

数据库系统概论

A First Course in Database

上海交通大学
电类工程导论

2018-3

数据库的重要地位

数据库技术产生于**50年前**，是数据管理的最新技术，是计算机科学的重要分支。首先使用“DataBase”一词的是美国系统发展公司在为美国海军基地在60年代研制数据中引用。1969年，IBM公司Mc Gee等人开发的层次式数据库系统的IMS系统发表，它可以让多个程序共享数据库。1969年10月，CODASYL(1959 Conference on Data Systems Languages)数据库研制者提出了网络模型数据库系统规范报告DBTG 。

数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。

衡量信息化程度的重要标志

信息资源已成为各个部门的重要财富和资源。

数据库市场应用

IT 时代主题

一、智慧地球 IBM(e-business/ e-business on demand/...)

实体和信息基础设施不应该分开建设，而应该是统一的智能基础设施

二、社交网络服务 SNS（Social Network Service）

通过网络这一载体把人们连接起来，从而形成具有某一特点的团体

硬件、软件、服务及应用

e-mail -> BBS -> 即时通信（QQ）和博客（Blog） -> 微信

生活：线下到线上 O2O -> 生态

三、物联网/移动应用 The Internet of things(工业4.0基础)

通过RFID、红外感应、全球定位、激光扫描等传感设备，按约定的协议，把物品与互联网相连接，进行信息交换和通信，以实现对物品的智能化识别、定位、跟踪、监控和管理的一种网络。

四、云计算 cloud computing (阿里云等)

基于互联网的相关服务的增加、使用和交付模式，通常涉及通过互联网来提供动态易扩展且经常是虚拟化的资源。

五、大数据 Volume, Variety, Value, Velocity (Hadoop)

从各种各样类型的数据中，快速获得有价值信息的能力

数据体量巨大(TB到PB级别)；

数据类型繁多(网络日志, 视频, 图片, 地理位置信息, 微信+微博)；

价值密度低(视频监控, 有用的数据仅仅有一两秒)。

处理速度快；

六、企业应用(ERP BI/HR/CRM/SRM)

SAP/Oracle/IBM/Microsoft/HP...

普华永道(PWC)、德勤(DTT)、毕马威(KPMG)、安永(EY)

其它领域

一、手机/PDA等移动设备

移动数据库 Sqlite/ASA

二、财务软件

Oracle/Access/Sql server

三、普通网站(淘宝)

MySQL

四、游戏

Access/Ms sql server/My sql/Oracle

五、办公自动化 OA

企业除了生产控制之外的一切信息处理与管理的集合, 面向不同层次的使用者, 便有不同的功能表现: 高层领导的决策支持系统 (DSS) + 中层管理的信息管理系统 (IMS) + 普通员工事务/业务处理系统。

Mysql / sql server / Oracle

六、零售业: WAL—MART(物流/CRM/SRM…… 多次列为全球榜首)

七、医疗 DB2(IBM 智慧地球) /Oracle/Sql server/ASE/VFP

常用的数据库管理系统

- **Oracle** 大型关系数据库管理系统 (**Ellison** 网格 + 云数据库)
- **DB2** **IBM**公司的运行于主机 + **AS/400**的大型数据库。
目前支持各种平台(如:**Aix/linux/windows**)
- **MS SQL Server** 是建立在 **NT server** 上的**RDBMS**。
- **SAP** 大型“客户机/服务器数据库体系结构”的**DBMS** 具有分布式处理等优点(云 **HANA+ASE+ASA+IQ+ESP+RPS**复制)
- **Teradata** 用于大型数据仓库的专业结构的数据库(平行)
- **My SQL** 运行于**Linux/windows**的免费数据库系统(**TB级**)。
- **PostgreSQL** 是一个开放源码的免费数据库系统(**Ingres**分家)。
- **XBASE** 一种大众化的关系型数据库系统；类似产品有：**DBASE、FOXPRO、ACCESS**等系列产品。
- **Sqlite** 小型数据库，适合移动设备。
- **Informix** 是最早具有“可伸缩、高性能”体系结构的**DBMS**。(**New Era**) -> 被**IBM**收购
- **Ingres** 历史上重要的大型数据库，产生许多商业数据库。
- **VoltDB** 内存开源**OLTP ACID**，**Postgres**和**Ingres**创始人**Mike Stonebraker**领导开发它能在现有的廉价服务器集群上实现每秒数百万次数据处理。

国产数据库管理系统

- **Kingbase ES**: 北京人大金仓信息技术有限公司, 金仓数据库管理系统**Kingbase Enterprise Server** 人民大学数据与知识工程研究所 8%
- **达梦(DM)**系统: 在国防系统的支持下, 华中理工大学研制成功, 实现了**B1**级的安全性, 具有内置多媒体及地理信息数据类型和存储结构功能的大型数据库。
- **神通通用数据库**: 天津神舟通用数据技术有限公司, 致力于大型通用数据库产品的研发和产业化, 以神舟**OSCAR**数据库为基础, 融合南大通用**GBase**和东软**OpenBase**的技术, 形成统一品牌——国产神通数据库。
- **南大通用 8A列存数据库产品**
- **OpenBASE**: 1989年开始立项, 由东大阿尔派(东软)开发, 我国第一个出口国外、具有自主版权、可以替代进口的商品化数据库管理系统。产品包括: **OpenBASE** 多媒体数据库管理系统、**Web**应用服务器、**Mini**嵌入式数据库管理系统、**Secure**安全数据库系统等产品
- **OSCAR**: 北京神舟航天软件技术有限公司, 神舟**OSCAR**数据库系统基于**Client/Server**架构实现, 服务器具有通常数据库管理系统的一切常见功能, 此外还包括一些有助于提高系统对工程数据支持的特别功能, 而客户端则在提供了各种通用的应用开发接口的基础上, 还具有丰富的连接、操作和配置服务器端的能力。
- **iBASE**: 北京 国信贝斯软件有限公司

