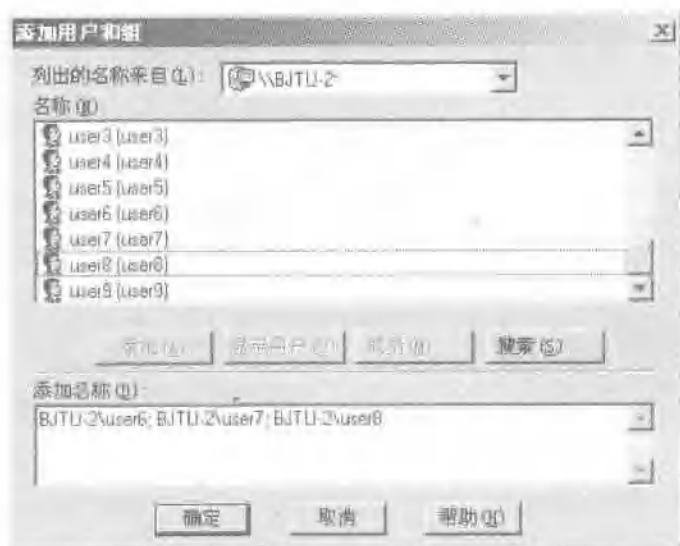


图 8.3 “添加用户和组”对话框

连续单击“确定”按钮,便完成了数据库角色的建立工作。这样,在“数据库角色管理器”对话框中便可以看到一个新的角色“一级主管”,如图 8.4 所示。

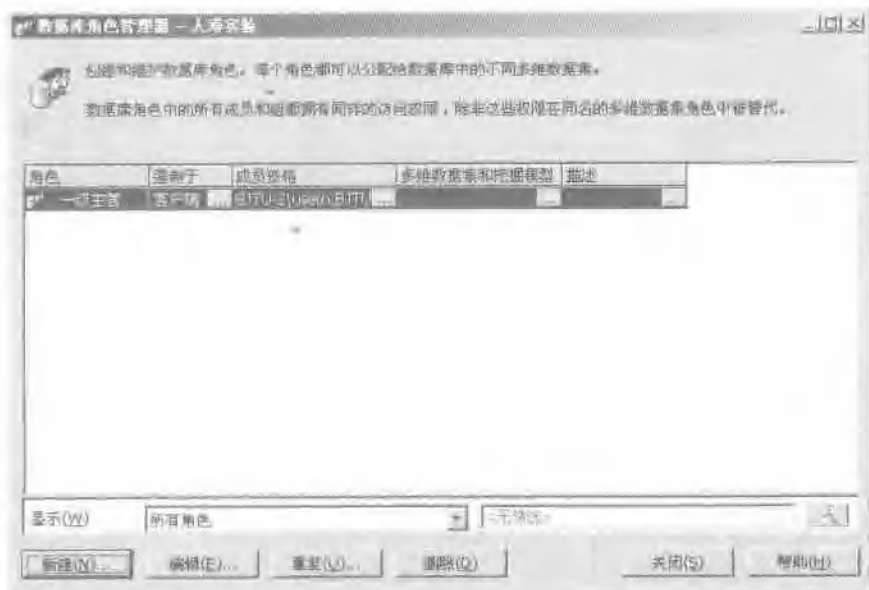


图 8.4 新的角色“一级主管”添加完成

第2步 管理角色的编辑。可以对已经存在的数据库角色进行编辑和修改,包括用户的添加与删除,以及对权限的设置。在此对刚刚建立的“一级主管”的角色进行编辑。选定“一级主管”所在行,单击“编辑”按钮,打开如图 8.5 所示的“编辑数据库角色”界面。

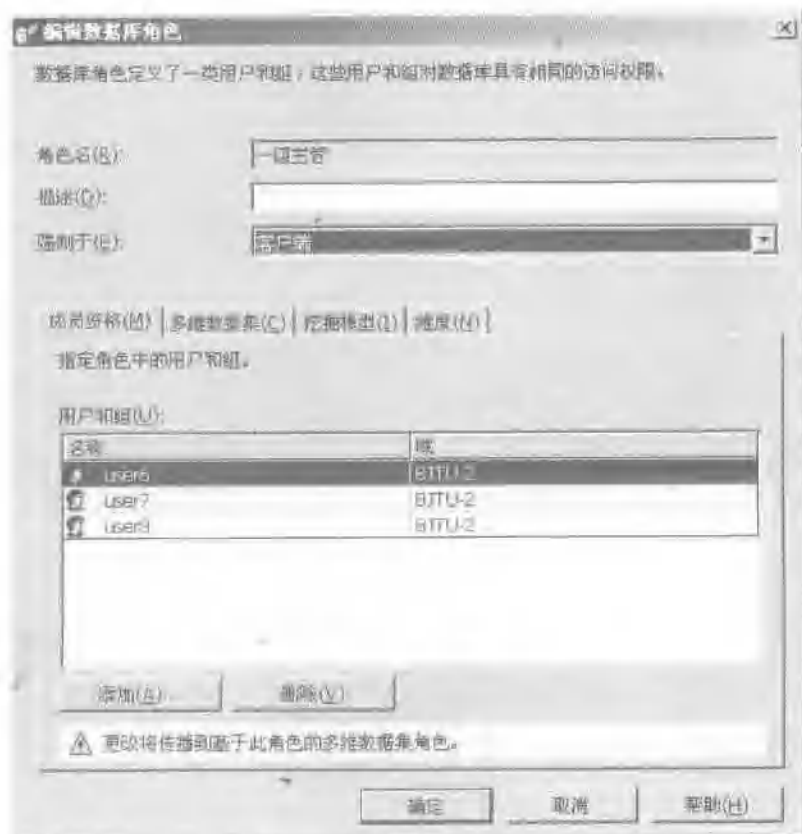


图 8.5 “编辑数据库角色”对话框

在“编辑数据库角色”对话框中,可以对角色进行编辑。在对话框中可以看出,编辑工作可以分为 4 个部分:成员资格、多维数据集、挖掘模型和维度。

当前屏幕显示的是“成员资格”选项卡,列出了该角色中所包含的用户和组。选中某一个用户,单击“删除”按钮,即完成了删除该用户的工作。同时,可以单击“添加”按钮,向角色中添加新的用户,如图 8.3 所示。

在如图 8.6 所示的“多维数据集”选项卡中,可以对多维数据集的权限进行设置,单击某一个多维数据集前面的复选框,可以为角色添加对该多维数据集的操

作权限;反之,可以取消角色对该多维数据集的操作权限。例如,这里只为“一级主管”添加“保单资源公用分析”这个多维数据集的管理权限。

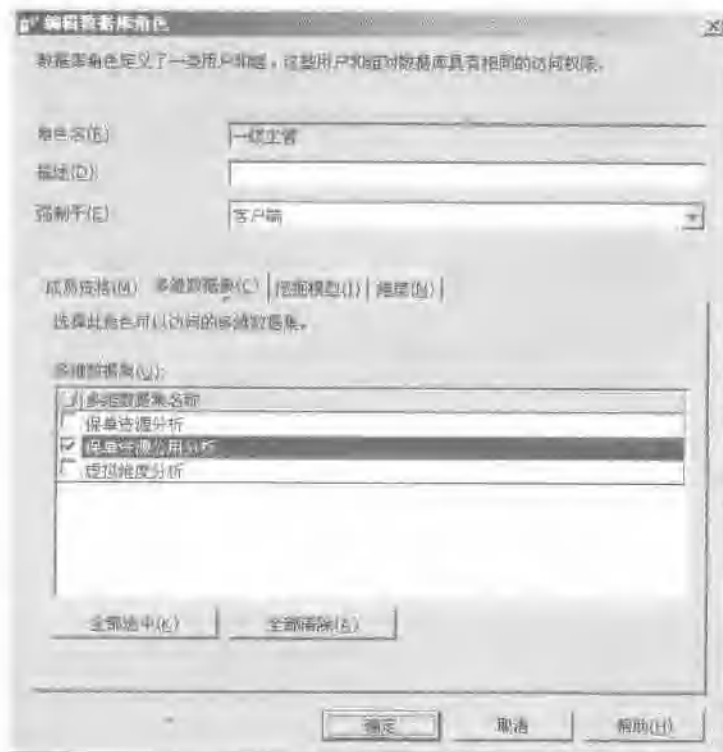


图 8.6 “多维数据集”选项卡

除了可以对多维数据集进行权限设置以外,还可以对维度进行权限的设置。可在如图 8.7 所示的“维度”选项卡中,对维度进行权限设置。

在“维度”选项卡中,可以对所有的共享维度进行权限上的设置,共有 3 种模式:无限制,完全受限和自定义。其中,“无限制”模式是指角色对该维度具有所有的权限,可以访问所有的成员;“完全受限”模式是指角色对该维度不具备任何访问权限;“自定义”模式比较灵活,可是按照所需要的条件加以限定。

下面以“区域_机构_业务员”这个维度为例,进行自定义设置。单击“区域_机构_业务员”一行中“自定义设置”下的  按钮,进入到如图 8.8 所示的“自定义维度安全性”对话框。

