

本书配有光盘提供

在Linux下基于web

的数据库应用整体解决方案

LINUX

数据库实用指南

LINUX

数据库
及
应用工具大全

软件工具列表

- ▲ Msql - 2 _ 0
- ▲ Mysql - 3
- ▲ Delphimy
- ▲ Genquery
- ▲ Javaclic
- ▲ Jmsl30b
- ▲ Mxodbc - 1
- ▲ Billowmy
- ▲ Webboard
- ▲ Mysqlwin
- ▲ Mysqlcom
- ▲ Mysqlgui
- ▲ Dbf2mysq.....

風暴系列

蒋雷波 著
符 鹏



北京腾图电子出版社

目 录

第一部分 数据库技术

第一章 数据库方法	(1)
1.1 数据库技术的发展及概述	(1)
1.2 数据库的体系结构	(1)
1.2.1 三级模式的结构	(1)
1.2.2 数据库系统	(1)
1.2.3 分布式数据管理系统	(2)
1.3 数据模型	(2)
1.3.1 实体-联系方法(Entity-Relationship Approach)也叫 E-R 方法。	(2)
1.3.2 结构数据模型? 也称数据模型	(2)
1.3.3 数据库设计	(4)
1.4 关系数据库语言 SQL(Structure Query Language)	(5)
1.4.1 SQL 发展历史	(5)
1.4.2 SQL 语言的特点	(5)
1.4.3 SQL 语言元素	(6)
1.4.3.1 SQL 支持的主要数据类型	(6)
1.4.3.2 标识符	(7)
1.4.3.3 SQL 支持的运算符	(7)
1.4.3.4 变量	(8)
1.4.3.5 流控制语句	(8)
1.4.4 函数	(10)
1.4.4.1 系统函数	(10)
1.4.4.2 日期函数	(10)
1.4.4.3 字符串函数	(10)
1.4.4.4 数学函数	(11)
1.4.4.5 集合函数	(12)
1.4.4.6 文本和图像函数	(12)
1.4.4.7 转换函数	(12)
1.5 表、视图与索引	(12)
1.5.1 表	(12)
1.5.1.1 建立数据表	(12)
1.5.1.2 修改表	(13)
1.5.1.3 删除表	(14)
1.5.2 表数据操作	(14)

1.5.2.1 添加数据	(14)
1.5.2.2 删除数据	(15)
1.5.3 索引	(16)
1.5.4 视图	(16)
1.5.4.1 建立视图	(16)
1.5.5 数据完整性	(16)
1.5.6 查询	(16)
1.5.6.1 简单查询	(17)
1.5.6.2 FROM 子句	(18)
1.5.6.3 选用 WHERE 子句限定搜索条件	(18)
1.5.6.4 查询结果排序	(19)
1.5.7 统计	(20)
1.5.7.1 GROUP BY 子句	(20)
1.5.7.2 COMPUTE 子句	(21)
1.5.8 利用查询结果创建新表	(22)
1.5.9 使用 UNION 运算符实现多查询联合	(22)
1.5.10 连接	(23)
1.5.10.1 等值连接和自然连接	(23)
1.5.10.2 不等连接	(23)
1.5.10.3 自连接	(23)
1.5.10.4 外连接	(23)
1.5.11 子查询	(24)
1.5.11.1 [NOT] IN 子查询	(24)
1.5.11.2 [NOT] EXISTS 子查询	(24)
1.5.11.3 由比较运算符引出的子查询	(24)
1.6 关系数据库概述	(25)
1.6.1 关系模型的基本概念	(25)
1.6.2 关系数据库	(26)
1.6.3 关系代数	(27)
1.6.4 关系模型的完整性	(29)
1.6.5 数据库的安全性、完整性和并发控制	(30)
1.6.6 数据库的安全性	(30)
1.6.6.1 保证数据安全的一般方法	(30)
1.6.6.2 特定系统中的安全措施。	(30)
1.6.7 数据库的完整性	(31)
1.6.8 并发控制	(32)
1.6.9 数据库的后备和恢复	(33)
1.7 关系数据库设计理论	(34)
1.7.1 需求分析和数据分析	(34)
1.7.2 概念设计	(34)

