


第一部分 关系型数据库基础



第 1 章 概 述

学习时间：第一天

地点：老田办公室

人物：老田、小天

本章要点

- 数据库是什么
- 为什么要使用数据库
- 数据库的基本概念
- 数据库的历史
- 数据库的分类
- SQL 语言及 SQL 标准
- 管理工具及其作用
- 针对熟悉工具的一系列小实践

本章学习线路

我们从一个问题来引入数据库是什么、为什么要用数据库、数据库的发展历史、SQL语言的概念和标准，最后将落点放到熟悉我们学习所用的工具使用实践。

全文以问答的方式，依照零基础学员的思维方式一步步地引导和讲解，以防止纯书面化那种严肃的文字风格导致学员无法耐心地去看完所有内容。

本书主要的出发点是引导一个完全没有编程思维的人一步步地具备这种以一个架构师的角度和方式去思考并解决问题的能力。为避免学员因为不熟悉工具而在学习中无谓地浪费时间，因此在本章后面有针对地对工具做了一系列的小实践。之后的章节则以开发和设计为主了。

1.1 什么是数据库

小天:老田,常听说学习编程必须学好数据库。这是为什么啊?数据库是干啥的呢?

老田:额……首先,数据库就是指存放数据的仓库,如同水库是存水的,军火库是放军火的。其次并非所有计算机程序都需要用到数据库,但是如果希望你的程序具备对大量数据的存储、整理、分析等,这就需要涉及数据库了。

小天:明白了,如果我的计算机程序只是存放极其少量的数据,则可以不必使用数据库,而是将数据存放在xml这类的文本文档中就OK了,只有当需要存储大量的数据,并且经常性地对这些数据进行操作时才需要用到数据库。但是我下一个问题又出来了,我问人家,我要学习编程,要学数据库的话学什么。人家好像说有什么SQL Server、MySQL、Oracle、DB2等,这些又是啥呢?不会是我学个编程要学这么多数据库吧?

老田:额……当然不用了,其实几种数据库除了各自的管理有些不同外,对于数据的操作都是大同小异的。因为SQL Server、MySQL、Oracle、DB2这么多数据库管理系统,我们完全可以简单地将它们理解为不同厂商或者组织开发的性质差不多的关系型数据库管理系统。对,它们就是一个数据库管理系统,进一步简化管理为存储和管理数据的工具就行了。

小天:不对啊,老田你在忽悠我吧?存储大量的数据和管理数据我觉得不一定要用数据库啊,我看很多人现在用Excel也挺不错的呀。

1.2 为什么要使用数据库

老田:我们来打个比方,如图1-1所示,你看看是不是类似这样的管理呢?

	A	B	C	D	E	F	G
1	员工姓名	目前所在区域	所属部门	部门主管	电话	所属分公司	公司领导
2	汪静远	北京	人力资源部	高大伟	15855445566	北京分公司	罗天伟
3	朱超	合肥	项目实施部	疯扯扯	15784547788	安徽分公司	黄世元
4	秦良骏	重庆	项目实施部	疯扯扯	36534324	重庆分公司	邹坤

图 1-1 Excel 表

小天:对啊,我觉得一目了然。

老田:那么当遇到频繁地做下面几种操作的时候你烦不烦:

- (1) 添加新员工;
- (2) 部门主管换电话了;
- (3) 部门换主管了;

- (4) 换公司领导了；
- (5) 多个员工换分公司；
- (6) 某个员工换部门；
- (7) 当员工数量超过500人。

接下来我们模拟做一下下面这几件事。

添加新员工：总体来说这个只要员工的归属资料齐全的话还是比较轻松的。万一不齐全就只好慢慢地一行行地查了。

部门主管换电话：如果你还不知道Excel可以全文搜索和替换的话呢，就只能一行行地改了，如果恰恰你之前录入某人资料的时候打错字了的话.....

小天：擦汗ing.....

老田：部门换主管，如果这个主管是直接走人了倒也干脆，否则的话先把新换上来的主管资料逐行改了还要继续为他安排新的工作。

当然，这些如果是数据量少其实都不是大问题了，但这个公司如果是中石油、中移动，那么数据量应该是不上万也是千吧？

还有个最严重的问题，你无法保证不会重复添加数据。

小天：别说啦，你说的这些情况我其实早烦透了，但是这些问题是否在你所谓的数据库中就解决了？如果真解决了，那快教我怎么用吧。

老田：当然解决了，因为我接下来要给你说的是关系型数据库，所以这些让人凌乱的关系问题都可以迎刃而解。比如上面这个难题，我们用数据库创建如下三张数据表，一次就解决了，以后再也不用担心那些麻烦事。

(1) 分公司表

分公司 ID	公 司 名	分公司领导	地 址
G1	北京总公司	罗天伟	北京 XX 路 XX 号
G2	安徽分公司	黄世元	合肥 XX 路 XX 号
G3	重庆分公司	邹坤	重庆 XX 路 XX 号

(2) 部门表

部门 ID	部门名称	部门领导	所属分公司的 ID
B1	人力资源部	高大伟	G1
B2	市场部	疯扯扯	G1
B3	综合管理部	李洁	G1
B4	项目实施部	天轰穿	G2

(3) 员工表

员工工号	姓 名	电 话	所属部门的 ID
U1	田皓文	18080801234	B2

U2	黄旺男	18080804321	B4
----	-----	-------------	----

老田：发现上面三张表之间的关系没有？有什么好处？

小天：发现了，公司表中全部是各分公司的信息。部门表中全部是部门的信息，同时有一个项是说明这个部门属于哪一个分公司。员工表中全部是员工个人信息，同时也有一个项目来说明其属于具体哪一个部门。这样可以顺着关系一层层地理清某个员工具体属于某个公司的某个部门。同时，要调动一个员工，只需要更改员工所属部门就可以了。要改变某个主管的信息也非常简单，直接改变就是了，因为一个主管实际上在数据库中只存在一条记录。但是这个要做成数据库应该很困难吧，一张表就可以搞定的，要写这么多内容。