在“自定义维度安全性”对话框中,区域_机构_业务员维度中的层次和成员均显示在“成员”区内,可以选中每个成员前面的复选框,选中的成员对该角色来

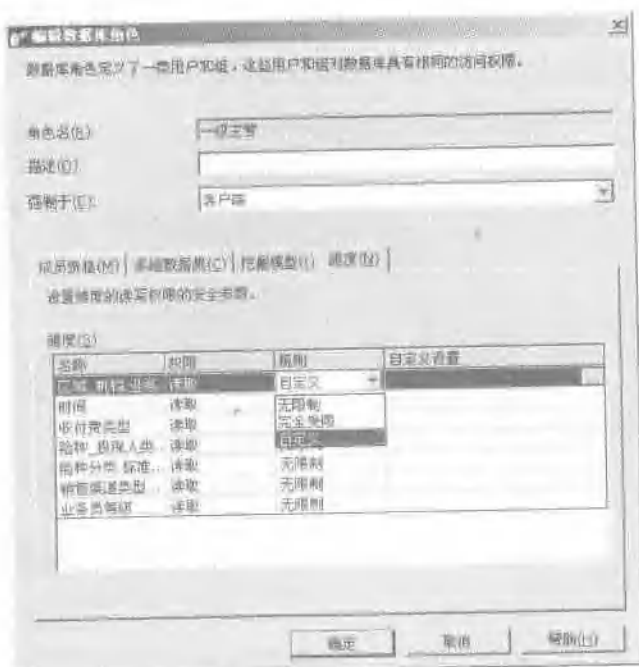


图 8.7 “维度”选项卡

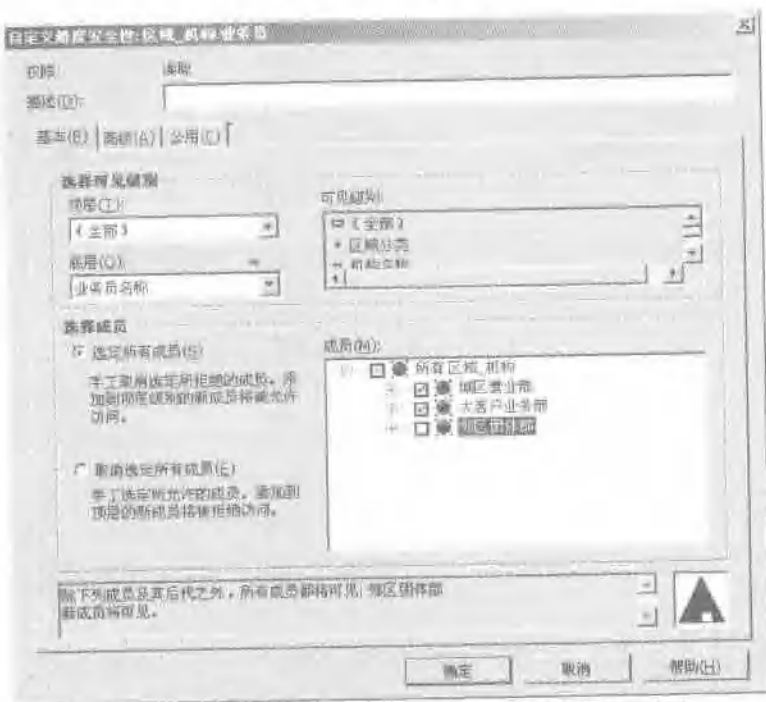


图 8.8 “自定义维度安全性: 区域_机构_业务员”对话框

说是可见的,反之不可见。在此例中,因为没有选择“其他工号”、“团体代理”、“银行保险”、“营销”等机构类型,所以“一级主管”中的所有成员将无法看到机构类型为“其他工号”、“团体代理”、“银行保险”、“营销”的业务员数据。

在“自定义维度安全性”对话框中,还有“高级”和“公用”两个选项卡。在这两个选项卡中,可以进行手工录入,并且可以使用“MDX 生成器”。“高级”选项卡,如图 8.9 所示。

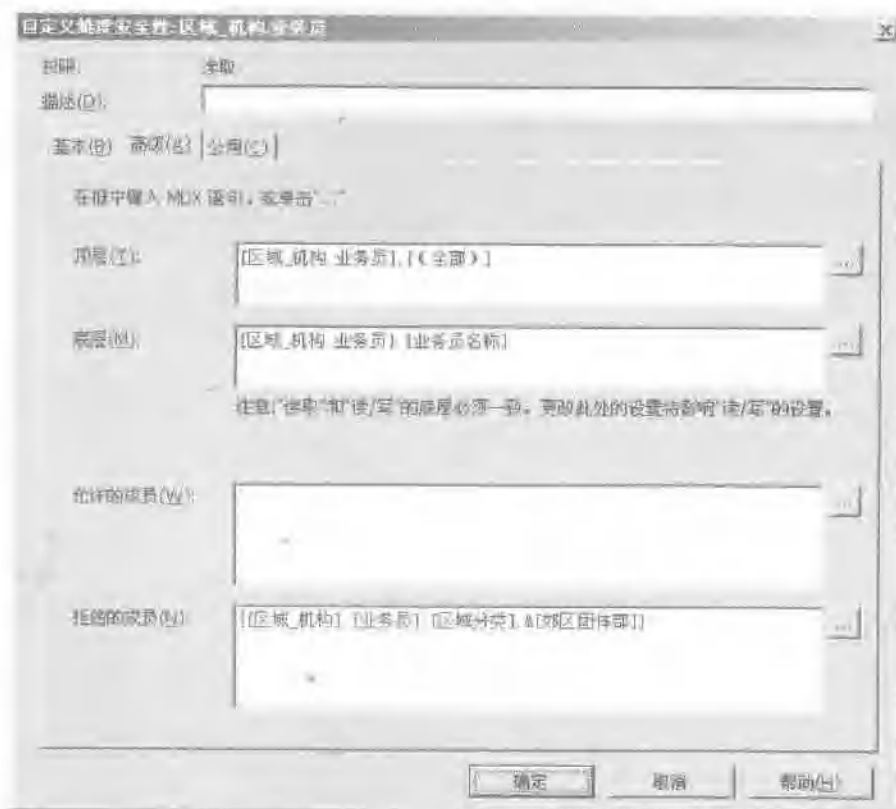


图 8.9 “高级”选项卡

单击该选项卡中的任何一个  按钮,都可以直接进入“MDX 生成器”。其中,“MDX 生成器”在前面提到的计算成员中,已经介绍了它的基本用法。

然后,依次单击“确定”按钮,完成对该数据库角色的编辑工作。

第3步 数据库角色的删除。在“数据库角色管理器”窗口中,除了能够对角色进行添加和编辑以外,还能删除现有的角色。操作十分简单,只需要在“数据库角色管理器”窗口中选择相应的角色,单击“删除”按钮即可。

8.4.2 多维数据集角色管理

数据库角色管理是针对数据库中的所有对象。同时,针对多维数据集可以进行多维数据集角色管理。下面主要介绍多维数据集角色管理。

第1步 打开“多维数据集角色管理器”对话框。在“Analysis Servers”窗口中,选中某个多维数据集(这里以“保单资源分析”为例)。右键单击该多维数据集,然后选择“管理角色”菜单选项,屏幕出现“多维数据集角色管理器”对话框,如图 8.10 所示。

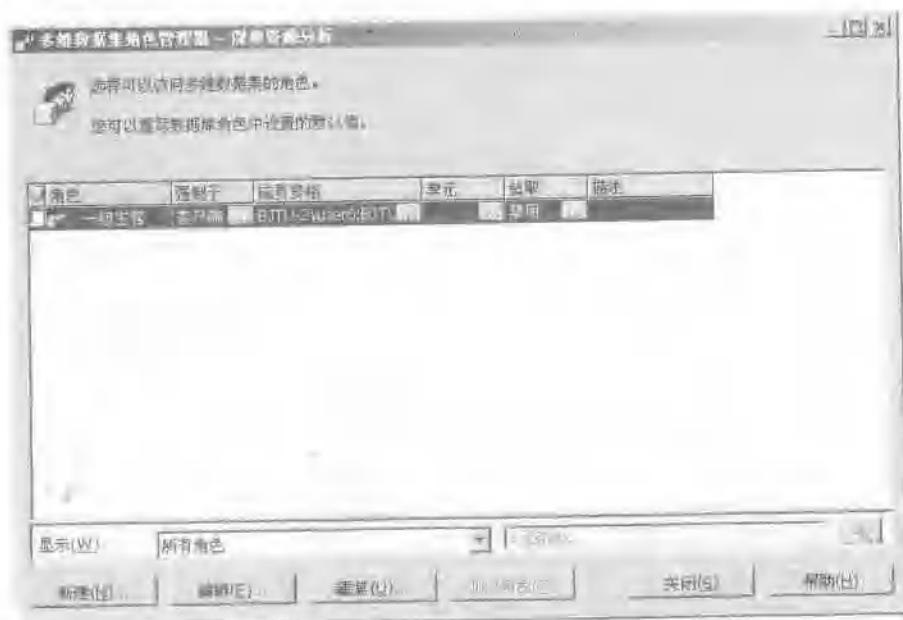


图 8.10 “多维数据集角色管理器_保单资源分析”对话框

该对话框中显示出该多维数据集中已经存在的一个角色。

第2步 角色的建立与删除。在“多维数据集角色管理器”对话框中,可以进行角色的添加和删除,其操作过程与“数据库角色管理器”对话框完全一致,这部分操作可参见上一节的第1步操作,这里不再赘述。

第3步 角色的编辑。在“多维数据集角色管理器”对话框中除了可以进行角色的创建和删除以外,还可以对角色进行编辑。选择要编辑的角色所在行(这里仍然以“一级主管”为例),单击“编辑”按钮,打开“编辑多维数据集角色”对话框,如图 8.11 所示。