1.7.3 逻辑设计	(34)
1.7.4 数据库实施和维护	(35)
1.7.5 关系数据库设计理论	(35)
1.7.5.1 函数依赖	(35)
1.7.5.2 属性集团包	(36)
1.7.5.3 规范化设计方法	(37)

第二部分 MySQL 数据库服务器

第二章 MySQL 概况介绍	(40)
2.1 连接 MySQL 步骤	(40)
2.1.1 连接 MySQL	(40)
2.1.2 连接到 MySQL 数据库服务器	(41)
2.3 MySQL 语言参考介绍	(42)
2.3.1 MySQL 的数据类型和表	(42)
2.3.1.1 MySQL 数据类型	(42)
2.3.1.2 MySQL 数据库表	(44)
2.3.1.3 表级操作命令	(44)
2.3.2 MySQL 数据类型简介	(45)
2.3.2.1 字符串数据类型	(45)
2.3.2.2 数值数据类型	(45)
2.3.2.3 NULL	(45)
2.3.2.4 数据库、表、索引和字段名	(45)
2.3.3 字段类型	(46)
2.3.3.1 数据长度信息说明.	(47)
2.3.3.2 数据类型	(47)
2.3.3.3 TIMESTAMP 时间戳数据类型	(48)
2.3.3.4 TEXT 和 BLOB 数据类型	(48)
2.3.3.5 ENUM 数据类型	(48)
2.3.3.6 SET 数据类型	(49)
2.3.4 定义合适数据类型字段	(49)
2.3.5 字段索引	(49)
2.3.6 多字段索引	(50)
2.3.7 数据类型映射图	(50)
2.3.8 在 SELECT 语句和 WHERE 子句中使用函数	(51)
2.3.8.1 分组(Grouping)函数	(51)
2.3.8.2 一般算数操作	(51)
2.3.8.3 Bit 运算函数	(51)
2.3.8.4 逻辑操作	(51)
2.3.8.5 比较运算符	(52)
2.3.8.6 字符串比较	(53)
2.3.8.7 流程控制函数	(53)

2.3.8.8	算术函数	(53)
2.3.8.9	字符串处理函数	(56)
2.3.8.10	日期和时间函数	(58)
2.3.8.11	其他杂项函数	(62)
2.3.8.12	GROUP BY子句使用的函数	(63)
2.3.9	建库 Create database	(64)
2.3.10	删库 Drop database.	(64)
2.3.11	建表 CREATE TABLE	(64)
2.3.12	改变表定义 ALTER TABLE	(66)
2.3.13	删除表 DROP TABLE	(67)
2.3.14	删除记录 DELETE	(67)
2.3.15	SELECT 查询	(67)
2.3.16	Join 连接	(69)
2.3.17	INSERT 插入数据	(69)
2.3.18	REPLACE 数据替换	(69)
2.3.19	LOAD DATA INFILE 从文件中导入数据	(69)
2.3.20	更新记录 UPDATE	(71)
2.3.21	显示表、字段信息 SHOW	(71)
2.3.22	EXPLAIN	(71)
2.3.23	DESCRIBE 语法	(74)
2.3.24	LOCK TABLES 锁定表	(74)
2.3.25	设置选项 SET OPTION	(74)
2.3.26	GRANT 授权	(75)
2.3.27	创建索引 CREATE INDEX	(75)
2.3.28	删除索引 DROP INDEX	(75)
2.3.29	注释 Comment	(75)
2.3.30	CREATE FUNCTION 创建函数	(76)
2.3.31	MySQL 保留字	(77)
2.4	MySQL 系统管理和配置	(78)
2.4.1	介绍	(78)
2.4.2	用户权限体系	(79)
2.4.2.1	权限表	(79)
2.4.2.2	host 权限表;	(80)
2.4.2.3	user 用户表	(80)
2.4.2.4	db 数据库表	(81)
2.4.3	管理权限系统例子	(81)
2.4.3.1	User 权限表	(82)
2.4.3.2	Db 权限表	(83)
2.4.3.3	Host 表	(83)
2.5	MySQL 支持的操作系统?	(84)