数据库系统概述

基本概念

- 一、数据 (Data Big data XML Jason)
- 二、数据库/数据仓库 (Database, DB; Data Warehouse DW/DWH)
- 三、数据库管理系统 (Database Management System, DBMS)
- 四、数据库系统 (Database System, DBS)

基本概念-数据

一、数据 (Data/Big data)

- 描述事物的符号记录
- 包括文字(text)、图形(figure)、图像(image)、音频(audio)、视频(video)...

- 数据的含义称为语义，数据和语义是密不可分的

Ex. (李明, 197205, 2000, IEEE)

Jason { “name” :” 李明” , “Bdate” :” 197205” }

XML <name>李明</name> <Bdate>197205</Bdate>

Big data: 数亿用户互联网服务,**RFID**等产生非结构化和结构化数据(**sensors**); 城市交通监控信息;医疗信息;微信;手机位置;用户行为.

基本概念-数据库

二、数据库 (Database, DB)

- 长期存储在计算机（外存储器）内的、有组织的、可共享的数据集合；
- 数据按一定的数据模型组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享；
数据库数据三个基本特点：永久存储、有组织、可共享
- 数据库技术是数据处理最有效的手段。

数据仓库 DW:

- 面向主题(**Subject Oriented**)
较高层次上数据综合、归并和分析，由许多表实现
- 集成的(**Integrate**)
许多表合并存在：同名异义，异名同义，单位统一，字长不一致
- 不可更新的(相对稳定的 **Non-Volatile**)
极少更新或不更新，是稳定的
- 随时间不断变化的数据集合(**Time Variant**)
一定周期更新，不是动态更新

基本概念-数据库管理系统

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

- 是位于用户和操作系统之间的一层数据管理软件, 客户端工具
- 包括如下主要功能:

1. 数据定义功能

数据定义语言 (Data Definition Language, DDL)

2. 数据组织、存储和管理

Data Dictionary, User data, 存取路径,

存取方法: index/hash/顺序 (IQ: 列存储)

3. 数据操纵功能

数据操纵语言 (Data Manipulation Language, DML)

Select/update/Delete/Insert

4. 数据库的事务管理和运行管理 (DCL)

OLTP, 并发, 恢复

5. 数据库的建立和维护功能 (数据导入, Monitor)

6. 其他功能 (API接口, 数据转换, Gateway)

Example:

Oracle, DB2, Ms SQL Server, SAP (ASE+ASA), Informix,

Ingres, My SQL, Access, Dbase+Foxpro (VFP)