老田：其实非常简单，当你学完本书第4章之后，你就会发现，原来一切如此简单。我现在就不用给你数据库的全部代码了，因为我就是想调下你的胃口，哈哈。我们说要学好一门技术，就必须一步步来，而不是一步登天。毕竟我们学的是科学，不是修真，即使修真也是一步步来的，否则让你直接去渡劫，一次玩完。

1.3 数据库的基本概念

要想了解数据库的前因后果，就得从数据库的概念到历史再到现状分别进行解释。这段你可以学得快一些，没必要又要记笔记又要背，老田我最恨背书的人。所以我会让你快乐地学到懂，而不是背到你死。

DBMS：数据库管理系统（Database Management System）是一种操纵和管理数据库的大型软件，用于建立、使用和维护数据库，简称DBMS。它对数据库进行统一的管理和控制，以保证数据库的安全性和完整性。用户通过DBMS访问数据库中的数据，数据库管理员也通过DBMS进行数据库的维护工作。它可使多个应用程序和用户用不同的方法在同时或不同时刻去建立、修改和访问数据库。DBMS提供数据定义语言DDL（Data Definition Language）与数据操作语言DML（Data Manipulation Language）供用户定义数据库的模式结构与权限约束，实现对数据的追加、删除等操作。

RDBMS：关系型数据库管理系统（Relational Database Management System）的概念更简单，就是在数据库管理系统的基础上增加关系。它通过数据、关系和对数据的约束三者组成的数据模型来存放和管理数据。

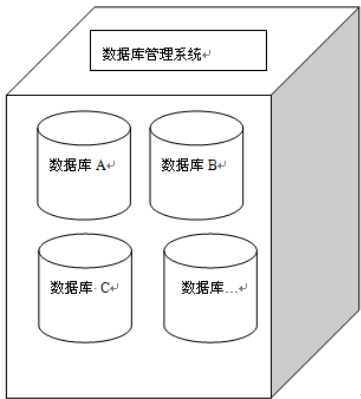


图 1-2 数据库管理系统与数据库

数据库：在一个数据库管理系统中，可以有多个数据库，如图1-2所示。

1.4 数据库的历史

数据库大体可分为网状数据库、层次数据库和关系数据库三类。

数据库从出生到现在只半个世纪左右，已经形成了坚实的理论基础、成熟的商业产品和广泛的应用领域，吸引越来越多的研究者加入。数据库的诞生和发展给计算机信息管理带来了一场巨大的革命。几十年来，国内外已经开发建设了成千上万个数据库，它已成为企业、部门乃至个人日常工作、生产和生活的基础设施。同时，随着应用的扩展与深入，数据库的数量和规模越来越大，数据库的研究领域也已经大大地拓广和深化了。

在数据库诞生之前：那时的数据管理非常简单。通过大量的分类、比较和表格绘制的机器运行数百万穿孔卡片来进行数据的处理，其运行结果在纸上打印出来或者制成新的穿孔卡片。而数据管理就是对所有这些穿孔卡片进行物理储存和处理。

1951年，雷明顿兰德公司（Remington Rand Inc）的一种叫做UNIVAC 1的计算机推出了一种一秒钟可以输入数百条记录的磁带驱动器，从而引发了数据管理的革命。

1956年，IBM生产出第一个磁盘驱动器——the Model 305 RAMAC。此驱动器有50个盘片，每个盘片直径是2英尺，可以储存5MB的数据。

1961年，通用电气公司（General Electric Company）的查尔斯·巴赫曼（Charles William Bachman）成功地开发出世界上第一个网状DBMS，也是第一个数据库管理系统——集成数据存储（Integrated DataStore, IDS），奠定了网状数据库的基础。

1968年开发的IMS（Information Management System），是一种适合其主机的层次数据库。这是IBM公司研制的最早的大型数据库系统程序产品。

1969年，埃德加·科德发明了关系数据库。1970年关系模型建立之后，IBM公司在San Jose实验室增加了更多的研究人员研究这个项目，这个项目就是著名的System R。其目标是论证一个全功能关系DBMS的可行性。该项目结束于1979年，完成了第一个实现SQL的DBMS。然而IBM对IMS的承诺阻止了System R的投产，一直到1980年System R才作为一个产品正式推向市场。

1970年，IBM的研究员埃德加·科德博士在刊物《Communication of the ACM》上发表了一篇名为“A Relational Model of Data for Large Shared Data Banks”的论文，提出了关系模型的概念，奠定了关系模型的理论基础。

1973年，加州大学伯克利分校的Michael Stonebraker和Eugene Wong利用System R

已发布的信息开始开发自己的关系数据库系统Ingres。他们开发的Ingres项目最后由Oracle公司、Ingres公司以及硅谷的其他厂商所商品化。

1976年，霍尼韦尔公司（Honeywell）开发了第一个商用关系数据库系统——Multics Relational Data Store。关系型数据库系统以关系代数为坚实的理论基础。其代表产品有Oracle、IBM公司的DB2、微软公司的MS SQL Server以及Informix、ADABASD等。