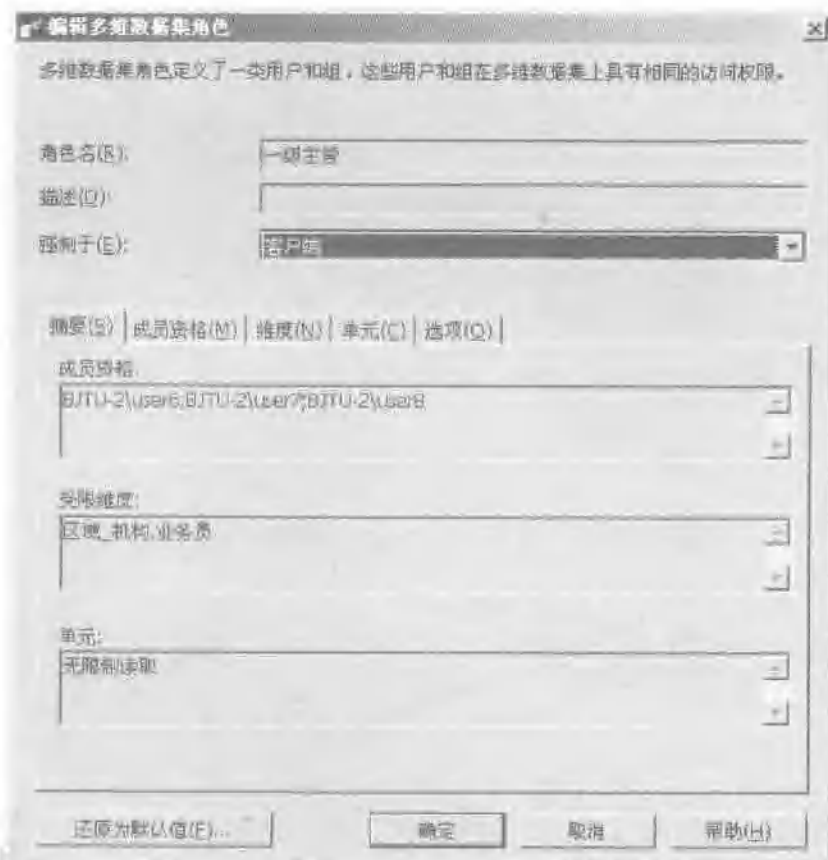


图 8.11 “编辑多维数据集角色”对话框

在“编辑多维数据集角色”对话框中，共有 5 个选项卡：摘要、成员资格、维度、单元和选项。其中，“摘要”对话框只是静态信息的提示，而“成员资格”和“维度”对话框与“数据库角色管理器”对话框中的功能和操作完全一致，这里不再重复。

在“单元”选项卡中，在“单元安全性策略”列表框中包括“无限制读取”和“高级”两个选项。前者对多维数据集中的单元不加以限制，后者进行了一些灵活的限制。选择“高级”选项，如图 8.12 所示。

在如图 8.12 所示的对话框中，单元的权限分为三种：“读取”，说明角色可以查看的单元；“读附随”，说明角色在条件限制下可以查看的单元；“读/写”，说明角色可以更新的单元。

同时，“高级单元安全性”提供了三种规则：无限制、完全受限和自定义。这三

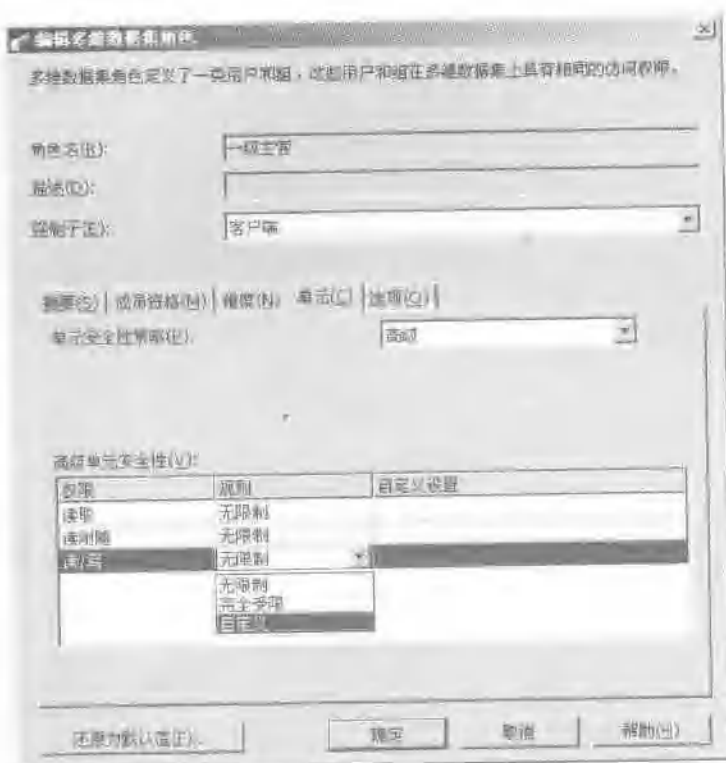


图 8.12 选择“单元”中的“高级”选项

种规则与“数据库角色管理器”十分相似。对于“自定义”规则，同样可以使用“MDX 生成器”进行自定义设置。

进入如图 8.13 所示的“选项”选项卡。

在“选项”选项卡中，可以选择“允许钻取”，“允许链接到此多维数据集”以及“允许向此多维数据集发送 SQL 查询”选项。可以根据需要或兴趣进行选择，以便查看其效果。

8.4.3 数据库的存档与恢复

对于 Analysis Servers 而言，基于安全性的考虑，有必要对数据库进行备份。除此之外，还有可能需要进行数据库的迁移。针对这些需求，Analysis Manager 提供了数据库的存档与恢复的功能。

第 1 步 数据库的存档。在对数据库进行存档的过程中，系统会先自动把分析数据库压缩成 .cab 文件，然后再加以保存。

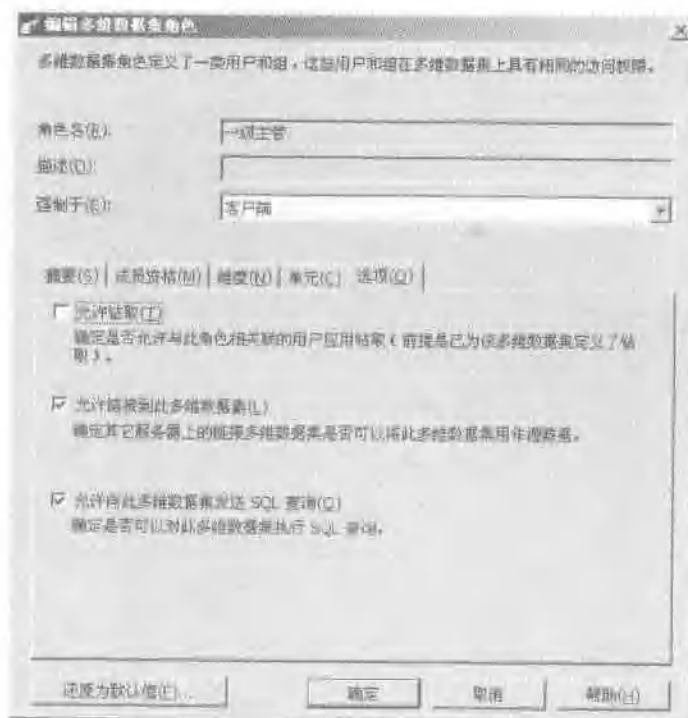


图 8.13 “选项”选项卡

在“Analysis Servers”窗口中,选中要进行备份的分析数据库,单击鼠标右键,然后在弹出的菜单中选择“将数据库存档”选项,如图 8.14 所示。

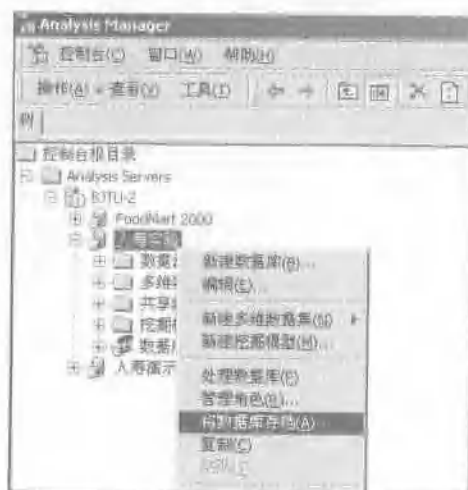


图 8.14 选择“将数据库存档”

在“将数据库存档”对话框中选择要存档的路径和文件名,如图 8.15 所示。

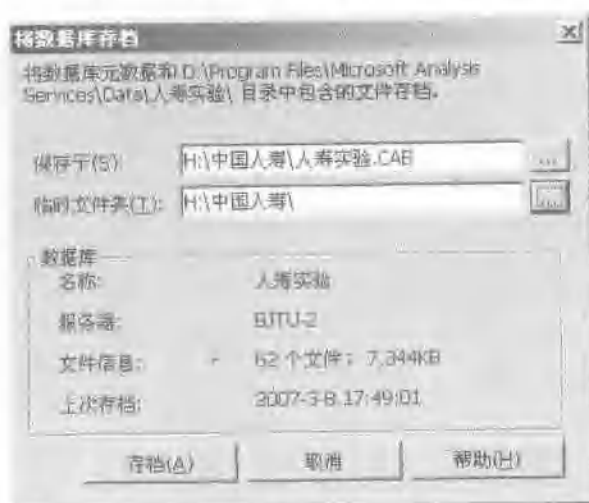


图 8.15 “将数据库存档”对话框

单击“存档”按钮,数据库存档成功,如图 8.16 所示。



图 8.16 数据库存档成功

单击“关闭”按钮,即完成了对数据库的存档操作。

第2步 数据库的恢复。在“Analysis Servers”窗口中,可以将 .cab 格式的分析数据库压缩文件还原,鼠标右键单击所连接的服务器,选择“还原数据库”菜单选项,如图 8.17 所示。

在如图 8.18 所示的“打开存档文件”对话框,选择所要还原的数据文件。

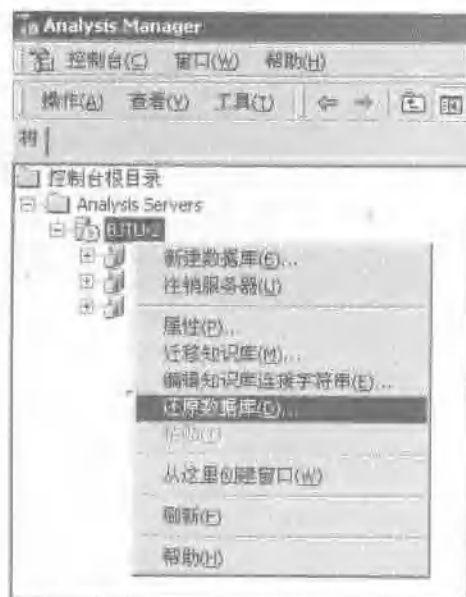


图 8.17 选择“还原数据库”