2.5.1 MySQL 不支持的功能	(84)
2.5.1.1 子查询	(84)
2.5.1.2 SELECT INTO TABLE	(85)
2.5.1.3 事务管理	(85)
2.5.1.4 触发器	(85)
2.5.1.5 外键	(85)
2.5.1.6 视图	(85)
2.5.1.7 - 注释起始符	(85)
2.6 MySQL 的主要特征	(86)
2.7 命令行命令	(87)
2.7.1 简介	(87)
2.7.2 核心 MySQL	(88)
2.7.2.1 MySQL 客户端程序	(88)
2.7.2.2 mysqlaccess	(89)
2.7.2.3 mysqladmin	(90)
2.7.2.4 mysqld	(91)
2.7.2.5 mysqldump	(93)
2.7.2.6 mysqlshow	(94)
2.7.2.7 isamchk	(94)
2.7.2.8 isamlog	(96)
2.7.2.9 safe-mysqld	(97)
2.7.3 非核心程序	(97)
2.7.3.1 comp-err	(97)
2.7.3.2 msqld2mysql	(97)
2.7.3.3 mysqlbug	(97)
2.7.3.4 perror	(98)
2.7.3.5 replace	(98)
2.7.3.6 which1	(98)
2.7.3.7 zap	(99)
2.8 C语言程序库 API	(99)
2.8.1 简介	(99)
2.8.2 准备知识	(99)
2.8.3 客户端函数	(101)
2.8.1.1 mysql-affected-rows	(101)
2.8.1.2 mysql-close	(102)
2.8.1.3 mysql-connect	(102)
2.8.1.4 mysql-create-db	(103)
2.8.1.5 mysql-data-seek	(103)
2.8.1.6 mysql-drop-db	(103)
2.8.1.7 mysql-eof	(103)
2.8.1.8 mysql-error	(104)
2.8.1.9 mysql-fetch-field	(104)

2.8.1.10	mysql-fetch-lengths	(104)
2.8.1.11	mysql-fetch-row	(104)
2.8.1.12	mysql-field-seek	(104)
2.8.1.13	mysql-free-result	(104)
2.8.1.14	mysql-get-client-info	(104)
2.8.1.15	mysql-get-host-info	(105)
2.8.1.16	mysql-get-proto-info	(105)
2.8.1.17	mysql-get-server-info	(105)
2.8.1.18	mysql-insert-id	(105)
2.8.1.19	mysql-list-dbs	(105)
2.8.1.20	mysql-list-fields	(105)
2.8.1.21	mysql-list-processes	(106)
2.8.1.22	mysql-list-tables	(106)
2.8.1.23	mysql-num-fields	(106)
2.8.1.24	mysql-num-rows	(106)
2.8.1.25	mysql-query	(107)
2.8.1.26	mysql-real-query	(107)
2.8.1.27	mysql-reload	(107)
2.8.1.28	mysql-select-db	(108)
2.8.1.29	mysql-shutdown	(108)
2.8.1.30	mysql-stat	(108)
2.8.1.31	mysql-store-result	(108)
2.8.1.32	mysql-use-result	(108)
2.9	MySQL 数据库服务器的 Perl 接口	(109)
2.9.1	简介	(109)
2.9.2	创建数据库连接	(110)
2.9.3	改变连接端口	(110)
2.9.3.1	SelectDB	(110)
2.9.3.2	ListFields	(111)
2.9.3.3	Query	(111)
2.9.3.4	ListDBS	(111)
2.9.3.5	ListTables	(112)
2.9.3.6	FetchCol	(112)
2.9.3.7	FetchHash	(112)
2.9.3.8	FetchRow	(113)
2.9.3.9	DataSeek	(113)
2.9.4	预先处理二进制字符串	(113)
2.9.5	释放句柄	(113)
2.9.6	数据库句柄	(113)
2.9.7	执行句柄 Statement Handle	(113)
2.10	JDBC 编程技术	(114)
2.10.1	标准 Java 接口、对象和异常	(114)