1979年，Oracle公司引入了第一个商用SQL关系数据库管理系统。

1983年，IBM 推出了DB2数据库产品。

1985年，为Procter & Gamble系统设计的第一个商务智能系统产生。

1987年，赛贝斯公司发布了Sybase SQL Server系统，用于UNIX环境。

1991年，W.H.Bill Inmon发表了“构建数据仓库”。

小天：哎呀，我要用要学的都是最新的，你给我说这么多干什么啊？

老田：好吧，时光如梭，转眼到了1998年，微软发布了Microsoft SQL Server 7.0版本，该版本在数据存储、查询、可伸缩性方面有了巨大的改进，也使它有了和IBM的DB2、甲骨文的Oracle、赛贝斯的Sybase ASE系统有了竞争的本钱。

2012年3月，微软发布Microsoft SQL Server 2012 RC0，同期市面上应用非常广泛的还有Oracle，也在Oracle 11G之后发布了Oracle 12C，MySQL这个开源数据库也发展到了MySQL 6了，IBM的DB2也发展到V10版本。

小天：打住，打住，你这时间飘得太快，一下飘了21年。我是否可以这样理解，从20世纪80年代开始，数据库技术就进入了关系数据库时代。而数据库经历了网状数据库、层次数据库和关系数据库三个时代，我们现在接触到的基本上都是关系型数据库，而Microsoft SQL Server、MySQL、Oracle、DB2等也只是关系数据库中的一种。

老田：随着互联网Web 2.0网站的兴起，传统的关系数据库在应付Web 2.0网站，特别是在超大规模和高并发的SNS类型的Web 2.0纯动态网站中已经显得力不从心，暴露出很多难以克服的问题，而非关系型的数据库则由于其本身的特点得到了非常迅速的发展。但是非关系型数据库并未形成一定标准，各种产品层出不穷，内部混乱，各种项目还需时间来检验。

小天：哇哦，非关系型数据库不成熟，咱不学，但是还有这么多数据库，就算它们大同小异，我们要学习总得具体落实到一个点上吧，我该怎么学啊，不会一次把所有数据库都学会吧？

老田：当然不会，事实上只要你掌握好一种数据库之后花很短的时间就可以掌握其他几种数据库。考虑到你对数据库几乎没有什么概念，我选择上手最容易，帮助文档最智能的Microsoft SQL Server作为学习的载体。当然，既然关系数据库发展到了顶峰，我们就有必要进一步解释一下：到底啥才是关系数据库。

1.5 关系数据库

关系数据库是建立在关系模型基础上的数据库，借助于集合代数等数学概念和方法来处理数据库中的数据。现实世界中的各种实体以及实体之间的各种联系均用关系模型来表示。关系模型是由埃德加·科德于1970年首先提出的，并配合“科德十二定律”。如今虽然对此模型有一些批评意见，但它仍是数据存储的传统标准。标准数据查询语言SQL就是一种基于关系数据库的语言，这种语言执行对关系数据库中数据的检索和操作。关系模型由关系数据结构、关系操作集合和关系完整性约束三部分组成，如图1-3所示。

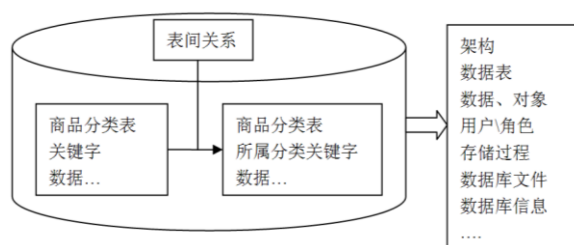


图 1-3 关系型数据库

小天：看着图1-3的意思好像是说，在数据库中有很多表，而数据则是按照一个个分类存储放在不同的表中？

老田：是的，所谓的关系，说的就是不同表之间的关系。那么这些表又是怎么被设计出来的呢？这就要说到另外一个在设计程序中非常有名的东东——实体关系模型（Entity-Relationship Model, E-R Model），它是陈品山（Peter P.S Chen）博士于1976年提出的一套数据库的设计工具，他运用真实世界中事物与关系的观念，来解释数据库中抽象的数据架构。实体关系模型利用图形的方式（实体—关系图（Entity-Relationship Diagram））来表示数据库的概念设计，有助于设计过程中的构思及沟通讨论。

关系模型就是指二维表格模型，因而一个关系数据库就是由二维表及其之间的联系组成的一个数据组织。当前主流的关系数据库有Oracle、DB2、Microsoft SQL Server、Microsoft Access、MySQL等。

1.6 为什么选择 SQL Server

小天：既然有这么多的数据库可以选择，为什么我们一定要学习Microsoft SQL Server呢？总不能说它最容易上手就学它，要是学会以后出去找不到工作找谁哭啊？

老田：你这个问题太尖锐了，多余的还是不要去说的好，就单论咱们的个体情况来说，我认为这样：

首先，微软的产品一贯传统良好的用户体验，无论界面操作还是系统帮助，都非常人性化。就这两点，对初学的人来说绝对是首选，因为相对入门来说，几种数据库查询语言和控制语言差异并不大，那么我们就选最容易学的入门。否则只一个安装过程就把我们给弄昏了，还咋学呢？

Oracle和Microsoft SQL Server还有一个对比，前者非常灵活，你需要哪个功能就购买哪个功能，不需要则不用多花钱，但是Microsoft SQL Server却是一次购买所有的，无论你是否需要。这看似是一个缺点，但作为初学者来说就不是缺点了，因为我们现在根本不知道该买什么，不该买什么。

至于其他的什么性能、售后、趋势等咱们不去得罪人，呵呵。

小天：照你这么说，我还是学Oracle吧，虽然学的时候复杂点，但是一次就学成了。

老田：事实上，如果学习数据库只是用于软件开发而不是作为数据库管理员使用的话，那么学什么数据库都差不多，以后并不需要专门去学什么，因为几种流行数据库的数据检索语句都遵循ANSI SQL标准，不同的是它们各自的维护语法。另外，从SQL Server 2000开始，Microsoft SQL Server也逐渐被大型的项目所接受了，而发展到当前的2012版本，很多方面已经丝毫不逊色，甚至在云计算方面已经超越Oracle了。

