

浅谈关系型数据库

周其乐

(廊坊职业技术学院 河北廊坊 065001)

摘 要: 本文概略地叙述了关系型数据库的发展历史, 简要地介绍了关系型数据库所具有的一般特点, 同时也指出了它的一些不足。

关键词: 关系型数据库 二维表格数据模型 检索

中图分类号: TP392

文献标识码: A

文章编号: 1672-3791(2009)02(a)-0015-01

网状数据库和层次数据库已经很好地解决了数据的集中和共享问题, 但是在数据独立性和抽象级别上仍有很大欠缺。用户在对这两种数据库进行存取时, 仍然需要明确数据的存储结构, 指出存取路径。关系数据库则较好地解决了这些问题。

1970年, E.F.Codd博士在刊物上发表了一篇文章, 提出了关系模型的概念, 奠定了关系模型的理论基础。

关系型数据库管理系统中储存与管理数据的基本形式是二维表。关系型数据结构把一些复杂的数据结构归结为简单的二元关系(即二维表格形式)。例如某单位的职工关系就是一个二元关系。关系型数据库允许数据被存储在多个平面文件中, 这些表通过被称为“键”的共享数据域而相互关联。关系型数据库提供了对特殊报表的更简单的访问机制一般通过, 并且通过减少冗余而改善了可靠性。

关系型数据库以行和列的形式存储数据, 以便于用户理解。这一系列的行和列被称为表, 一组表组成了数据库。由关系数据结构组成的数据库系统被称为关系数据库系统。

下面列出了关系模型中的常用概念。

关系: 可以理解为一组二维表, 每个关系都具有一个关系名, 就是通常说的表名。

元组: 可以理解为一组二维表中的一行, 在数据库中经常被称为记录。

属性: 可以理解为一组二维表中的一列, 在数据库中经常被称为字段。

域: 属性的取值范围, 也就是数据库中某一列的取值限制。

关键字: 一组可以唯一标识元组的属性。数据库中常称为主键, 由一个或多个列组成。

关系模式: 指对关系的描述, 其格式为: 关系名(属性1, 属性2, ..., 属性N)。在数据库中通常称为表结构。

常用的关系操作可分为两大类查询和数据更新。查询: 选择、投影、连接、除、并、交、差; 数据更新: 插入、删除、修改。查询的表达能力是其中最主要的部分。

用户用查询(Query)来检索数据库中的数据。一个Query是一个用于指定数据库中行和列的SELECT语句。关系型数据库通常包含下列组件:

客户端应用程序(Client)

数据库服务器(Server)

数据库(Database)

Structured Query Language(SQL) Client端和Server端的桥梁, Client用SQL来象Server端发送请求, Server返回Client端

要求的结果。

在关系数据库中, 对数据的操作几乎全部建立在一个或多个关系表格上, 通过对这些关系表格的分类、合并、连接或选取等运算来实现数据的管理。对于一个实际的应用问题(如人事管理问题), 有时需要多个关系才能实现。用数据库管理系统建立起来的一个关系称为一个数据库(或称数据库文件), 而把对应多个关系建立起来的多个数据库称为数据库系统。数据库管理系统的另一个重要功能是通过建立命令文件来实现对数据库的使用和管理, 对于一个数据库系统相应的命令序列文件, 称为该数据库的应用系统。因此, 可以概括地说, 一个关系称为一个数据库, 若干个数据库可以构成一个数据库系统。数据库系统可以派生出各种不同类型的辅助文件和建立它的应用系统。

关系型数据库按下列方式工作, 例如: 电话公司会围绕主键例如电话号码建立它的主表, 并把它与其它的基本客户信息储存起来。然后, 公司可以用那个主键和表示增值服务例如来电显示和呼叫等待的列来定义一个单独的表。它还可以定义另一个表来跟踪计费, 主键还是电话号码, 用另一列来存储计费数据。

虽然信息保存在不同的表中, 但最终用户可以很容易地按他们要提取子查询想要的信息。所以, 电话公司的客户服务代表可以提取用户的计费信息、特殊服务状态或最近收到的付费, 将它们显示在同一个屏幕上。

所以, 在层次上, 关系型数据库与实体间联系是固定的, 有良好的完整性支持, 对具有一对多的层次关系的部门描述自然、直观、容易理解。但是对查入和删除操作限制比较多, 查询子女结点必须通过双亲节点, 命令趋于程序化。

在结构上, 关系型数据库更为直接的描述现实世界, 有良好的性能, 存取效率较高。但是结构比较复杂, 而且随着应用环境的扩大, 数据库的结构越来越复杂, 不利于用户掌握, 其DDL, DML语言复杂, 用户不容易使用。

在关系上, 关系型数据库建立在严格的数学概念的基础上, 概念单一, 实体与实体间的联系都用关系表示, 故其数据结构简单、清晰, 存取路径对用户透明, 故有更高的数据独立性和更好的安全保密性。但是查询效率不如非关系型数据库, 故必须对查询进行优化, 增加了开发数据库管理系统的难度。

关系型数据库相比其他模型的数据库

而言, 有着以下优点。

(1) 容易理解: 二维表结构是非常贴近逻辑世界的一个概念, 建立在严格的数学概念的基础上, 概念单一, 数据结构简单、清晰。关系模型相对网状、层次等其他模型来说更容易理解。

(2) 使用方便: 通用的SQL语言易学易懂, 操作关系型数据库时也非常方便, 程序员甚至于数据管理员可以方便地在逻辑层面操作数据库, 而完全不必理解其底层实现。

(3) 易于维护: 丰富的完整性(实体完整性、参照完整性和用户定义的完整性)大大降低了数据冗余和数据不一致的概率。

(4) 更安全: 权限的分配, 使其较以往的数据库在安全性上要高的多。登录身份验证可以与Windows身份验证集成在一起, 安全可靠。

(5) 便于访问数据: 提供了诸如视图, 存储过程, 触发器, 索引等对象。

但是, 关系型数据库能很好地处理所谓的“表格型数据”, 却对技术界出现的越来越多的复杂类型的数据无能为力, 例如关系数据库所采用的二维表格数据模型不能有效地处理在大多数事务处理应用中典型的多维数据。另外, 关系型数据库在对GUI和Web事务处理的真实世界中其性能却并不令人满意, 除非用户能在价格昂贵的硬件上大量投资。第三是伸缩性上的限制, 关系数据库技术在有效支持应用和数据复杂性上的能力是受限制的。

虽然如此, 现在关系型数据库系统产品仍在世界各地的许多领域得到了广泛的应用, 尤其是在图书馆的图书管理系统, 会计核算软件、商品仓储管理系统和具有一定约束要求的计算机辅助设计等方面。

参考文献

- [1] 成昊 王诚君. Visual Foxpro 程序设计教程[M]. 科学出版社, 2006.
- [2] 闪四清. 编写SQL Server系统管理指南[M]. 清华大学出版社, 2001.
- [3] 张健沛. 关系数据库系统Delphi及其应用[M]. 中国水利水电出版社, 1999.