2.10.1.1 CallableStatement 接口	(115)
2.10.1.2 Connection 接口	(115)
2.10.1.3 DatabaseMetaData 接口	(115)
2.10.1.4 Driver 接口	(115)
2.10.1.5 PreparedStatement 接口	(115)
2.10.1.6 ResultSet 接口	(115)
2.10.1.7 ResultSetMetaData 接口	(115)
2.10.1.8 Statement 接口	(116)
2.10.2 Java 对象	(116)
2.10.2.1 Date 对象	(116)
2.10.2.2 DriverManager 对象	(116)
2.10.2.3 DirverPropertyInfo 对象	(116)
2.10.2.4 Time 对象	(116)
2.10.2.5 TimeStamp 对象	(116)
2.10.2.6 Types 对象	(116)
2.10.3 JDBC 的异常类	(117)
2.10.3.1 DataTruncation 异常	(117)
2.10.3.2 SQLException 异常	(117)
2.10.3.3 SQLWaring 异常	(117)
2.10.4 JDBC 驱动程序	(117)
2.10.4.1 Statement	(119)
2.10.4.2 结果集	(120)
2.11 MM.MySQL 的 JDBC 应用	(121)
2.11.1 尚未实现的 JDBC 1.1 方法	(122)
2.11.2 安装 MM.MySQL 驱动程序包	(122)
2.11.3 设置类路径 CLASSPATH	(123)
2.11.4 建立数据库连接	(123)
2.11.5 ResultSet 结果集	(125)

第三部分

第三章 Postgres 概述	(131)
3.1 PostgreSQL 数据库管理系统概述?	(131)
3.1.1 PostgreSQL 的海量数据容量	(133)
3.1.2 PostgreSQL 的前端 GUI 工具	(133)
3.1.3 PostgreSQL 的集成开发环境	(134)
3.1.4 PostgreSQL 的 ODBC 驱动程序	(134)
3.1.5 PostgreSQL 的 JDBC 驱动程序	(134)
3.1.6 PostgreSQL 的 Java	(135)
3.2 Postgres 的一般描述	(135)
3.2.1 创建一个新的类	(135)

3.2.2	实例化一个类	(135)
3.2.3	查询一个类	(136)
3.2.4	重定向 SELECT 查询	(136)
3.2.5	更新记录	(137)
3.2.6	删除记录	(137)
3.2.7	使用聚合函数	(137)
3.3	高级的 Postgres SQL 特性	(138)
3.3.1	继承性	(138)
3.3.2	原子值	(139)
3.3.3	数组	(139)
第四章	PostgreSQL 语法	(142)
4.1	PostgreSQL 基本语法	(142)
4.1.1	关键字	(142)
4.1.1.1	保留关键字	(142)
4.1.1.2	非保留字	(144)
4.1.1.3	注释	(146)
4.1.1.4	命名习惯	(146)
4.1.1.5	常量	(146)
4.1.1.6	字符串常量	(146)
4.1.1.7	整型数常量	(146)
4.1.1.8	浮点数常量	(147)
4.1.1.9	Postgres 用户自定义型常量	(147)
4.1.1.10	数组常量	(147)
4.1.2	字段和列的区别	(147)
4.1.2.1	字段	(147)
4.1.2.2	列	(147)
4.1.2.3	操作符	(148)
4.1.2.4	表达式	(148)
4.1.2.5	参数	(148)
4.1.2.6	函数表达式	(148)
4.1.2.7	聚合表达式	(148)
4.2	数据类型	(149)
4.2.1	数值型数据类型	(150)
4.2.2	序列数型数据类型	(151)
4.2.3	货币型数据类型	(151)
4.2.4	字符型数据类型	(151)
4.2.5	日期/时间数据类型	(152)
4.2.6	SQL92 时间表示规范	(153)
4.2.6.1	Date/Time 表示风格	(153)
4.2.6.2	时区	(153)