OceanBase

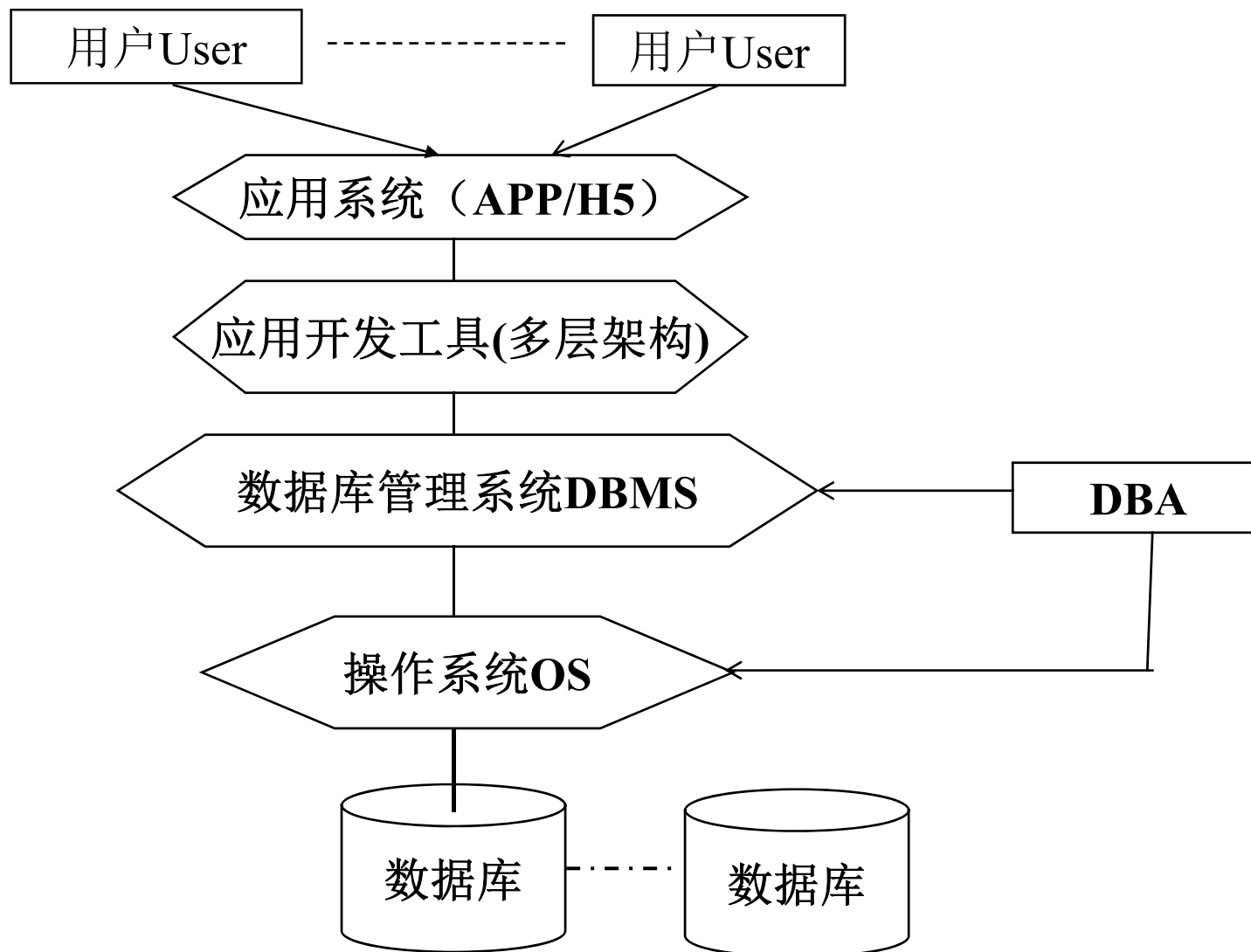
基本概念-数据库系统

四、数据库系统 (Database System, DBS)

- 是指在计算机系统中引入数据库后的系统
- 包括如下构成要素:
 - 数据库(Database=System Engine)
 - 数据库管理系统(Database Management System)
 - 应用开发工具(Developing Tools)
 - 应用系统(Application Systems)
 - 数据库管理员(Database Administrator DBA)
 - 用户(User)