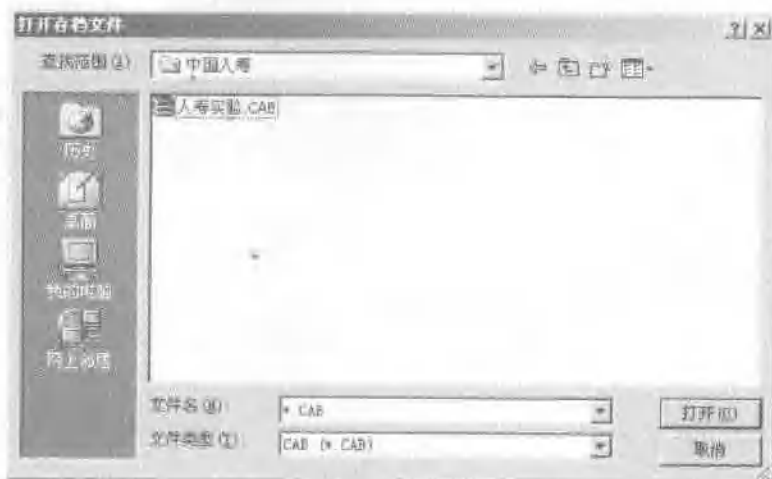


图 8.18 “打开存档文件”对话框

选择好要还原的数据库文件之后,进入“还原数据库”对话框,如图 8.19 所示。

单击“还原”按钮,进行数据库的还原,如图 8.20 所示。

单击“关闭”按钮,即完成了 Analysis Servers 的还原工作。

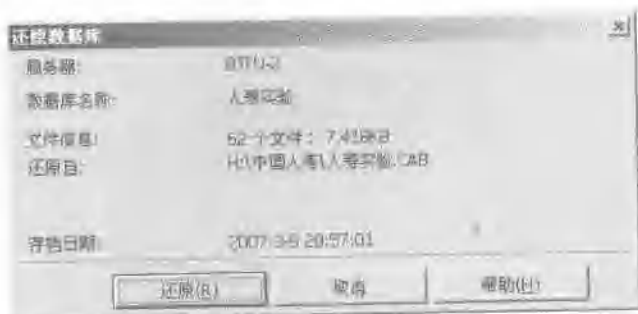


图 8.19 “还原数据库”对话框

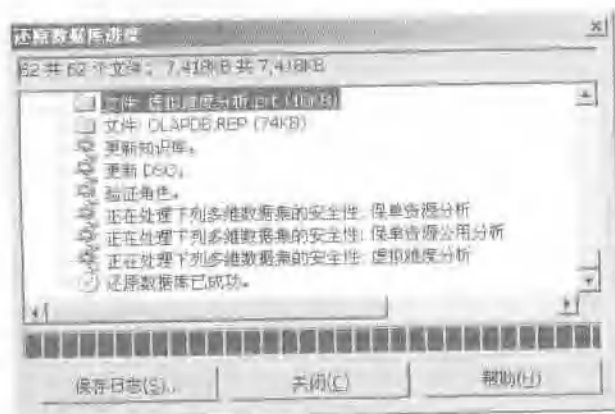


图 8.20 分析数据库的还原

8.4.4 多维数据集的优化

建立好一个多维数据集之后,一项很重要的工作就是考虑存储空间和处理效率之间的平衡点。当“汇总项目”比较多时,处理速度会相对加快,而存储空间则会增大。为了在节约存储空间的情况下提升处理效率,菜单选项需要进行多维数据集的优化工作。

第1步 使用“使用分析”菜单选项。在进行优化之前,可以使用“使用分析”菜单选项来查看目前的处理状况。分析报表的内容是由“查询日志文件”产生的。选择要分析的多维数据集(这里选择“保单资源分析”),然后单击鼠标右键,选择“使用分析”菜单选项,如图 8.21 所示。

屏幕显示出“使用分析向导”对话框,如图 8.22 所示。

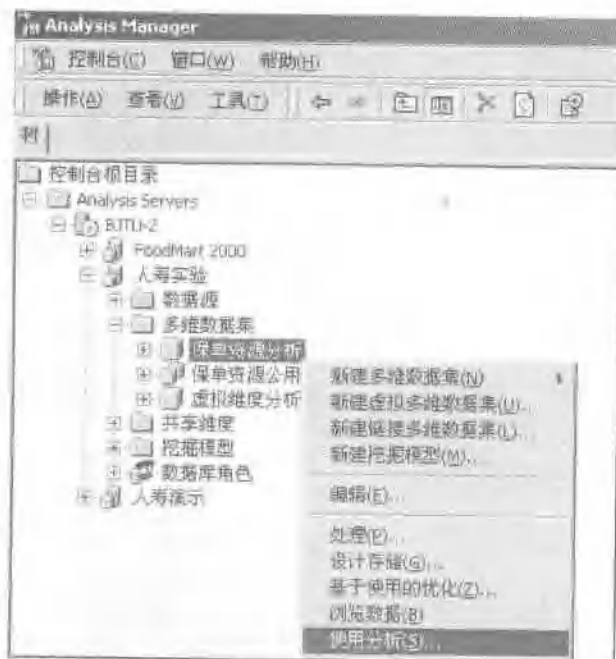


图 8.21 使用“使用分析”菜单选项

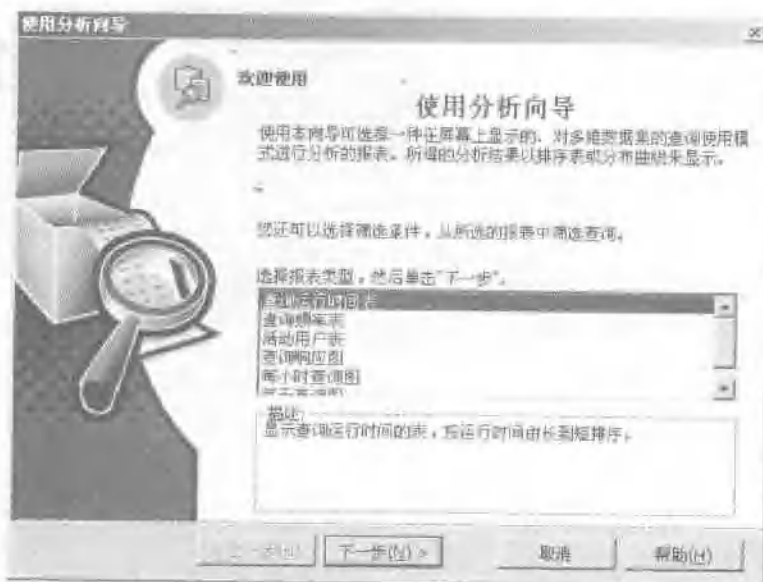


图 8.22 “使用分析向导”对话框

在“使用分析向导”对话框中,共有6个分析报表:查询运行时间表、查询频率表、活动用户表、查询响应图、每小时查询图、每天查询图。

选择其中的一个报表,如“查询运行时间表”,单击“下一步”按钮,进入到“选择此报表的筛选条件”对话框,如图8.23所示。

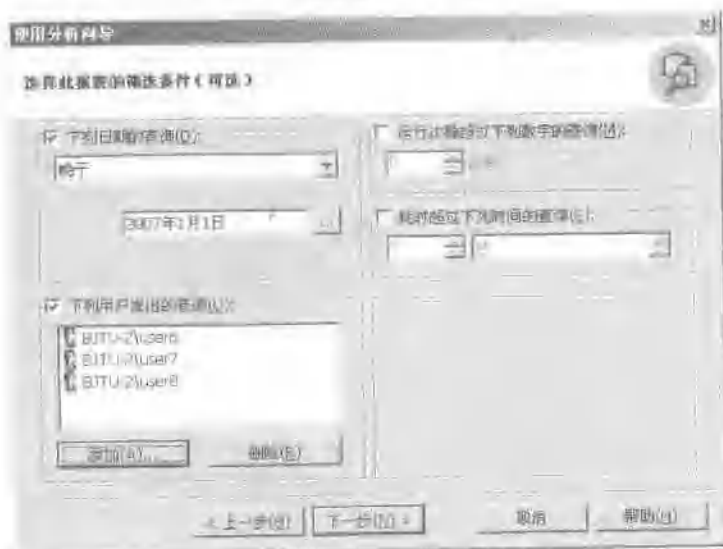


图 8.23 “选择此报表的筛选条件”对话框

在该对话框中,可以设定4种筛选条件:按照查询时间进行筛选、按照允许次数进行筛选、按照查询时间长度进行筛选以及按照查询用户进行筛选。

设置好筛选条件以后,单击“下一步”按钮查看报表分析结果。同样,其他5个报表的分析过程与之类似,读者可根据筛选条件进行多次尝试。

第2步 多维数据集的优化。进行了“使用分析”之后,对目前的运行状况有了初步的了解,下面可以进行多维数据集的优化处理。

在“Analysis Servers”窗口中,可以进行基于使用的多维数据集的优化工作。选择要优化的多维数据集,单击鼠标右键,选择“基于使用的优化”菜单选项,进入“基于使用的优化向导”对话框,单击“下一步”按钮,打开“选择要优化的查询”对话框,如图8.24所示。

该对话框提供了4种查询条件,这4种查询条件与“选择此报表的筛选条件”对话框中的4种筛选条件相一致,使用者可以选择其中的一项,也可以混合使用。系统根据查询条件的设置来决定要调整的分析项目。通常,根据“使用分析向导”