4.2.6.3	Date/Time 输入格式	(154)
4.2.6.4	datetime 类型	(156)
4.2.6.5	timespan 类型	(157)
4.2.6.6	abstime 类型	(157)
4.2.6.7	reltime 类型	(158)
4.2.6.8	timestamp 类型	(158)
4.2.6.9	interval	(158)
4.2.6.10	tinterval 数据类型	(158)
4.3	几何型数据类型	(158)
4.3.1	Point 数据类型	(159)
4.3.2	lseg 线段类型	(159)
4.3.3	Box 矩形类型	(159)
4.3.4	Path 路径类型	(159)
4.3.5	Polygon 多边形类型	(160)
4.3.6	Circle 圆类型	(160)
4.4	操作符	(160)
4.4.1	操作符优先级	(161)
4.4.2	普通操作符	(161)
4.4.3	数值型操作符	(162)
4.4.4	几何操作符	(163)
4.4.5	时间间隔操作符	(163)
4.4.6	IP V4 CIDR 操作符	(164)
4.4.7	IP V4 INET 操作符	(165)
4.5	函数	(165)
4.5.1	SQL 函数	(165)
4.5.2	数学函数	(166)
4.5.3	字符串函数	(166)
4.5.4	Date/Time 日期/时间函数	(167)
4.5.5	几何函数	(168)
4.5.6	IP V4 函数	(170)
第五章	PostgreSQL 应用开发	(171)
5.1	PostgreSQL 系统 API	(171)
5.1.1	createdb	(171)
5.1.2	createlang	(172)
5.1.3	create user	(173)
5.1.4	dropdb	(174)
5.1.5	droplang	(175)
5.1.6	dropuser	(176)
5.1.7	ecpg	(177)

5.1.8	pgaccess	(177)
5.1.9	pgadmin	(179)
5.1.10	pg-dump	(179)
5.1.11	pg-dumpall	(181)
5.1.12	pgtclsh	(182)
5.1.13	pgtksh	(182)
5.1.14	vacuumdb	(183)
5.2	Postgres 数据库系统组成	(184)
5.2.1	Postgres 系统表	(184)
5.2.2	内部函数	(186)
5.2.3	联编的 C 语言函数	(186)
5.2.4	基于基本数据类型的 C 语言函数	(186)
5.2.5	面向复合数据类型的 C 语言函数	(190)
5.2.6	函数重载	(191)
5.3	Libpq 编程	(191)
5.3.1	数据库连接函数	(191)
5.3.2	查询执行函数	(196)
5.3.3	异步查询处理	(199)
5.3.4	Fast Path	(201)
5.3.5	异步消息	(202)
5.3.6	COPY 命令的相关函数	(203)
5.3.7	libpq 跟踪函数	(204)
5.3.8	libpq 控制函数	(204)
5.3.9	环境变量	(205)
5.3.10	大对象操作接口	(215)
5.3.10.1	创建大对象	(215)
5.3.10.2	导入大对象	(215)
5.3.10.3	导出大对象	(215)
5.3.10.4	打开一个现有的大对象	(216)
5.3.10.5	写大对象	(216)
5.3.10.6	读大对象	(216)
5.3.10.7	定位大对象	(216)
5.3.10.8	关闭大对象标识符	(216)
5.3.11	内嵌函数	(216)
5.4	pgtcl	(222)
5.4.1	pgtcl 命令	(222)
5.4.1.1	pg-connect	(223)
5.4.1.2	pg-disconnect	(223)
5.4.1.3	pg-conndefaults	(223)
5.4.1.4	pg-exec	(224)