Example: 支付宝系统, 微信系统, SAP系统, 用友ERP

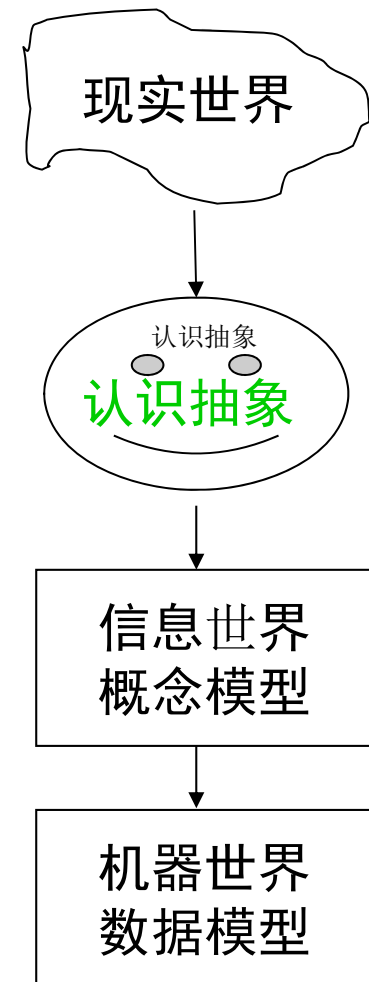
DBS结构



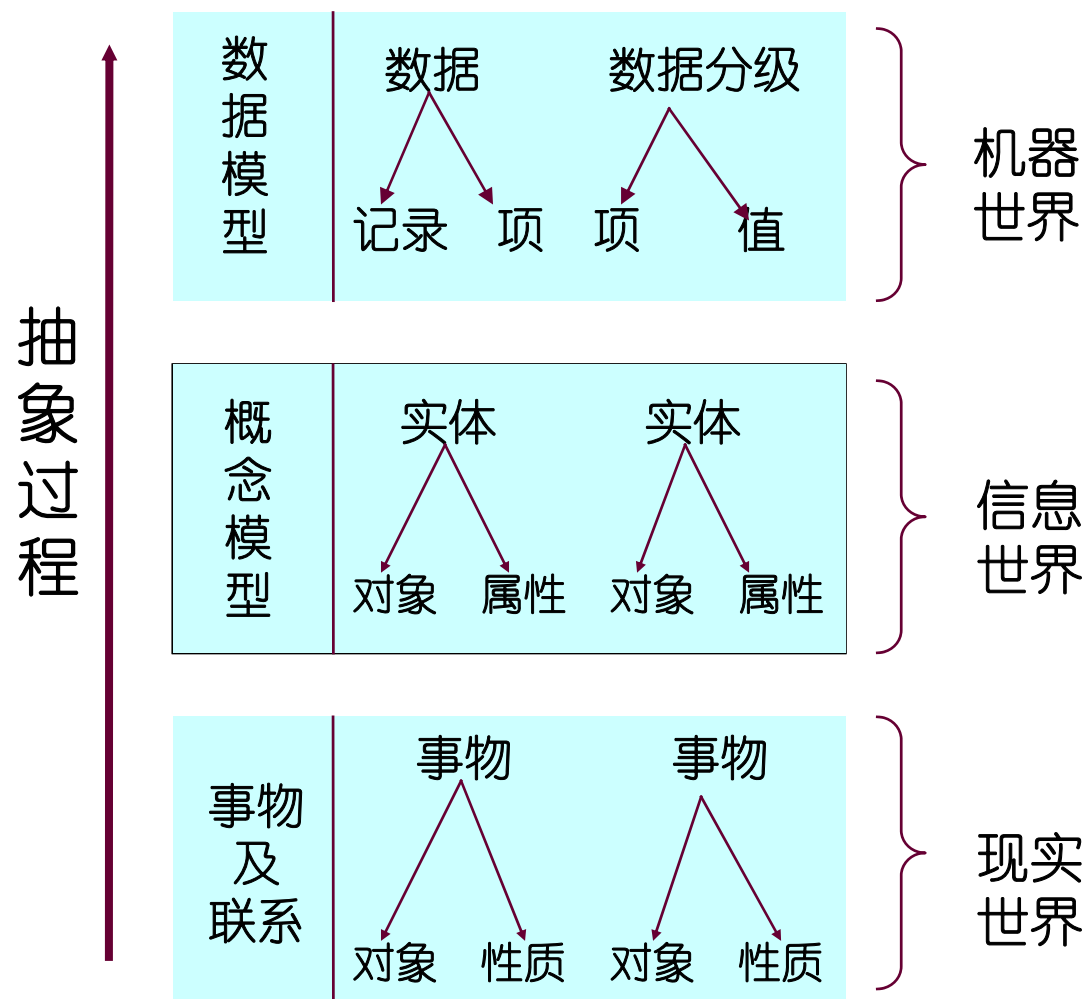
数据模型 (Data Model)

——现实世界数据特征的抽象

- 从现实生活中的客观事物到存放于计算机中的数据，这样一个加工过程可划分为三个领域：
- 现实世界、信息世界、机器世界
- 现实世界是信息之源，是设计**DB**的出发点。
- 概念模型和数据模型是对客观事物的两级抽象描述。
- 数据库的核心问题是数据模型。
- 要得到正确的数据模型，必须首先充分了解客观事物。



三个数据加工领域的关系



概念模型

- **概念模型** 用于信息世界的建模，是现实世界到信息世界的的第一层抽象，是数据库设计人员进行数据库设计的有力工具，也是数据库设计人员和用户之间进行交流的语言，因此概念模型一方面应具有较强的语义表达能力，能够方便、直接地表达应用中的各种语义知识，另一方面它还应该简单、清晰、易于用户理解。
- **主要概念** 有实体、属性、码、域、实体型、实体集、联系等。

实体 (Entity)

- 客观存在并可相互区分的事物叫实体。实体可以是人，也可以是物；可以指实际的对象，也可以指某些概念；可以指事物本身，也可以指事物与事物之间的联系。例如，一个职工，一个学生，一个部门，一门课，学生的一次选课，部门的一次定货。
- 对应OOT中的对象 (Object)。

注意: 10个学生是10个对象或10个实体, 用学号可以区分

属性 (Attribute)

- 实体所具有的某一特性。一个实体可以由若干个属性来刻画。例如，学生实体可以由学号、姓名、年龄、系、年级等属性组成。（7806032，王平，19，男，计算机系，三年级）这些属性值组合起来表征了一个学生。
- 对应OOT中的属性 (Attribute) 。

码 (Key)

- 唯一标识实体的属性集称为码。例如，学号是学生实体的码。
- 对应OOT中的对象标识符 (OID, Object Identifier) 。

主码:Primary Key PK

外码:Foreign Key FK

域 (Domain)

- 某个（些）属性的取值范围。例如，学号的域为7位整数，姓名的域为字符串集合，年龄的域为小于35的整数，性别的域为（男，女）。

Example: int/long/double/real/decimal(15, 2)
varchar(100), image, blob, text
bit

age int null constraint c1 check(age<20)

常用数据模型

- 建立了概念模型后，就可以建立数据模型了
- 数据库领域中最常用的数据模型有四种：
 - 层次模型 (Hierarchical Model)
 - 网状模型 (Network Model)
 - 关系模型 (Relational Model)
 - 面向对象模型 (Object Oriented Model)
- 关系模型是目前广泛采用的数据模型
- 层次模型和网状模型基本上成为了历史
- 面向对象模型是目前数据库研究热点之一，已开始商业应用 (Java/UML)