图 8.24 “选择要优化的查询”对话框

对话框中产生的报表,来设定比较重要的查询条件。

设定好查询条件以后,单击“下一步”按钮,显示出查询的结果。单击“下一步”按钮,进入到“选择数据存储类型”对话框,如图 8.25 所示。

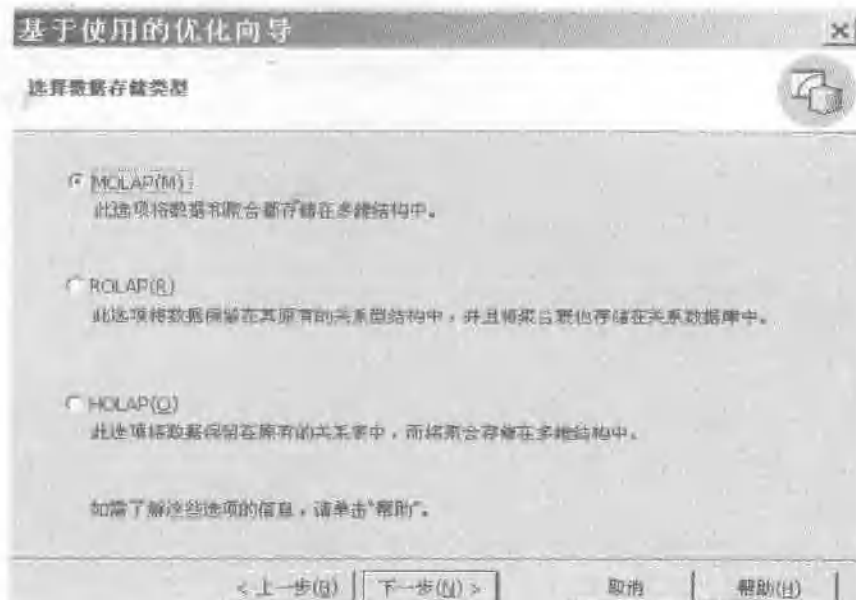


图 8.25 “选择数据存储类型”对话框

在该对话框中提供了三种存储模式,在前面已经介绍过这三种模式,这里选择“MDLP”模式,然后单击“下一步”按钮,设置聚合选项。该步骤在前面也已经介绍过,这里不再重复。单击“下一步”按钮,开始对多维数据集进行优化处理。

8.4.5 系统环境的设置

系统环境的设置是指设置 Analysis Servers 的工作环境,这项工作主要是在所连接的服务器的“属性”中加以设置。

第 1 步 常规性设置。在“Analysis Servers”窗口中,鼠标右键单击所连接的服务器,选择“属性”选项,进入“属性”对话框,如图 8.26 所示。

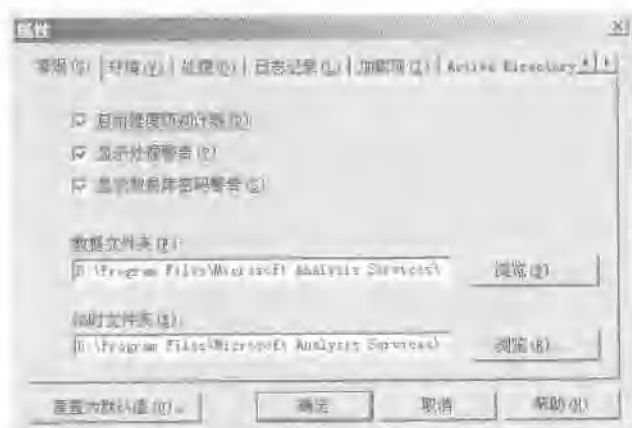


图 8.26 “属性”对话框

在该对话框中共有 6 个选项,下面将依次对比较常用的选项加以介绍。

在“常规”选项卡中,有 3 个可选项。

(1) 启用维度级别计数:使得 Analysis Servers 可以随时计算新维度级别中的成员数目。并且,当成员较多的层次被放置到成员较少的层次的上方时,发出系统警告。

(2) 显示处理警告:当用户在一个多维数据集中进行修改或更新,或是建立新的多维数据集之后,需要进行存储设计,否则产生警告信息。

(3) 显示数据库密码警告:当数据库的连接属性中设置了用户名和密码,并且把密码存储在 Analysis Servers 中,会发出警告信息。

对于以上三种参数设置,基于系统安全性的考虑,可以选择,也可以不选择。

文件夹的设置是系统数据默认的存放路径,可以设置新的文件夹,同时临时

文件夹也会做出相应的改变。

第2步 环境的设置。完成常规设置后,可以进行运算环境的设置,打开如图 8.27 所示的“环境”选项卡。

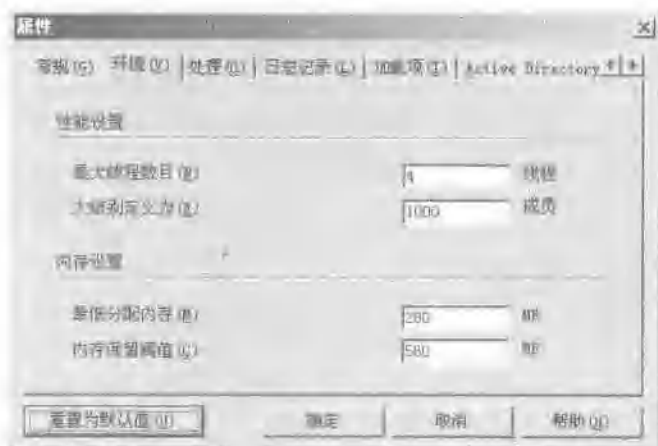


图 8.27 “环境”选项卡

在“环境”选项卡中,可以输入一些性能设置和内存设置的参数,包括最大线程数目,大级别定义为,最低分配内存和内存保留阈值。

- (1) 最大线程数目:设置服务器中可以同时使用 CPU 的线程数目。
- (2) 大级别定义为:设置一个维度层次被视为大级别所必须含有的成员数目。
- (3) 最低分配内存:设置最少配置到 Analysis Servers 的内存总量。
- (4) 内存保留阈值:设置内存的临界值,接近此临界值时,就开始降低内存的使用需求。

第3步 处理的设置。在“属性”对话框中的“处理”选项卡中,进行“处理”设置,如图 8.28 所示。

进行“处理”设置,可以改善系统的优化配置,提高处理的效率。“处理”选项卡包括以下参数。

- (1) 预读缓冲区大小:设置读取数据库时,可以写入内存中的数据的最大量。
- (2) 处理缓冲区大小:设置内存处理多少数据后进 I/O 处理。
- (3) 服务器超时:设置 Analysis Servers 处理数据对象时,等待查询数据源结果的最长时间。

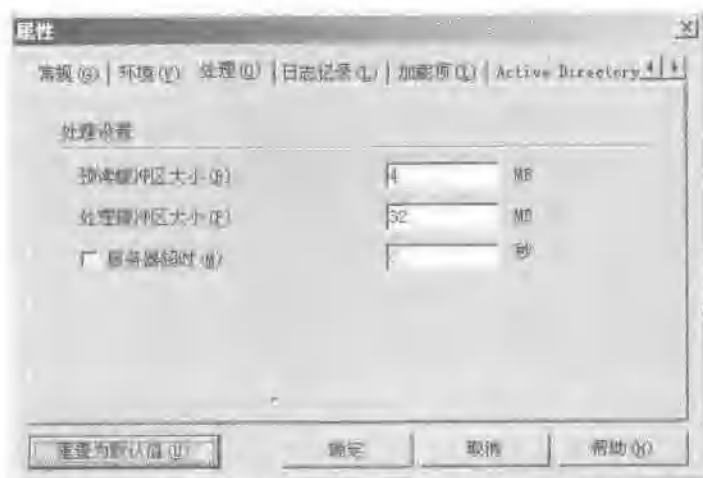


图 8.28 “处理”选项卡

除此之外,在服务器的“属性”设置中还包含一些其他的设置,如在“日志记录”和“加载项”等选项卡中的设置,这里不再一一详述。

8.5 扩展实验

请读者根据本章内容的学习和演示,独立完成下述练习。

1. 数据库角色的管理

创建名称为“管理员”的“数据库角色”,并向这一角色中添加用户和组,分别添加“BJTU\user007”和“BJTU\user008”两个用户;然后,对“管理员”角色进行编辑。添加新的用户“BJTU\user009”,并删除该用户;对多维数据集“法人客户价值分析保单数量”、“法人客户价值_虚拟维度”和“业务经理专用”的用户和组进行访问权限的设置;对“挖掘模型”和“维度”进行权限的设置。这里,把“开户年_法人客户”维度的规则设置为“自定义”,对该“维度”的安全性进行自定义设置,在“基本”选项卡中,将“选择可见级别”下的“顶层”设为“全部”,将底层设为“客户名称”,选择“2003”作为成员,“高级”和“公用”两个选项卡使用默认设置。当然,也可以删除新建的“管理员”的“数据库角色”。

2. 多维数据集角色管理

为多维数据集“法人客户价值分析保单数量”添加和删除管理角色。这里,没

有具体要求,读者可自行设计多维数据集管理角色的添加和删除。编辑“管理员”角色,对“摘要”、“成员资格”、“维度”、“单元”和“选项”卡进行编辑,其中“成员资格”和“维度”卡的编辑与“编辑数据库角色”对话框中的一致,在“单元”选项卡中将“读取”设置为“自定义”,并使用 MDX 自定义权限,其他选项使用默认设置。

3. 数据库的存档与恢复

将“人寿实验”数据库存档,存到“F:\人寿实验.CAB”,临时文件夹为“F:\新建文件夹\”;然后,对保存好的“人寿实验.CAB”进行数据库还原。

4. 多维数据集的优化

对“法人客户价值分析保单数量”多维数据集进行优化,管理员“BJTU \ user007”用户发出晚于“2005 年 1 月 1 日”的“运行时间”,对报表进行筛选输出。同样,可以使用“基于使用的优化”对该多维数据集进行优化。

5. 系统环境的设置

在服务器的“属性”对话框中对“常规”、“环境”和“处理”的选项卡进行设置,要求与实验一致。

本章小结

本章主要向读者介绍了如何进行数据库角色管理、如何对数据库进行备份和恢复、如何进行多维数据集的优化以及系统环境的高级设置等内容。本章内容的主要学习目的是,使读者在建立数据库系统之后能够进一步管理该系统,同时对系统实现一定程度的优化,使系统的使用效率达到最优,并使其安全性能得到进一步的保障。