小天：也就是说学你这本书就只能用SQL Server？

老田：当然不是了，这本书重点讨论的是利用关系数据库作为数据存储来开发各种类型的计算机应用程序。换句话说，只要是遵循了SQL标准的关系数据库都可以看这本书，比如Oracle、MySQL、PostgreSQL等。

1.7 SQL 语言

小天：你说的ANSI SQL标准是个神马东西？

老田：要说ANSI SQL标准就得先说神马是SQL，SQL（Structured Query Language）结构化查询语言是一种数据库查询和程序设计语言，用于存取数据以及查询、更新和管理关系数据库系统。同时也是数据库脚本文件的扩展名。读音可以是S-Q-L，逐个发音，不过大多数牛人更习惯读作“sequel”。SQL是高级的非过程化编程语言，是沟通数据库服务器和客户端的重要工具，允许用户在高层数据结构上工作。它不要求用户指定对数据的存放方法，也不需要用户了解具体的数据存放方式，所以具有完全不同底层结构的不同数据库系统，可以使用相同的SQL语言作为数据输入与管理的SQL接口。它以记录集作为操作对象，所有SQL语句接受集合作为输入，返回集合作为输出。这种集合特性允许一条SQL语句的输出作为另一条SQL语句的输入，所以SQL语句可以嵌套，这使它具有极大的灵活

性和强大的功能。多数情况下，在其他语言中需要一大段程序实现的功能只需要一个SQL语句就可以达到目的，这也意味着用SQL语言可以写出非常复杂的语句。

SQL语言包含以下3个部分。

- 数据定义语言（Data Definition Language，DDL），定义（definition）。例如：CREATE、DROP、ALTER 等语句。
- 数据操作语言（Data Manipulation Language，DML），操作（make）。例如：INSERT（插入）、UPDATE（修改）、DELETE（删除）等语句。
- 数据控制语言（Data Controlling Language，DCL），控制（control）。例如：GRANT、REVOKE、COMMIT、ROLLBACK 等语句。

小天：我在其他的SQL Server的书常常看到“T-SQL语言”这个词，是什么？

老田：T-SQL，全称Transact-SQL。它也遵循SQL标准，但是在这个基础上，做了少量的扩展。换句话说，T-SQL语言是遵循SQL标准的，专门为SQL Server做了少量扩展的扩展SQL语言。

小天：我又要担心了，我要学习的是数据库，不是单纯的SQL Server。

老田：在本书范围内所讲的99%的T-SQL语法，都是完全遵循SQL标准的。否则我干脆叫它为大话SQL Server，何必叫它大话数据库呢？

1.8 SQL 标准

小天：也是哦，那你上面说“具有完全不同底层结构的不同数据库系统，可以使用相同的SQL语言作为数据输入与管理的SQL接口”，可这个SQL语言总得有个标准规范，所有数据库厂商都去遵守吧？

老田：说你看书不认真，你还真是。上面不就说到ANSI SQL标准吗？这个标准是ISO和IEC发布的SQL国际标准，在1992年制定了一个标志性的版本，称为SQL-92。1999年，再次修订。目前最新的版本是SQL-2008，而这期间还有SQL-2003、SQL-2006。从SQL-99到SQL-2008，可以看到标准修订的周期越来越短，多少也反映了对技术的需求变化之快。当然，这些事并不是我们这种浅层次应用的菜鸟需要担忧的，因为国际标准出来，MS Access、DB2、Informix、MySQL、MS SQL Server、Oracle、Sybase等都必须去遵循。另外，从SQL-99的改进，主要是针对XML、Window函数、Merge语句等，特别是对XML的增强。对于基本的数据定义和操作语言并有更多的修正，所以目前大部分的数据书籍更多还是重点在ANSI SQL的标准范畴内。

注：美国国家标准局（American National Standards Institute，ANSI）与国际标准化组

织（International Organization for Standardization, ISO）已经制定了SQL标准。ANSI是一个美国工业和商业集团组织，负责开发美国的商务和通信标准。ANSI同时也是ISO和IEC（International Electrotechnical Commission）的成员之一。

小天：SQL标准是不是就像中国的普通话一样，是一个大家都遵从的执行标准？

老田：是的，简而言之，1992年标准就是一个所有关系数据库都遵从的数据操作语言标准。1974年，IBM的Ray Boyce和Don Chamberlin将Codd关系数据库的12条准则的数学定义以简单的关键字语法表现出来，里程碑式地提出了SQL语言。SQL语言的功能包括查询、操纵、定义和控制，是一个综合的、通用的关系数据库语言，同时又是一种高度非过程化的语言，只要求用户指出做什么而不需要指出怎么做。SQL集成实现了数据库生命周期中的全部操作。

小天：这么说来，只要我学习一种遵从ANSI SQL标准的SQL语言就可以在所有流行的关系型数据库中都适用？

老田：是这么个理，但并不完全，不同的数据库语言之间确实有一些细节的差异。除了SQL标准之外，大部分SQL数据库程序都拥有它们自己的私有扩展。当然，纯粹的数据查询、增加、删除、修改是一致的，主要在控制语言、附加语言元素等方面有一些差异，这也是为什么前面我多次提到，作为一个程序员来说，学什么数据库对后面影响都不大，但要做DBA（数据库管理员）就必须选择你对口的数据了。

1.9 10 分钟探索 IDE

小天：明白了，话虽这样说，要学SQL编程，总得有个IDE（继承编程环境）吧？比如.NET编程有Visual Studio。

老田：当然有了，不过本书的讲解从表面来看基本上是倚重SQL Server Management Studio，但这并不妨碍你使用其他数据库来学习，只是说如果你完全零基础的话，我推荐你跟着我的脚步来走。