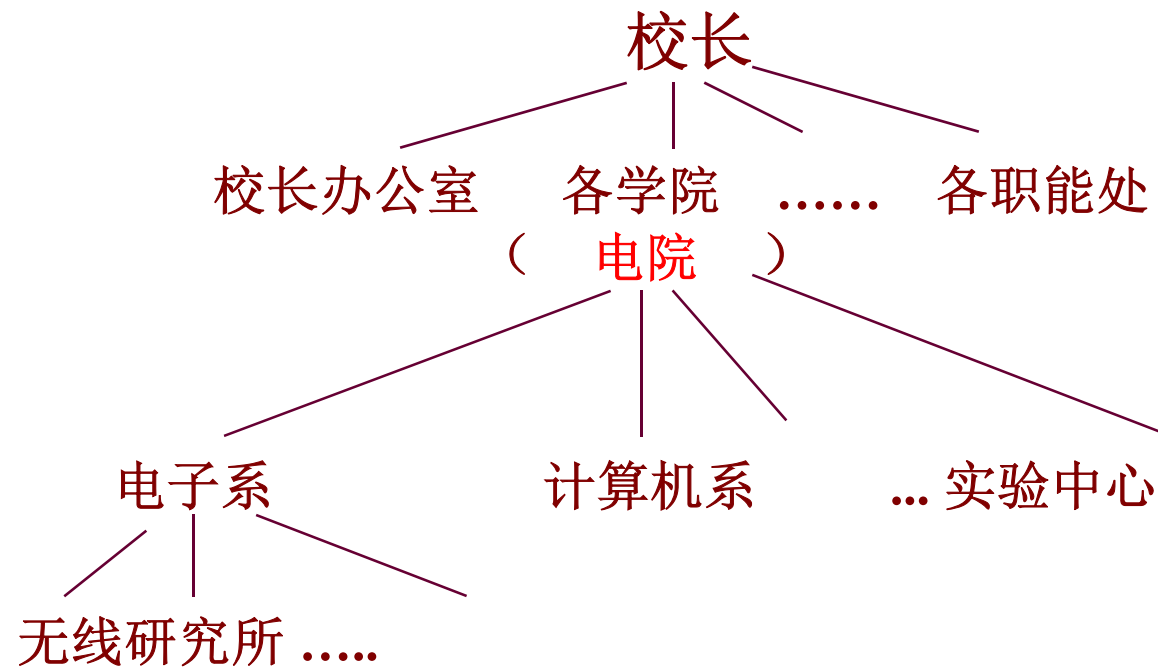
层次模型

- 描述层次（树形）结构的模型。

特点：

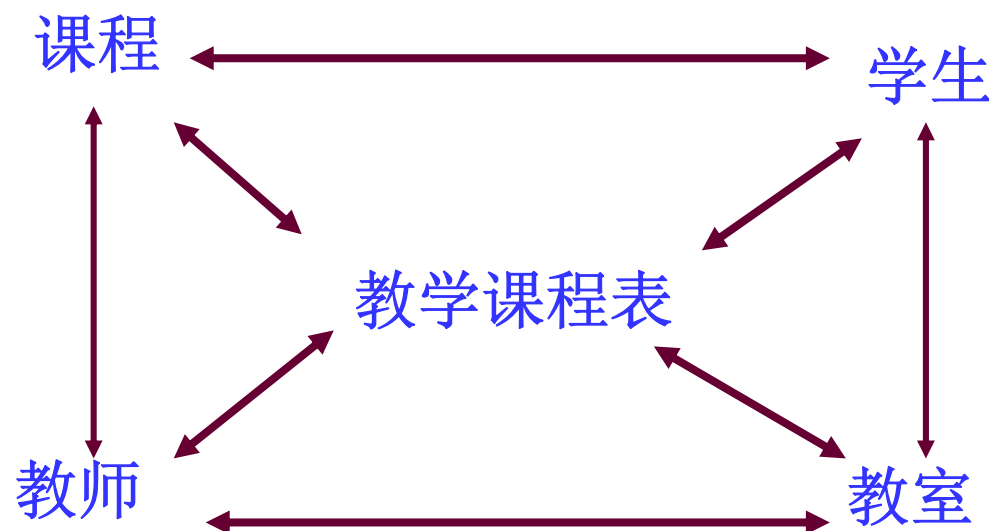
- 每个模型中只有一个称为根的最高结点
- 其它结点都只能和一个父结点相连接（1：M）
- 查询、访问都必须从根结点开始
- 最有影响的层次模型的DBS是60年代末，IBM公司推出的IMS层次模型数据库系统(晚于网状模型)。

层次模型示意图



网络模型

- (1) 允许一个以上的结点无双亲
 - (2) 一个结点可以有多于一个的双亲
- 例如，教务管理系统



最早出现的是网状 DBMS，是美国通用电气公司Bachman等人在1961年开发成功的IDS（Integrated DataStore）。

1969年10月，CODASYL (Conference on Data Systems Languages) 数据库研制者Bachman等提出了网络模型数据库系统规范报告DBTG，成为数据库历史上具有里程碑意义的文献(ACM)。

关系模型

商店关系

店 名	地 址	经办人	电话
解放路食品店	解放路262号	李国基	2-5036
桃园商场	桃园路6号	张山	6-6161
香香瓜果店	北大街26号	王宏	3-6201
白塔干鲜果店	西大街56号	宋良	3-3637
北大街果品店	北大街231号	林青	3-1116