附录 A 常用 MDX 函数介绍

MDX(Multidimensional Expressions, 多维表达式), 是支持多维度对象的数据定义与运用的一种语言。MDX 在许多方面与 SQL(Structured Query Language, 结构化查询语言)的语言类似, 二者都可以用来查询, 但 MDX 处理的对象是经过 OLAP 处理的多维数据集。

在语法结构上, 二者都包含“筛选对象”(select 子句)、“数据源”(from 子句)、可选的筛选条件(where 子句)。与 SQL 语言一样, MDX 语言也包含功能强大的函数, 以协助数据处理与数据挖掘, 并提供用户自定义函数功能来扩充 MDX 的运用范畴。MDX 也提供“数据定义语言(DDL)”语法来管理数据结构, 包含建立、删除多维数据集、维度、度量值以及其他附属对象的命令。但是, 与 SQL 语句不同的是, MDX 的筛选对象是多维数据集子集, 多维数据集子集由维度中的成员表达式构成, 一个 MDX 查询中最多可以指定 128 个坐标轴。MDX 的数据源是多维数据集。MDX 查询中的条件也是针对成员表达式的。下面将着重讲述 MDX 的主要函数, 包括成员函数、集合函数、维度函数、级别函数和数值函数。

A.1 成员函数

1. 成员的概念

“成员”是“维度”中的一个项目。在数据库中, 成员与维度的关系, 就是某一字段中的“值域”(不重复出现的值的集合)与该“字段”的关系。“成员”是 MDX 中用来描述多维数据集的最低参照层次。

例如, 在本书的范例中, 曾建立“时间”维度, 由“年”、“季度”、“月”、“日”构成。一般使用如下方式表示“年”与“季度”成员:

[时间].[年]

[时间].[季度]

如果要表达 2005 年的第 1 季度的 3 月, 表达方式为:

[时间].[所有时间].[2005].[第 1 季].[03]

如果要表示 2004 年的成员, 有下列两种方式:

[时间].[2004]

[时间].[年]&[2004]

其中,“&”字符可在任何 MDX 表达式中用来指定成员的索引键。

2. 成员函数集

1) ancestor 函数

ancestor 函数用来返回某个成员的上层层次或者距此成员指定距离的层次。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 4 月的上一层次(即第 1 季度),表达方式如下:

ancestor([时间].[所有时间].[2003].[第 2 季].[04],1)

2) firstchild 函数

firstchild 函数用来返回成员的第 1 个子成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2003 年 7 月的方式有两种。因为 7 月是第 3 季度的第 1 个成员,所以这两种目的相同但表达不同的方式如下:

[时间].[所有时间].[2003].[第 3 季].[07]

[时间].[所有时间].[2003].[第 3 季].firstchild

3) lastchild 函数

lastchild 函数用来返回指定成员的最后一个子成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2005 年 6 月的方式有两种。因为 6 月是第 2 季度的最后一个成员,所以这两种目的相同但表达不同的方式如下:

[时间].[所有时间].[2005].[第 2 季].[06]

[时间].[所有时间].[2005].[第 2 季].lastchild

4) firstsibling 函数

firstsibling 函数用来返回成员的父成员的第 1 个子成员,即返回与该成员同一层次上的第 1 个成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2003 年 7 月的方式有三种,因为 7 月是第 3 季度的第 1 个成员,所以这三种目的相同但表达不同的方式如下:

[时间].[所有时间].[2003].[第 3 季].[09].firstsibling

[时间].[所有时间].[2003].[第 3 季].[07]

[时间].[所有时间].[2003].[第3季].firstchild

5) lastSibling 函数

lastSibling 函数用来返回成员的父成员的最后一个子成员,即返回与该成员同一层次上的最后一个成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2005 年 6 月的方式有三种,因为 6 月是第 2 季度的最后一个成员,所以这三种目的相同但表达不同的方式如下:

[时间].[所有时间].[2005].[第2季].[05].lastSibling

[时间].[所有时间].[2005].[第2季].[06]

[时间].[所有时间].[2005].[第2季].lastchild

6) parent 函数

parent 函数用来返回一个成员的父成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2005 年第 2 季度成员,可以由第 2 季度的任一成员(4、5、6)3 个月来表达。以下表达式具有相同的意义。

[时间].[所有时间].[2005].[第2季].[06].parent

[时间].[所有时间].[2005].[第2季]

7) members 函数

成员函数集中的 members 函数用来返回由具有 MDX 格式的字符串指定的成员。例如:

members("[2003]")返回

[时间].[所有时间].[2003]

8) currentMember 函数

currentMember 函数用来返回迭代过程中指定的维度或层次结构的当前现行成员。

例如,描述维度中当前现行的成员名称为“新单保费”的表达方式如下:

[measures].currentMember.name = "新单保费"

又如,在建立自定义汇总公式时,以“季度”层次成员中的第 1 个子成员与最后一个子成员的和为汇总的值,即每个季度第 1 个月与最后一个月的和,则其表达式如下:

[时间].currentMember.firstChild + [时间].currentMember.lastchild

9) nextmember 函数

nextmember 函数用来返回指定成员在同一层次(维度)中的下一个成员。

例如,在“时间”维度中,包含“年”、“季度”、“月”、“日”层次。要表达 2005 年第 2 季度成员,可以由以下方式表示:

```
[时间].[所有时间].[2005].[第 1 季].nextmember  
[时间].[所有时间].[2005].[第 2 季]
```

10) prevmember 函数

prevmember 函数用来返回指定成员在同一层次(维度)中的前一个成员。同样以“时间”维度为例,以下表达方式同样可以表示第 2 季度的成员:

```
[时间].[所有时间].[2005].[第 3 季].prevmember  
[时间].[所有时间].[2005].[第 2 季]
```

11) lead 函数

lead 函数用来返回在所在层次(维度)中位置比指定成员靠后且靠后位数为指定位数的成员。因此,在 lead 后面,有一个数字表达式<numeric expression>。如果表达式为正,则是往前算;如果为负,则往后算。例如,lead(1)相当于 nextmember,lead(-1)相当于 prevmember。以“时间”维度为例,以下表达方式可以表示 2005 年第 3 季度成员:

```
[时间].[所有时间].[2005].[第 1 季].lead(2)  
[时间].[所有时间].[2005].[第 2 季].lead(1)  
[时间].[所有时间].[2005].[第 4 季].lead(-1)
```

12) lag 函数

lag 函数用来在该成员所在层次(维度)中位置比指定成员靠前且靠前位数为指定位数的成员。当查找的方向相反时,lag(n)相当于 lead(-n)的查找结果。因此,在 lag 后面,也有一个数字表达式<numeric expression>。如果表达式为正,则是往后算;如果为负,则往前算。例如,lag(1)相当于 prevmember,lead(-1)相当于 nextmember。以“时间”维度为例,以下表达方式可以表示 2005 年第 1 季度成员:

```
[时间].[所有时间].[2005].[第 4 季].lag(3)  
[时间].[所有时间].[2005].[第 3 季].lag(2)  
[时间].[所有时间].[2005].[第 2 季].lag(1)
```

A.2 集合函数

集合函数主要是用来返回 MDX 中的集合,使用户易于建立动态定义的集合,

快速建立可以重复使用的命名集。其中,最常使用的集合函数是 `members` 函数。

1) `members` (集) 函数

该函数用来返回某个维度、级别或层次结构中的成员集。例如,将“客户”维度中的“客户名称”层次中的所有成员作为一个集合来返回,并且以“列”的格式来显示,具体表达方法如下:

```
{[客户].[客户名称].members} on rows
```

又如:

```
{[险种_险种类型分类].[险种名称].members} on rows
```

```
{[生效时间].[年].members} on columns
```

2) `bottomcount` 函数

该函数根据指定表达式或度量值,从集合中最小的值开始,按升序对集合进行排序,并返回指定数目的最小值元组。基本语法如下:

`bottomcount`(<集合>, 指定数目, 表达式或度量值)

例如,返回业务员级别 03 中客户数量最少的 5 位业务员,表达式如下:

```
bottomcount([业务员等级].[级别 03].members,5,[客户数量])
```

3) `bottompercent` 函数

该函数根据指定表达式或度量值,从集合中最小的值开始,按升序对集合进行排序,并返回一个最小值元组集,该元组集的累积合计等于或小于指定的百分比。基本语法如下:

`bottompercent`(<集合>, 指定百分比, 表达式或度量值)

例如,返回业务员级别 03 中客户数量最少的前 20% 的业务员,表达式如下:

```
bottompercent([业务员等级].[级别 03].members,20,[客户数量])
```

4) `bottomsum` 函数

该函数根据指定表达式或度量值,从集合中最小的值开始,按升序对集合进行排序,并返回一个最小值元组集,该元组集的合计等于或小于指定值。基本语法如下:

`bottomsum`(<集合>, 指定数值, 表达式或度量值)

例如,返回业务员级别 03 中最少客户数量的业务员名称的汇总,从最少的业务员开始,直到客户数量达到 200 为止。具体表达式如下:

```
bottomsum([业务员等级].[级别 03].members,200,[客户数量])
```

5) `children` 函数

该函数用来返回某一指定成员的子成员。例如,要返回 2005 年第 2 季度的所

有子成员,则表达式如下:

[时间].[所有时间].[2005].[第 2 季].children

6) crossjoin 函数

该函数用来返回一个或多个集合的叉积。在进行多维数据查询时,使用 crossjoin 函数,可以实现多维数据的嵌套查询。该函数不仅可以用在“rows”上,也可以用在“columns”上。基本语法如下:

crossjoin(<集合 1>,<集合 2>)

例如,要查询不同单位各年在 3 种不同险种类别中的保险金额,则表达式如下:

crossjoin([单位性质].[单位性质名称].members,[保单类型].[险种名称].members)

另一种直接表达方法如下:

<集合 1>* <集合 2>

因此,上述实例也可以如下表达:

[单位性质].[单位性质名称].members*[保单类型].[险种名称].members

7) filter 函数

该函数用来返回根据搜索条件对指定集合的每个成员进行筛选后得到的成员集合。基本语法如下:

filter(<集合>,<筛选条件>)

例如,要找出 2005 年第 2 季度级别 03 的客户数量比第 1 季度衰减的所有业务员的名,则表达式如下:

filter([业务员等级].[级别 03].members,([客户数量],[时间].[所有时间].[2005].[第 1 季])<([客户数量],[时间].[所有时间].[2005].[第 2 季]))

如果要找出 2005 年连续两个季度客户数量都衰减的数据,则要配合 and 关键字来书写如下表达式:

filter([业务员等级].[级别 03].members,([客户数量],[时间].[所有时间].[2005].[第 1 季])<([客户数量],[时间].[所有时间].[2005].[第 2 季]) and([客户数量],[时间].[所有时间].[2005].[第 2 季])<([客户数量],[时间].[所有时间].[2005].[第 3 季]))

8) head 函数

该函数用来返回集合中从集合的第 1 个成员开始算起,指定数目的前几个元

素,同时保留重复项。基本语法如下:

```
head (<集合> , <数字表达式> )
```

例如,要得到时间中上半年的成员作为数据分析的依据,则表达式如下:

```
head ([时间].[月].members,6)
```

```
head ([时间].[季度].members,2)
```

9) order 函数

该函数按照指定的运算方式或字段对特定集合的成员进行排序,可以选择保留或打乱原有的层次结构。基本语法如下:

```
order(<集合> ,运算方式或字段,排序方式)
```

使用 order 函数进行排序时,不管是层次式还是非层次式,都可以进行“升序排序”和“降序排序”,但是关键字不同。在层次式中,用“asc”表示升序排序,用“desc”表示降序排序;而在非层次式中,用“basc”表示升序排序,用“bdesc”表示降序排序。系统默认值是“asc”升序排序。

例如,希望得到以 2005 年第 2 季度“客户数量”为分类标准的“保单类型”中“个单”的数据,具体表达式如下:

```
order([保单类型].[所有保单类型].[个单].members,([2005].[第 2 季],  
[客户数量]),desc) on rows
```

10) subset 函数

该函数可以从指定的集合中,根据设置的参数返回特定“分割”的元组的子集。基本语法如下:

```
subset (<集合> , <开始分割> , <返回个数> )
```

本函数返回的是 <集合> 中由 <开始分割> 开始的 <返回个数> 个 tuple。
<开始分割> 以 0 为基础,即 0 对应于集合中的第 1 个 tuple,1 对应于第 2 个 tuple,依次类推。如果没有指定 <返回个数>,则会返回从 <开始分割> 集合中开始到最后一个所有的 tuple。

例如,要仅显示第 2~4 季度的数据,则表达式如下:

```
subset( {[时间].[季度].members} ,1,3) on columns
```

当然,subset 函数还可以与其他函数配合使用,来进行特定的查询。

又如,配合 order 函数,可以得到客户数量在 5~10 名级别为 03 的业务员名称数据作为“列”的分类标准,具体表达式如下:

```
subset( {order([业务员等级].[级别 03].members,[客户数量],desc) } ,4,
```

6) on rows

11) tail 函数

该函数从集合的最后一个 tuple 开始,返回特定个 tuple。基本语法如下:

tail(<集合>, <返回个数>)

本函数从<集合>的最后一个 tuple 开始,返回<返回个数>个 tuple。<返回个数>的默认值是 1,如果<返回个数>的值小于 1,则返回空集合;相反,如果<返回个数>的值超过集合中的 tuple 总数,则返回原集合。

例如,要以“列”的分类方式,显示第 3 季度、第 4 季度的数据,表达式如下:

tail({[时间].[季度].members},2) on columns

同样,tail 函数也可以与其他函数配合使用,以进行特定的查询。

又如,配合 order 函数,可以得到客户数量是“倒数前三名”的级别为 03 的业务员名称数据作为“列”的分类标准,具体表达式如下:

Tail({order([业务员等级].[级别 03].members,[客户数量],desc)},3)

on rows

另外,用 bottomcount 函数也可以实现上述功能,表达式如下:

bottomcount([业务员等级].[级别 03].members,3,[客户数量]) on rows

12) topcount 函数

该函数根据特定表达式或度量值,按降序对集合进行排序,从集合中最大的值开始,返回指定数目的最大值元素。基本语法如下:

topcount(<集合>,特定数目,表达式或度量值)

例如,要返回客户数量最多的前 5 名级别为 03 的业务员数据,表达式如下:

topcount([业务员等级].[级别 03].members,5,[客户数量])

13) toppercent 函数

该函数根据特定表达式或度量值,按降序对集合进行排序,从集合中最大的值开始,返回“累积总和达到指定百分比”的最大值元素。基本语法如下:

toppercent(<集合>,指定百分比,表达式或度量值)

例如,要返回客户数量前 30% 的级别是 03 的业务员的信息,表达式如下:

toppercent([业务员等级].[级别 03].members,30,[客户数量])

14) topsum 函数

该函数根据特定表达式或度量值,按降序对集合进行排序,从集合中最大的值开始,返回“累积总和达到指定数值”的最大值元素。基本语法如下:

topsum(<集合>,指定数值,表达式或度量值)

例如,要得到在级别为 03 的由客户数量最多的开始汇总,直到客户数量总和达到 5000 为止的业务员,则表达式如下:

topsum([业务员等级].[级别 03].members,5000,[客户数量])

A.3 维度函数与级别函数

该函数是从层次结构、级别或成员返回对维度或者级别的引用。维度函数、层次结构函数和级别函数对遍历多维表达式 (MDX) 中的多维结构非常有用。通常,将此类函数和其他函数一起使用,以获得有关维度、层次结构或级别的成员信息。

1) dimension 函数

该函数用来返回包含指定的层次结构、级别或成员的维度。例如:下列示例返回维度“时间”。

[2003].dimension

[年].dimension

[quarter 1].dimension

[季度].dimension

2) level 函数

level 函数返回成员的级别。基本语法如下:

<member>.level

例如在本书实验中,要从各季度的角度查看 2005 年的新单保费,语句如下:

with member[measures].[新单保费] as'[time].&[2005].&[q1].level.name'

返回的是 2005 年四个季度的新单保费数据。

A.4 数值函数

在 MDX 函数中,多数是用来定义处理的对象,如“成员函数”、“维度函数”、“层次函数”等;而与运算有关的函数,最主要是“数值函数”,此类函数可以用来执行各种汇总和统计的计算。

1) aggregate 函数

aggregate 函数返回一个标量值,该标量值由对指定集的元组的度量值聚合而得,或通过使用指定的数值表达式对指定集的元组计算而得。基本语法如下:

```
aggregate( <set> [ , <numeric expression> ] )
```

此函数不能用在计算成员上。

以本书实验为例,在保单资源分析中,利用该函数建立新的计算成员。将[时间]维度中的 2003 年、2004 年、2006 年的平均数据定义为[历史年份平均],2006 年定义为[当前年份]。其定义语句如下:

```
with
```

```
member[时间].[历史年份平均] as 'aggregate({[时间].[所有时间].[2003],[时间].[所有时间].[2004],[时间].[所有时间].[2005]})/3'
```

```
member[时间].[当前年份] as '[时间].[所有时间].[2006]'
```

在应用程序界面中输入如下的语句:

```
with
```

```
member[时间].[历史年份平均] as 'aggregate({[时间].[所有时间].[2003],[时间].[所有时间].[2004],[时间].[所有时间].[2005]})/3'
```

```
member[时间].[当前年份] as '[时间].[所有时间].[2006]'
```

```
select
```

```
{[measures].members} on axis(0)
```

```
{[时间].[历史年份平均],[时间].[当前年份]} on axis(1)
```

```
from 保单资源分析
```

在该语句中,首先定义了两个计算成员[历史年份平均]和[当前年份],然后,查询所有历史年份的平均度量值数据和当前年份的度量值数据。

2) count 函数

count 函数用来计算维度或数据集中成员的个数,基本语法如下:

```
count( <集合> , excludeempty | includeempty )
```

其中,可以用 excludeempty 或 includeempty 参数排除或包含空单元格。这对于某些数据的计算很重要。

结合本书实验中的保单资源分析,这里用该函数计算业务员的数量。建立新的计算成员[业务员个数]以统计业务员的数量。在应用程序界面中,输入以下语句:

```
with
```

```
member[区域_机构].[业务员].[业务员个数] as 'count([区域_机构].[业务员].[业务员名称].members, excludeempty)'
```

```
select
```

```
    {[时间].[年].members} on axis(1)
```

```
    {[区域_机构].[业务员].[业务员个数]} on axis(0)
```

```
from 保单资源分析
```

在该语句中,首先定义了新的计算成员[业务员个数],用来计算业务员的数量。然后,查询出每个年份业务员的数量。其中,excludeempty 参数排除了空集合,返回了实际的业务员数量。

3) iif 函数

进行比较式的判断,返回当判断视为真或假时相应的数值或字符串。基本语法如下:

数字

```
iif( <logical expression> , <numeric expression1> , <numeric expression2> )
```

如果 <logical expression> 取值为 true,则此函数返回 <numeric expression1>; 否则,返回 <numeric expression2>。

字符串

```
iif( <logical expression> , <string expression1> , <string expression2> )
```

如果 <logical expression> 取值为 true,则此函数返回 <string expression1>; 否则,返回 <string expression2>。

例如,在本书实验中希望能建立一个计算成员,名称为“绩优”,用来判断实际收入是否大于 2000,可以通过以下的方式来建立。

```
with
```

```
member[measures].[绩优] as iif (measures.[实际收入] > 2000, "是", "否")
```