首先建议你安装SQL Server 2008。如果不会安装的话，去搜索一下安装SQL Server 2008的图文教程，网上很多。安装完成后记得将下载包：\示例数据库\SQL2008.Adventure Works_All_Databases.x86这个文件安装了。在本书第2章会多次用到这个实例数据库。

接下来要简单地介绍一下后面整个学习过程都要接触到的工具——SQL

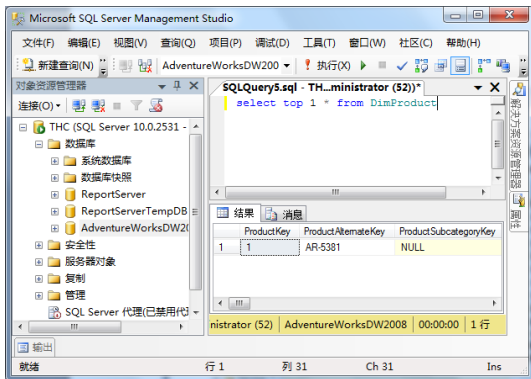


图 1-4 SQL Server Management Studio 主窗口

Server Management Studio。它是从SQL Server 2005版本开始出现的一种新的集成环境，将各种图形化工具和多功能的脚本编辑器组合在一起，完成访问、配置、控制、管理和开发SQL Server的所有工作。后面我们学习中大部分的实践都在这里。它启动后主窗口如图1-4所示。

小天：等等，现在微软的SQL Server 2012都出来了，图1-4我怎么看都像是SQL Server 2005或者SQL Server 2008，为啥不用最新的啊？还有，你不要老讲理论了，咱们时间紧迫，哪怕是讲这个，你也可以找点实例来让我做，这样既学会了工具又能够多少学点东西。

老田：好吧，我选择SQL Server 2008作为演示的最大原因是它有“动态帮助”，这个小功能对初学者的意义非常大，后面我会讲到如何用。对于实际动手方面，我列一个清单，我们一个个地来做一次，做完后，基本上SQL Server Management Studio你就应该会用了。

- 启动 SQL Server Management Studio。
- 与已注册的服务器和对象资源管理器连接。
- 更改环境布局。
- 显示文档窗口。
- 显示对象资源管理器详细信息页。
- 选择键盘快捷方式方案。
- 设置启动选项。
- 还原默认的 SQL Server Management Studio 配置。

小天：老田，你不要糊弄我啦，你这个压根就是我们安装SQL Server的附带《SQL Server 教程》第一页就有的东西，你直接粘贴过来就想把我打发了，过分哦。

小提示：打开SQL Server 教程的方法，在“开始”菜单上，指向“所有程序”，再指向 Microsoft SQL Server 20xx（2005、2008、2012），然后再打开文档和教程，最后单击 SQL Server 教程。

老田：怕你啦，你说你希望做些什么嘛？

小天：我希望实践的顺序如下。

（1）打开SQL Server Management Studio 我知道了，但是我还不很清楚怎么进到图1-4所示的那个主窗口去。

（2）修改登录验证模式。

（3）SQL Server Management Studio和数据库服务器之间如何联系的？可以同时管理多个服务器吗？

（4）可以在这里修改数据库服务器的一些属性吗？

- (5) 可以在这里直接启动和停止数据库服务器吗?
- (6) 如何创建T-SQL脚本?
- (7) 如何创建表、视图、存储过程等?
- (8) 如何创建新的登录账户?
- (9) 如何管理报表?

老田: Stop, 越说越过分了, 你一口吃得下这么多吗? 就算你一口吃得下, 我一天教得完吗。前5个问题下面我分别回答, 其余4个问题以及你还没有问出来的, 我们会在本书后面的章节中陆续进行讲解。

小天: 那你就不能按照项目来讲吗? 我就是想在学习的过程中, 一次性学会如何做项目。

老田: 哥, 我说你小子怎么就这么浮躁呢? 听过“爬都没有学会就想跑”这句话没有? 你现在需要的是将最基本的使用搞懂, 下一章开始我们就涉及到如何对数据库进行创建、删除、修改、备份、压缩等, 以便你在完全熟练工具的同时学习到程序员必须学会的数据库维护技术。之后依次从SQL语言开始讲解一直到如何设计一个最简单的数据库到大型的数据库。一步步来嘛, 一口是吃不胖的, 但是可以一口噎住自己。

小天: 好吧, 我们开始熟悉工具吧, 但是你最好简单点, 什么安装配置这些我可以在网上搜索, 不需要你用大篇幅地给我截图、讲解。

1.9.1 启动和登录 SQL Server Management Studio

老田: 知道, 我也不想干那种事, 开始我们的第一个实例——启动和登录SQL Server Management Studio, 步骤很简单, 如下所示。

- (1) 单击Windows开始菜单。
- (2) 选择“程序”或者“所有程序”命令。
- (3) 选择Microsoft SQLServer 2008目录。
- (4) 选择SQL Server Management

Studio, 启动后首先是连接到数据库服务器, 如图1-5界面。

服务器类型: 就是你要登录的那一项服务, 比如数据库引擎、分析服务、报表服务等。

服务器名称: 你要登录到的服务器



图 1-5 登录 SQL Server Management Studio

名字或者IP地址+实例名（默认实例名则可以不填写），例如THC\SQLEXPRESS或者192.168.11.11。如果是本地服务器同时又是默认实例名，则直接用“.”代替。