关系模型

在关系模型中，数据的逻辑结构是一张二维表。下面我们以学生登记表为例，介绍关系模型中的主要术语。

- 关系 (Relation) 一个关系对应与我们平常讲的一张表。
- 元组 (Tuple) 表中的一行称为一个元组。
- 属性 (Attribute) 表中的一列称为属性，给每一列起一个名称即属性名。
- 主码 (Key) 表中的某个属性组，它们的值唯一的标识一个元组。
- 域 (Domain) 属性的取值范围。
- 分量 (Component) 元组中的一个属性值。
- 关系模式 (Relational Schema) 对关系的描述，用关系名 (属性名1, 属性名2, 属性名n) 来表示。

商店的数据关系模型

关系（库名）SHOP				属性（字段）
关系框架 或 库结构	店 名	地 址	经办人	电话
	解放路食品店	解放路262号	李国基	2-5036
	桃园商场	桃园路6号	张山	6-6161
	香香瓜果店	北大街26号	王宏	3-6201
	白塔干鲜果店	西大街56号	宋良	3-3637
	北大街果品店	北大街231号	林青	3-1116
元组 或 记录				属性 “电话” 的 值

关系模型优缺点

优点:

严格的数学概念

概念单一:关系

存取路径用户透明(数据独立性+安全性)

缺点:

技术)

路径透明造成查询效率不高(提出:segment数据存放,分区

数据库系统的构成

DBS = 数据库 + 硬件 + 软件 + 人员

- 硬件：
指运行DBS的硬件资源；
- 软件：指DBMS、操作系统，应用开发工具等；
- 人员：指开发、管理和使用数据库系统的人员。

关系数据库标准语言— SQL

SQL (Structured Query Language, 可读作Sequel) 语言是1974年由Boyce和Chamberlin提出的。由于它功能丰富、使用方式灵活、语言简洁易学等突出优点, 在计算机工业界和计算机用户中倍受欢迎并深深扎根。1986年10月, 美国国家标准局 (ANSI) 的数据库委员会批准了SQL作为关系数据库语言的美国标准。此后不久, 国际标准化组织 (ISO) 也做出了同样的决定。

此后ANSI不断修改和完善SQL标准, 并于1989年公布了SQL—89标准, 1992年又公布了SQL—92标准; 后面出现SQL—99 (SQL3), SQL-2003 (XML/window函数/merge语句 SQL4), SQL-2006和SQL-2008, SQL-2011。

SQL成为国际标准后, 数据库产品的各个厂家纷纷推出各自的支持SQL的软件或者与之接口的软件。这样就造成了一种极大的可能性, 不论是微机、小型机还是大型机, 不管是哪种数据库系统, 都采用SQL作为共同的数据存取语言 and 标准接口。

SQL标准发展的简要历史

- 1986年 ANSI X3.135-1986, ISO/IEC 9075:1986, SQL-86
- 1989年 ANSI X3.135-1989, ISO/IEC 9075:1989, SQL-89 120页
- 1992年 ANSI X3.135-1992, ISO/IEC 9075:1992, **SQL-92**(SQL2) 622 页
- 1999年 ISO/IEC 9075:1999, 1700页, SQL-1999(**SQL99**, SQL3)
- 2003年 ISO/IEC 9075:2003, 3600页, SQL-2003 SQL4 **merge + xml + window 函数**
- 2005年 SQL-2006 xml增强 xml+sql 3777页
- 2006年 SQL-2008 3777页
- 2010年 SQL 2011 Temporal databases(时态数据库,缓慢变化维)

SQL 概述

SQL的英文名称是结构化的查询语言。实际上它的功能包括

- 查询 (Query DML)
- 操纵 (Manipulation DML)
- 定义 (Definition DDL/DSDL)
- 控制 (Control DCL)

四个方面，是一个综合的、通用的、功能极强的关系数据库语言。

SQL动词

SQL功能	动词
数据查询	SELECT
数据操纵 DML	INSERT, UPDATE, DELETE
数据定义 DDL	CREATE, DROP, ALTER
数据控制 DCL	GRANT, REVOKE

扩充：Dump/Backup/Load/Restore/Export/IMP/EXP

模式的定义与删除

一、定义模式:

Create schema <模式名> authorization <用户名>

不指定模式名,与用户名一样

Ex: Create schema “S-T” authorization wang

Create database <数据库名> on filespec log on filespec

二、删除模式:

Drop schema <模式名> <cascade|restrict>

cascade:内部的表一并删除

restrict:存在表不允许删除模式

Ex: Drop schema zhang cascade

Drop database <数据库名>

建立、删除和修改基本表

一、定义基本表

一般格式：

```
CREATE TABLE <表名>  
(<列名> <数据类型> [列级完整性约束条件]  
[,<列名> <数据类型> [列级完整性约束条件]]...  
[,<表级完整性约束条件>]);
```

```
CREATE TABLE Student  
(sno CHAR(10) primary key,  
  Name CHAR(20) NOT NULL UNIQUE,  
  Sex CHAR(1), Age INT,  
  Department CHAR(20));
```



sno	Name	Sex	Age	Department

表级完整性约束条件举例

建立SC:

```
CREATE TABLE SC  
(  
    sno CHAR(10) not null,  
    cno CHAR(20) not null,  
    grade smallint null,  
    primary key (sno,cno), /*主码*/  
    foreign key(sno) references student(sno),  
    /*参照完整性*/  
    foreign key(cno) references course(cno)  
);
```