5.4.1.5	pg-result	(224)
5.4.1.6	pg-select	(225)
5.4.1.7	pg-listen	(225)
5.4.1.8	pg-lo-creat	(226)
5.4.1.9	pg-lo-open	(226)
5.4.1.10	pg-lo-close	(226)
5.4.1.11	pg-lo-read	(226)
5.4.1.12	pg-lo-write	(227)
5.4.1.13	pg-lo-lseek	(227)
5.4.1.14	pg-lo-tell	(227)
5.4.1.15	pg-lo-unlink	(228)
5.4.1.16	pg-lo-import	(228)
5.4.1.17	pg-lo-export	(228)

第六章 PostgreSQL 命令 (230)

6.1	ABORT 命令	(230)
6.2	ALTER TABLE 命令	(230)
6.3	ALTER USER 命令	(232)
6.4	BEGIN 命令	(233)
6.5	CLOSE 命令	(234)
6.6	CLUSTER 命令	(235)
6.7	COMMIT 命令	(235)
6.8	COPY 命令	(236)
6.9	CREATE AGGREGATE 命令	(238)
6.10	CREATE DATABASE 命令	(239)
6.11	CREATE FUNCTION 命令	(240)
6.12	CREATE INDEX 命令	(241)
6.13	CREATE LANGUAGE 命令	(242)
6.14	CREATE OPERATOR 命令	(245)
6.15	CREATE RULE 命令	(245)
6.16	CREATE SEQUENCE 命令	(247)
6.17	CREATE TABLE 命令	(249)
6.18	CREATE TABLE AS 命令	(258)
6.19	CREATE TRIGGER 命令	(259)
6.20	CREATE TYPE 命令	(260)
6.21	CREATE USER 命令	(262)
6.22	CREATE VIEW 命令	(263)
6.23	DECLARE 命令	(264)
6.24	DELETE 命令	(266)
6.25	DROP AGGREGATE 命令	(267)

6.26	DROP DATABASE 命令	(267)
6.27	DROP FUNCTION 命令	(268)
6.28	DROP INDEX 命令	(269)
6.29	DROP LANGUAGE 命令	(269)
6.30	DROP OPERATOR 命令	(270)
6.31	DROP RULE 命令	(270)
6.32	DROP SEQUENCE 命令	(271)
6.33	DROP TABLE 命令	(272)
6.34	DROP TRIGGER 命令	(272)
6.35	DROP TYPE	(273)
6.36	DROP USER 命令	(273)
6.37	DROP VIEW 命令	(274)
6.38	END 命令	(275)
6.39	EXPLAIN 命令	(275)
6.40	FETCH 命令	(276)
6.41	GRANT 命令	(278)
6.42	INSERT 命令	(280)
6.43	LISTEN 命令	(281)
6.44	LOAD 命令	(282)
6.45	LOCK 命令	(282)
6.46	MOVE 命令	(284)
6.47	NOTIFY 命令	(285)
6.48	RESET 命令	(286)
6.49	REVOKE 命令	(286)
6.50	ROLLBACK 命令	(288)
6.51	SELECT 命令	(289)
6.52	SELECT INTO 命令	(294)
6.53	SET 命令	(294)
6.54	SHOW 命令	(296)
6.55	TRUNCATE 命令	(297)
6.56	UNLISTEN 命令	(297)
6.57	UPDATE 命令	(298)
6.58	VACUUM 命令	(299)
第七章 Postgre Java 编程		(301)
7.1	运行 JDBC	(301)
7.1.1	编译驱动程序	(301)
7.1.2	安装驱动程序	(301)
7.2	运行 JDBC 的准备工作	(301)