身份验证：在安装SQL Server的时候，我们选择身份验证模式了，我提醒过最好是选择混合模式，如果在那里选择了“Windows身份验证模式”，这里就只能选择“Windows身份验证”，否则的话在这里选择“SQL Server身份验证”，用户名填写“sa”，密码就是安装时候你设置的那个。

最后单击“连接”按钮，如果你的机器快的话就可以进入了，否则就等下了。

小天：错误了，你看图1-6所示。

老田：发生这样的错误，一般来说只有两种情况：第一种情况是你的“服务器名称”项填写错误了，第二种情况就是检查你的服务器是否启动了，检查的方法如下。

首先，打开SQL Server 配置管理器，方法如下。

- (1) 单击Windows开始菜单。
- (2) 选择“程序”或者“所有程序”命令。
- (3) 选择 Microsoft SQLServer 2008命令。

- (4) 选择配置工具命令。

- (5) 选择SQL Server配置管理器，打开后窗口如图1-7所示。

注意看你的服务器是否如图1-7所示，要连接的服务器是否处于“正在运行”状态。启动模式就好解释了，自动就是随系统启动，手动就是啥时候需要了自己来启动。

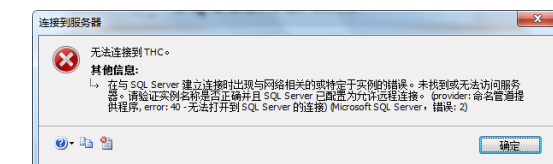


图 1-6 登录时发生“无法连接”的错误

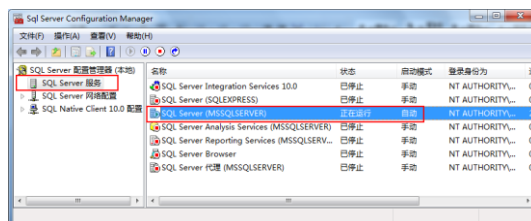


图 1-7 服务器状态

1.9.2 修改登录验证模式

小天：懂了，我就是在安装的时候为了节省计算机资源，所以设置了全部为手动启动，于是这里服务就没有启动了。

奇怪，为什么我只能用Windows身份验证模式登录呢？而用SQL Server身份验证模式登录的话，则弹出如图1-8所示的错误提示信息。

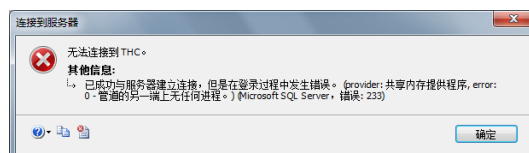


图 1-8 登录错误

老田：如果遇到这个问题，多是因为SQL Server服务器没有启用TCP/IP协议，按照下面的步骤检查一下设置。

- (1) 单击Windows开始菜单。
- (2) 选择“程序”或者“所有程序”命令。

(3) 选择Microsoft SQLServer 2008 命令。

- (4) 选择配置工具命令。
- (5) 单击SQL Server配置管理器。

(6) 展开配置管理器左边的“SQL Server网络配置”节点。

(7) 单击“MS SQL SERVER的协议”选项。

(8) 得到图1-9，设置启用它的Named Pipes(命名管道)和TCP/IP协议。

双击TCP/IP选项，弹出如图1-10所示的对话框。在“协议”选项卡中将“已启用”选项设置为“是”，然后单击“确定”按钮。此时系统会要求你重新启动服务器，先不管它，继续启用Named Pipes协议。最后的结果是除VIA协议外，其他协议全部启用。

小天：在这里重新启动服务器怎么做，不是要我重新打开SQL Server Management Studio用Windows身份验证连接到服务器后再重启服务器吧，太麻烦了。

老田：你想复杂了，重新启动服务器其实很简单的。在如图1-11所示的对话框中，选择左边窗格中的“SQL Service服务”选项，此时右边窗格中会显示该计算机上全部的SQL Server服务项目。选择你要重启的选项，单击鼠标右键，在弹出的快捷菜单中选择“重新启动”命令，这样就可以关闭该窗口，就可以在SQL Server Management Studio中使用SQL Server身份验证模式登录了。