4) avg 函数

针对数据集进行数值平均来新建计算成员,基本语法如下:

```
avg( <set> [ , <numeric expression> ] )
```

继续以本书实验案例加以说明,如果希望计算 2003 年、2004 年和 2005 年的交易数据的平均值,则一样是用 with、member 语法,将原来的 aggregate 函数,以 avg 函数来完成,语句如下:

```
with
```

```
member[时间].[历史年份平均] as avg( {[时间].[所有时间].[2003],[时间].[所有时间].[2004],[时间].[所有时间].[2005]})
```

5) max 函数

max 函数返回集合上的数字的最大值。语法如下：

max(< set > [, < numeric expression >])

max 函数返回数值表达式的最大值,该表达式在 < numeric expression > 中指定,对在 < set > 中指定的集合进行求值。例如,在本书实验中,求 2003 年、2004 年、2005 年、2006 年这 4 年中“个人保单总支出”最大的那一年的数据,语句如下:

max({2003,2004,2005,2006} , [个人保单总支出])

系统返回 2005 年的个人保单总支出数据。

6) min 函数

min 函数返回集合上数字的最小值。基本语法如下：

min(< set > [, < numeric expression >])

min 函数返回数值表达式的最小值,该表达式在 < numeric expression > 中指定,对在 < set > 中指定的集合进行求值。

例如,在本书实验中,求 2003 年、2004 年、2005 年、2006 年这 4 年中“个人保单总支出”最小的那一年的数据,语句如下:

min({2003,2004,2005,2006} , [个人保单总支出])

系统返回 2006 年的个人保单总支出数据。

7) sum 函数

sum 函数返回集合上的数值的汇总。基本语法如下：

sum(< set > [, < numeric expression >])

例如,在本书实验中,求 2003 年、2004 年、2005 年、2006 年这 4 年中“个人保单总支出”的总和,语句如下:

sum ({2003,2004,2005,2006} , [个人保单总支出])

系统返回这 4 年的个人保单总支出数据的总和。

8) median 函数

median 函数返回集合上数字的中间值。基本语法如下：

median(< set > [, < numeric expression >])

median 函数返回数值表达式的中值,该表达式在 < numeric expression > 中指定,对在 < set > 中指定的集合进行求值。

例如,在本书实验中,求 2003 年、2004 年、2005 年这 3 年中“个人保单总支出”居中的那一年的数据,语句如下:

```
median({2003,2004,2005},[个人保单总支出])
```

系统返回 2004 年的个人保单总支出数据。

附录 B “人寿实验 . CAB”数据库文档的还原

在完成整个实验后,可以对“人寿实验 . CAB”数据库进行还原,以检验自己的操作结果是否正确。当然,如果在操作过程中遇到解决不了的问题时,也可以及时将该数据库还原,避免不必要的实验错误的发生,以节省实验时间,提高实验效率,得到正确的实验结果。下面介绍还原实验数据库的具体步骤。

第1步 单击“开始”→“程序”→“Microsoft SQL Server”→“Analysis Services”→“Analysis Manager”菜单选项,如图 B.1 所示。

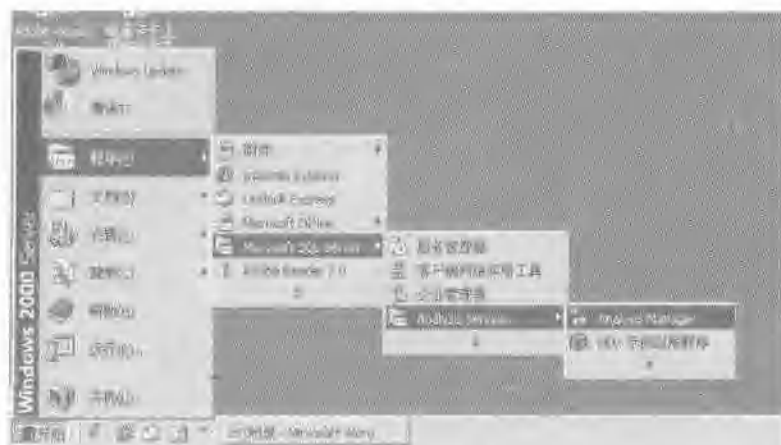


图 B.1 选择菜单选项

第2步 在“Analysis Manager”窗口中,单击“控制台根目录”下“Analysis Servers”文件夹图标前面的“+”号,出现名称为本计算机名称的服务器。然后,用鼠标右键单击本计算机的“服务器”,选择“还原数据库”,出现如图 B.2 所示的“打开存档文件”对话框。

第3步 在“打开存档文件”对话框中找到已经保存好的文档(在第8章中已详细介绍了数据库的存档),选中“人寿实验 . CAB”,单击“打开”按钮。出现“还原数据库”对话框,如图 B.3 所示。

第4步 在“还原数据库”对话框中,单击“还原”按钮,系统将自动处理数据库的还原任务。数据库还原成功后,单击“关闭”按钮,完成“人寿实验”的还原。“还原数据库进度”对话框如图 B.4 所示。



图 B.2 “打开存档文件”对话框



图 B.3 “还原数据库”对话框

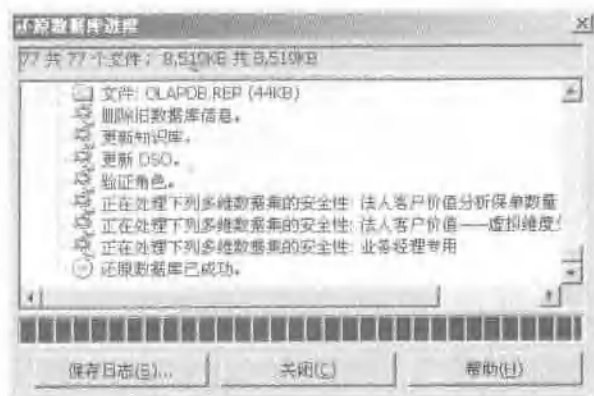


图 B.4 “还原数据库进度”对话框

附录 C 实验报告格式及说明

实 验 报 告

实验项目名称 _____
所属课程名称 _____
实 验 类 型 _____
实 验 日 期 _____
班 级 _____
学 号 _____
姓 名 _____
成 绩 _____

实验概述:	
【实验目的及要求】	
【实验原理】	
【实验准备工作】	
实验内容:	
【实验方案设计】	
【实验过程】(实验步骤、记录、数据、分析)	
【结论】(结果)	
【实验收获、疑难及需解决问题】	
指导教师评语及成绩:	
评语:	
成绩:	指导教师签名:
	批阅日期:

实验报告说明

(1) 实验项目名称: 要用最简练的语言反映实验的内容。要求与实验指导书中相一致。

(2) 实验类型: 一般需说明是验证型实验还是设计型实验, 是创新型实验还是综合型实验。

(3) 实验目的与要求: 目的要明确, 要抓住重点, 符合实验指导书中的要求。

(4) 实验原理: 简要说明本实验项目所涉及的理论知识。

(5) 实验环境: 实验用的软、硬件环境(配置)。

(6) 实验方案设计(思路、步骤和方法等): 这是实验报告极其重要的内容。概括整个实验过程。

① 对于操作型实验, 要写明依据何种原理、操作方法进行实验, 要写明需要经过哪几个步骤来实现其操作。

② 对于设计型和综合型实验, 在上述内容基础上还应该画出流程图、设计思路和设计方法, 再配以相应的文字说明。

③ 对于创新型实验, 还应注明其创新点、特色。

(7) 实验过程(实验中涉及的记录、数据、分析): 写明具体上述实验方案的具体实施, 包括实验过程中的记录、数据和相应的分析。

(8) 结论(结果): 即根据实验过程中所见到的现象和测得的数据做出结论。

(9) 小结: 对本次实验的心得体会、思考和建议。

(10) 指导教师评语及成绩: 指导教师依据读者的实际报告内容, 用简练的语言给出本次实验报告的评价和价值。

参考文献

- [1] 姚家奕等. 多维数据分析原理与应用. 北京:清华大学出版社,2004.
- [2] 林宏谕. SQL 2000 决策分析 OLAP 建置与应用. 北京:中国铁道出版社,2001.
- [3] 蒋秀英. SQL Server 2000 数据库与应用. 北京:清华大学出版社、北京交通大学出版社,2006.
- [4] 施伯乐. 数据库与智能数据分析--技术、实践与应用. 上海:复旦大学出版社,2003.
- [5] 罗运模. SQL Server 2000 数据仓库应用与开发. 北京:人民邮电出版社,2001.
- [6] 章立民. SQL Server 2000 完全实战--数据转换服务(DTS). 北京:中国铁道出版社,2002.
- [7] 林宇等. 数据仓库原理与实践. 北京:人民邮电出版社,2003.
- [8] 池太藏. 数据仓库结构设计及实施:建造信息系统的金字塔. 北京:电子工业出版社,2005.
- [9] 苏新宁等. 数据仓库和数据挖掘. 北京:清华大学出版社,2006.
- [10] 陈京民等. 数据仓库与数据挖掘技术. 北京:电子工业出版社,2002.
- [11] Kambayashi, Y, Winiwarter, Werner, Arikawa, Masatoshi. Data warehousing and knowledge discovery. New York: Springer-Verlag, 2001.
- [12] Mahlach, Efrem G. Decision support and data warehouse systems. Beijing: Tsinghua University Press, 2001.
- [13] Singh, Harry. Data warehousing: concepts, technologies, implementations, and management. Upper Saddle River, N. J.: Prentice-Hall PTR, 1998.
- [14] 克伦克著,施伯乐译. 数据库处理:基础、设计与实现. 北京:电子工业出版社,2003.
- [15] 徐义峰,徐云青,诸葛理绣. 基于SQL的OLAP多维数据分析. 微机发展, 2005(7).
- [16] 张旭,董有田等. OLAP多维数据分析与应用研究. 黑龙江科技学院学报, 2002(3).
- [17] 李业丽,徐秀花,李丽娟. 基于多维数据模型的数据仓库的构建. 北京印刷学院学报, 2006(3).
- [18] Approach to conceptual data integration for multidimensional data analysis in e-connnerc. Zhang Zhe, Huang Pei. 系统工程与电子技术:英文版, 2006(3).