模式与表

一. 表名前指定模式名(数据库名)

Oracle: create table "S-T".student
sql server/ASE: create table S-T.dbo.student

二. 建模式同时建表

create schema test authorization zhang
create table tab1(col1 int);

三. 指定当前的模式或数据库

set search_path to "S-T",public (S-T不存在才用public)

sybase/ms sql:
use S-T

常用数据类型

ANSI-SQL:

int,integer	4bytes	smallint	2	bigint	8
decimal(15,2)		numeric(15,2)		float	
real		double precision			
char(20)		varchar(254)			
date		time			
clob	字符串大对象	blob	二进制大对象	interval	

扩展:

tinyint	bit	money
datetime (Timestamp)		smalldatetime
Blob	image	(long)binary
Text	

修改基本表

一般格式:

```
ALTER TABLE <表名>  
[ADD <新列名> <数据类型> [完整性约束]]  
[DROP <完整性约束名>]  
[Alter Column <列名> <数据类型>] /*oracle/ms sql*/  
[MODIFY <列名> <数据类型>];      /*sybase/ASA*/
```

例:

```
ALTER TABLE Student ADD SCome DATE;  
ALTER TABLE Student  
                DROP UNIQUE(sno);
```

删除基本表

一般格式:

```
DROP TABLE <表名> [restrict | cascade]
```

```
SQL99标准 [restrict | cascade]
```

例: DROP TABLE Student

**注意: 数据库允许删除方可, 不破坏数据库完整性等约束。
有些dbms删除基本表后,索引一定删除外,关联定义没有删除(不符合SQL99),如:完整性PK/View
restrict很多dbms不支持**

SQL查询语句

查询语句是SQL的核心语句,一般格式为:

```
SELECT [ ALL | DISTINCT ] <目标列表达式>[, <目标列表达式>]
  FROM <表名或view>[, <表名或view>] ...
[WHERE 条件表达式]
[GROUP BY 列名1] [HAVING <条件表达式>]
[ORDER BY 列名2 [ASC|DESC]];
```

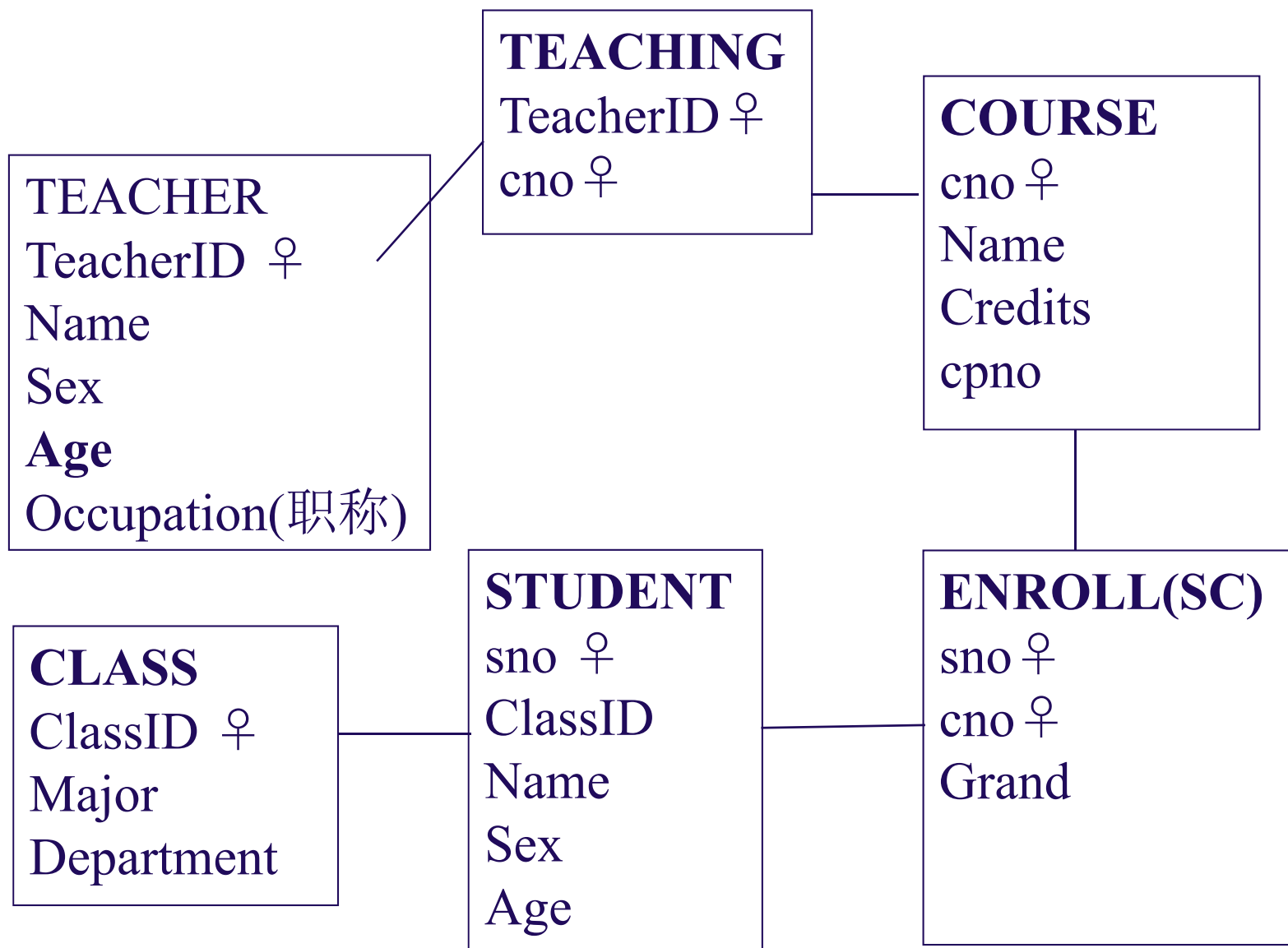
Sql server(支持OLAP 包括oracle)

```
SELECT [ ALL | DISTINCT ] select-list
...[ INTO { host-variable-list | variable-list } ]
...[ FROM table-list ]...[ WHERE search-condition ]
...[ GROUP BY column-name, ... ]...[ HAVING search-condition ]
...[ ORDER BY { expression | integer } [ ASC | DESC ], ... ]
```