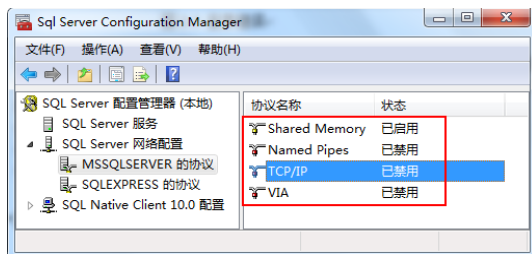


图 1-9 检查连接协议

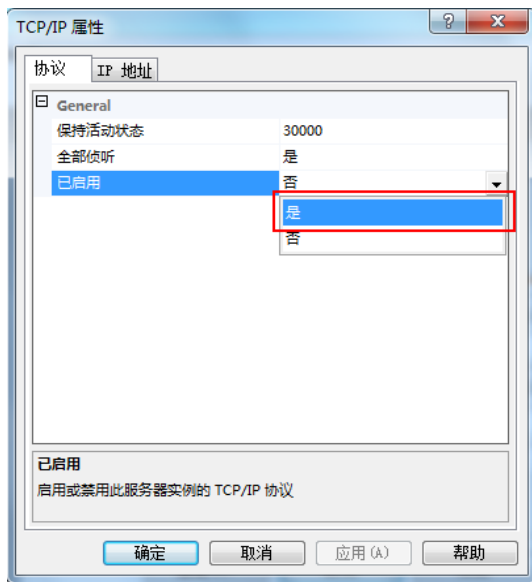


图 1-10 启用协议

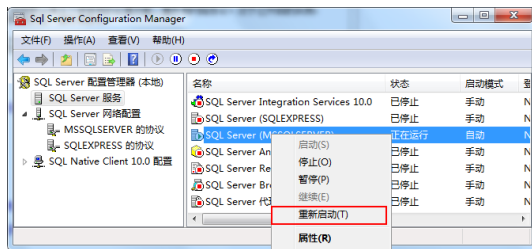


图 1-11 重新启动服务器

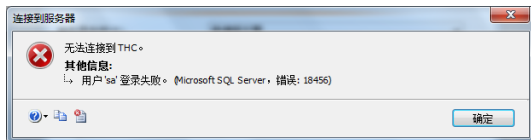


图 1-12 登录错误提示

小天：老田你就忽悠我吧，怎么还是不行！你看，又换了个错误提示信息，如图1-12所示。

老田：如果继续出现错误提示，那就是你的服务器中还未设置“混合验证模式”，或者是“sa”账户还处于禁用状态。按照下面的步骤检查设置，解决你所说的问题。

(1) 设置“混合验证模式”

使用Windows身份验证登录SQL Server服务器。在“对象资源管理器”中选择要设置的数据库，单击鼠标右键，在弹出的快捷菜单中选择“属性”命令，如图1-13所示。

在打开的“服务器属性”对话框中选择左侧列表框中的“安全性”选项，并在“服务器身份验证模式”选项组中，选中“SQL Server和Windows身份验证模式”单选按钮，如图1-14所示。

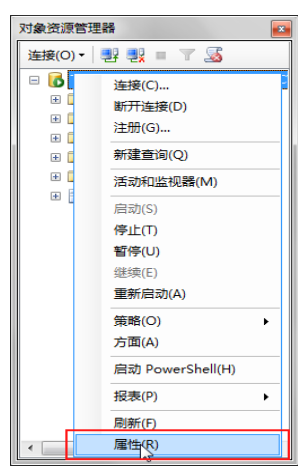


图 1-13 选择“属性”命令

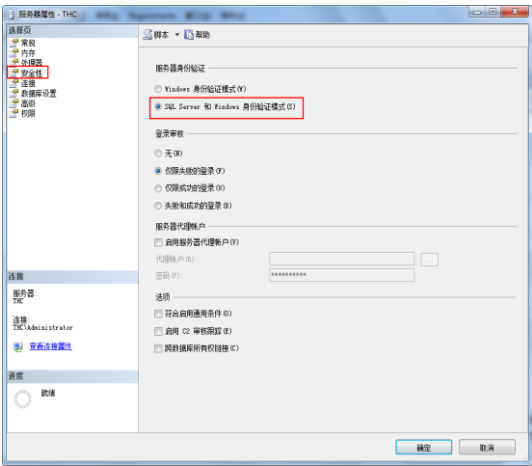


图 1-14 修改服务器身份验证模式

(2) 启用“sa”账户

启用方法如下。

① 打开“对象资源管理器”，找到“指定的Sql Server实例”，打开该实例下面的“安全性”选项，再打开“登录名”选项，找到“sa”账户，然后双击“sa”账户，或者在“sa”账户上单击鼠标右键，在弹出的快捷菜单中选择“属性”命令，打开登录属性窗口对话框，如图1-15所示。

② 如果忘记了安装时为“sa”账户设置的密码，那么就在图1-15中标注

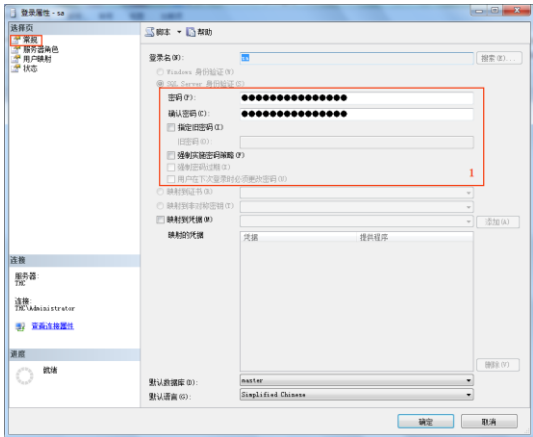


图 1-15 登录属性对话框

“1”的位置重新输入密码，需要重复输入两次以确认。

③ 接着取消选中下面的“强制实施密码策略”复选框。关于密码策略后面会详细讲到。

④ 在“登录属性”对话框中选择左边的“状态”选项，为该账户设置“授予”连接数据库引擎的权限，并在“登录”选项组中选择“启用”选项，如图 1-16所示。

设置完成后单击“确定”按钮，关闭该对话框。最后是重新启动服务器，就按照前面所说的步骤执行。好了，现在你可以再次登录了。另外插一句话，关于“sa”账户，可以修改名称，但最好不要去做，改了出问题我可不负责哟！