7.2.1	调用驱动程序	(301)
7.2.1.1	导入 JDBC 驱动程序	(301)
7.2.1.2	装入驱动程序到内存	(302)
7.2.1.3	连接数据库	(302)
7.2.1.4	发送查询、处理结果集	(302)
7.2.1.5	调用 Statement 接口	(303)
7.2.1.6	调用 ResultSet 接口	(303)
7.2.1.7	执行更新操作	(303)
7.2.1.8	关闭数据库连接	(303)
7.2.1.9	处理大对象数据	(303)
7.3	Postgres 对 JDBC API 的扩展	(304)
7.3.1	调用扩展特性	(305)
7.3.1.1	Class postgresql.Connection	(305)
7.3.1.2	Class postgresql.Fastpath	(306)
7.3.1.3	Class postgresql.fastpath.FastpathArg	(309)
7.3.2	几何型数据类型	(310)
7.3.2.1	Class postgresql.geometric.PGbox	(310)
7.3.2.2	Class postgresql.geometric.PGcircle	(312)
7.3.2.3	Class postgresql.geometric.PGline	(315)
7.3.2.4	Class postgresql.geometric.PGseg	(317)
7.3.2.5	Class postgresql.geometric.PGpath	(319)
7.3.2.6	Class postgresql.geometric.PGpoint	(321)
7.3.2.7	Class postgresql.geometric.PGpolygon	(324)
7.3.3	处理大对象数据	(326)
7.3.3.1	Class postgresql.largeobject.LargeObject	(326)
7.3.3.2	Class postgresql.largeobject.LargeObjectManager	(330)
7.3.4	对象序列化	(332)
7.3.4.1	Class postgresql.util.Serialize	(334)
7.3.5	工具类	(338)
7.3.5.1	Class postgresql.util.PGmoney	(338)
7.3.5.2	Class postgresql.util.PGobject	(339)
7.3.5.3	Class postgresql.util.PGtokenizer	(340)
7.3.5.4	Class postgresql.util.UnixCrypt	(343)

第一部分 数据库技术

第一章 数据库方法

数据库技术是管理数据的一种最新方法,它研究如何组织数据,存储数据,如何能够高效地获取和处理数据的方法。

1.1 数据库技术的发展及概述

阶段 1: 人工管理阶段(50 年代中期)

没有统一的数据管理软件

阶段 2: 文件系统阶段(50-60 年代中)

有统一的软件处理数据(如, COBOL 语言)

数据可以组织成:顺序、索引、直接文件形式

处理特点:程序+数据

阶段 3: 数据库阶段(60 年代末至今)

有统一的数据管理软件,管理数据

数据具有独立性

强调面向企业面向系统组织数据

1.2 数据库的体系结构

1.2.1 三级模式的结构

数据库的体系结构分成三级:

内模式又称存储模式,是数据在数据库系统的内部表示,即为数据的物理结构和存储方式的描述。

外模式又称用户模式,表示用户能看到的数据。

概念模式又称逻辑模式,是 DB 中全体数据的逻辑结构和特征的描述。

DBMS 在这三级结构之间提供了两层映象:

外模式/模式映象、模式/内模式映象。

1.2.2 数据库系统

一个数据库系统是一个实际可运行的,按照一定的数据模型存贮,维护并向应用系统提供数据或信息支持的系统。它由以下部分组成:

数据库(Database, DB)

数据库管理系统(Database Management System, DBMS)

DBMS 的主要功能:

a. 数据库定义:包括全局逻辑结构(Schema)的定义。局部逻辑结构(subschema)的定义,安全保密等的定义。

b. 数据库管理:数据存取、完整性、安全性、并先控制等。

c. 数据库的建立和维护、建库、库的迁移,再组织,数据的后备和恢复、性能监控等。

应用系统开发工具(程序设计软件)

数据库管理员(Database Administrator, DBA)

1.2.3 分布式数据管理系统

通讯模式:

client/server server/server client/App server/DB server

1.3 数据模型

在 DB 技术中我们使用模型的概念描述数据库的结构和语义。即描述现实世界中对象集及对象集之间关系的模型称作数据模型。

目前广泛使用的数据模型分为两种类型:

概念数据模型:描述信息世界中数据及数据之间关系的模型,它独立于计算机。

结构数据模型:描述计算机世界中,数据及数据之间表示、存贮、处理特征的模型,它依赖于具体的计算机。

1.3.1 实体—联系方法(Entity-Relationship Approach)也叫 E-R 方法。

实体:实体表示一个现实和抽象事物的集合,这些事物必须具有相同的属性和特征。

实体用长方形表示,框内写上实体名,如部门用 DEPT 表示。

联系:表示实体和实体之间的关系或联系

实体之间的联系用菱形表示,在菱形边框内写上联系名。两个实体之间联系的基数(类型)可以是 1:1, 1:n, m:n, 可写在连线旁,例如,实体之间的联系为:

属性,用于描述实体的性质和特征,用椭圆形表示。例如:

实体实例:实体中的一个元素就是这个实体的实例。

例如:计算机系、电子系、职工张三、李四等。

属性值,为一个属性的取值,例如,ENAME 的取值、王明、张丽等。

1.3.2 结构数据模型,也称数据模型

结构数据模型直接面向数据库中数据的逻辑结构。它一方面把信息世界的信息在机器世界中以数据形式存储,同时,描述了数据世界中存储的数据集上的操作。因此一个数据模型由三个部分组成:

对象类型的集合:操作符的集合和完整性规则的集合。

对象类型的集合:描述全部数据的逻辑结构。

操作符集合:表示数据结构支持的操作。

完整性规则集合:描述数据及其联系所指定的制约或依存规则。

目前,实际的数据库系统支持的数据模型有:

层次模型、网状模型、关系模型和 OO 模型。

层次模型：

层次模型是用树结构表示实体及其实体之间的联系。它的主要特征是一棵有向树。树的结点是记录类型。根结点只有一个，根结点以外的结点有且只有一个父结点。

图中 R1 是根，R2, R3 是 R1 的子女，R2, R3 为兄弟，R2, R4, R5 称为叶子。

IMS (Information Management SYStem) 是 IBM 公司研制的最早的大型数据库系统，也是层次模型的典型代表。

网状模型

用网络结构表示实体以其实体之间联系的数据模型称为网状模型。网状模型满足下列条件：

- a. 有一个以上结点无双亲
- b. 至少有一个结点有多于一个双亲

网状模型数据库的代表是 DBTG 系统。DBTG(Data Base Task Group 数据库任务组)是美国 CODASYL(conference On Data System Language 数据系统语言协商会)下属的一个组织，主要研究数据库语言。1969 年 DBTG 小组提出了一份报告，即有名的“DBTG 报告”。1971 年报告被正式通过。

关系模型

用表格结构表示实体以及实体间联系的数据模型叫关系模型。关系模型是由若干“关系”(Relational)组成的集合。

在网状和层次模型中用指针的链接表示实体之间的联系。

例如：

而在关系模型中，通过子女关系继承父亲关系的主码表示关系或表格之间的联系。例如：

DEPT 关系:EMP 关系

DNO	DNAME	DIRECTOR
D ₁	Computer	Wang
D ₂	Radio	Li
D ₃	Automation	Zhang

EMPNO	ENAME	JOB	SAL	DEPTNO
E ₁	L ₁	clerk	500	D ₁
E ₂	wang	manager	900	D ₁
E ₃	liu	worker	450	D ₁
E ₅	gao	analyst	850	D ₂

“关系”本身涉及存储结构、访问技术等细节，与网状、层次模型比起来实现较复杂，效率也低。但提供给用户的界面较简单，容易为初学者接受。

面向对象模型(Object - Oriented model)

面向对象模型中最基本的概念是对象(Object)和类(Class)。对象的概念与记录元组的概念相似，但远比它们复杂。每个对象都有一个唯一的标识符，而且把一个状态(State)和一个行为封装在一起。对象的状态是该对象属性值的集合，对象的行为是在对象状态上操作的方法(程序代码)的集合。

类的概念类似于表格或记录型，它把共享同一属性集合和方法集合的所有对象组合在一起，命名为类。

面向对象模型特点：