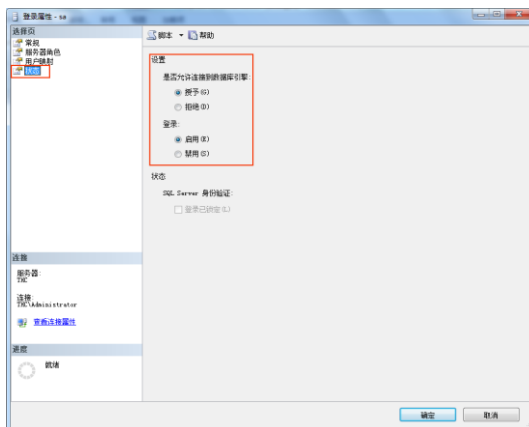


图 1-16 修改 sa 属性

1.9.3 注册数据库服务器

小天：好了，两种模式都可以登录了，下面的问题是如何再添加一个服务器呀？

老田：看图 1-17，注意红线标注了的那个位置。

直接选择自己看，和第一个问题差不多，我就不多说了。

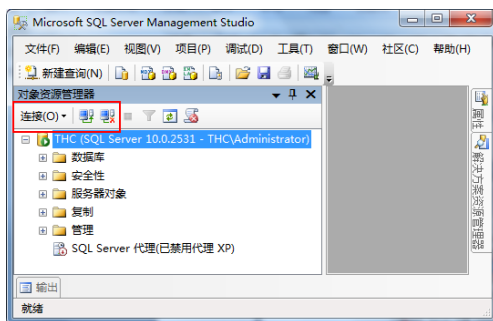


图 1-17 注册新的服务器

1.9.4 修改数据库服务器属性

既然说到修改数据库服务器的属性，我们就先熟悉下数据库服务器的常规操作菜单，在对象资源管理器中选择你要修改的服务器，如图 1-18所示。

要修改属性，当然是在右键菜单中选择“属性”命令。打开数据库服务属性对话框后，左边是属性分类，右边则是每个

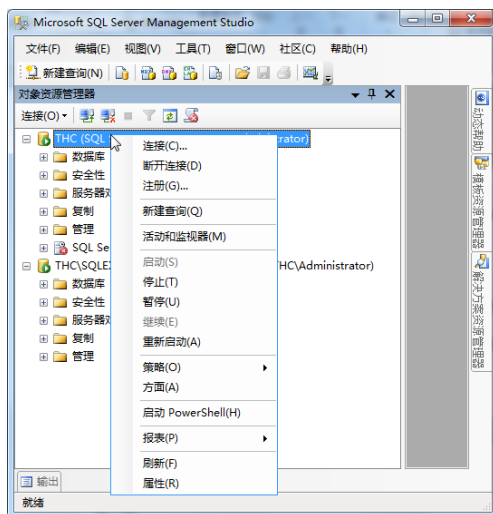


图 1-18 数据库服务器的常规操作菜单

分类的详细设置，修改完后记得单击“确定”按钮。

1.9.5 启动和停止服务器

小天：修改完，单击“确定”按钮后出现提示，如图1-19所示，要求重新启动SQL Server，那如何重启呢？

老田：其实坦白地说，这个问题你不该问啊，小天，学习中要多看，多尝试，不要什么都依赖外物。在你修改属性之前，单击鼠标右键弹出的快捷菜单中难道没有看见有启动、停止、暂停、重新启动命令吗。

1.9.6 创建查询

如图1-20所示中红线标注的位置，单击左上角的“新建查询”图标，而具体的SQL语句则是写在右边红线标注区域中。最终查询结果和消息显示在右下区域。

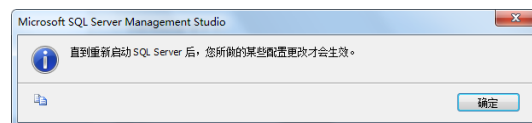


图 1-19 提示信息

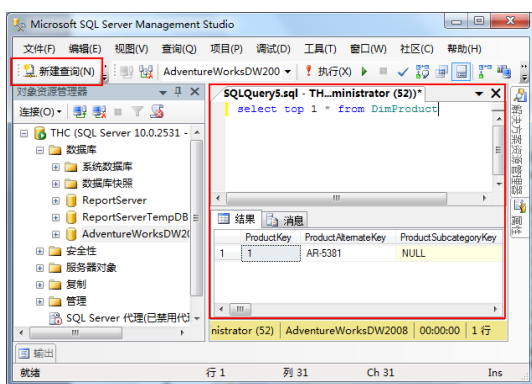


图 1-20 新建查询

1.9.7 使用指定数据库

小天：老田，在一个服务器（数据库管理系统）中存在多个数据库，按照图1-21这样执行一个操作，SQL Server Management Studio怎么就知道你是要操作的那张表呢？你的SQL语句中只指定了表名，又没有数据库的名字。

老田：这个问题问得好，有两种方式可以指定，一种是直接在SQL Server Management Studio中选择，如图1-21所示。

另外一种方式就是，执行一条SQL语句，格式为use+空格+要使用的数据库名称，例如要使用AdventureWorksDW2008这个数据库，就执行：

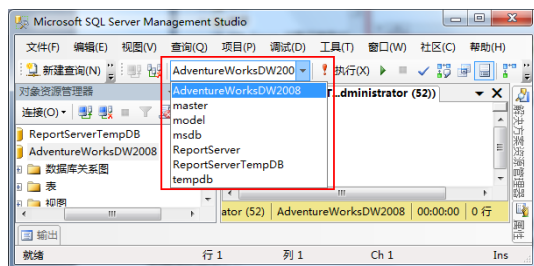


图 1-21 选择要使用的数据库

本章小结

本章我们首先解释了什么是数据库，然后分析为什么要使用数据库，接着罗列出数据库的发展历史和SQL Server 2008的框架结构，通过这四点让学员对此有一个感性的认识。

接下来我们进入实际操作阶段。

首先是介绍数据库的安装规划。

接着是安装步骤。

然后对安装好的数据库各项进行简要的配置。

第三个阶段就是带领学员一起了解常用的管理工具，让学员能够对数据库做浅显的接触。

最后我们对数据库和数据库对象的特点进行分析。

问 题

1. 数据库是什么？
2. DBMS的中文和英文全称分别是什么？
3. 目前数据库市场上主流的数据库有哪些，同时说出对应的厂商名称？
4. 只要遵循SQL标准的数据库系统，其SQL语法是否完全一样？
5. SQL标准的解释。
6. 执行基本的SQL脚本是在什么工具中？
7. 数据库服务实例名是什么？
8. 什么是Windows登录验证？什么是混合验证？
9. 如何修改sa的密码？