SQL简单查询语句

从一个表中查询满足条件的所有元组，这可能是最基本而又常用的数据库操作。为完成上述操作，需要用到SQL的三个子句：SELECT、FROM和WHERE。

示例数据库



例1：列出所有男性教师的详细信息

TEACHER	
✓ TeacherID	♀
✓ Name	
✓ Sex	= '男'
✓ Age	
✓ Occupation	

```
SELECT * FROM TEACHER WHERE Sex = '男'
```

SQL中的投影操作

例1中的查询语句执行后，查询结果中将包含所有的字段。但在实际应用中，往往希望结果集中只包含感兴趣的字段。在SELECT子句中用具体的字段名代替*号即可达到目的。因此，SELECT子句对应于关系代数中的投影算子。

例2：列出所有男性教师的工号、姓名及职称

```
SELECT TeacherID, Name, Occupation  
FROM TEACHER WHERE Sex = '男';
```

集合函数

SQL提供了五个集合函数（aggregate functions），也称为内建函数（built-in functions），如COUNT、SUM、AVG、MAX和MIN等。这些函数大大地提高了SQL的统计计算能力。这些集合函数的功能简述如下(distinct 取消重复记录)：

COUNT	计算某一列中值的个数 count([distinct all])
COUNT(*)	计算记录总数
SUM	计算某一列中值的总和（该列必须是数值型）
AVG	计算某一列中值的均值（该列必须是数值型）
MAX	计算某一列值中的最大值
MIN	计算某一列值中的最小值

例[集合-1]: 统计教师中教授的人数、最大年龄、最小年龄及平均年龄。

```
SELECT COUNT(*) As Number,  
MAX(Age) As max_age,MIN(Age) As min_age,  
AVG(Age) As avg_age  
FROM TEACHER WHERE Occupation = '教授'
```

子查询(Sub query)

通过以上的学习，我们应该深深体会到这样的事实：选择查询语句的结果也是一个关系。这个关系可能为空，可能只包含一条元组，也可能包含成千上万条元组。

本质上，查询语句是针对关系进行操作，这个关系可能是可以永久存储的物理表，也可能是其它查询等结果。当一个查询基于另一个查询的结果时，就要用的子查询。

子查询也称为嵌套查询。子查询是指一个SELECT-FROM-WHERE查询块嵌入另一个查询块之中。SQL中允许多重嵌套。

例：请找出年龄最大的教师工号及其姓名

(请找出年龄最小的教师工号及其姓名)

请找出小于平均年龄的所有教师工号及其姓名)

```
SELECT TeacherID, Name From TEACHER  
where age = (Select Max(Age) From TEACHER)
```

(Max->min->avg)

例15：列出所有上课教师的工号及姓名。

```
SELECT TeacherID, Name  
FROM TEACHER  
WHERE TeacherID IN  
(SELECT TeacherID FROM TEACHING);
```

数据更新

插入数据

一、插入单个元组

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

INSERT INTO STUDENT
VALUES('200111', '小王' , '男' , 22);

INSERT INTO STUDENT
(sno, Name, Sex, Age)
VALUES('200111', '小王' , '男' , 22);

Getdate(),datalength(),sign(),abs()等

修改数据

```
UPDATE <表名>  
SET <列名> = <表达式>[,<列名> = <表达式>]...  
[WHERE <条件>];
```

一、修改单个元组

```
UPDATE Student SET Age = Age + 2  
WHERE sno = '9504101'
```

二、修改多个元组

```
UPDATE Student SET Age = Age + 1
```

三、关联修改

```
update student set age = student.age + 1 from class where  
student.classid=class.classid and class.department='CS'  
update (select age from student,class where  
student.classid=class.classid and class.department='CS')  
set age=age +1
```

删除数据

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

删除单个/多个元组

```
DELETE FROM Student WHERE sno = '9504101'
```

删除所有记录比较

```
DELETE FROM Student  
Truncate table student
```

关联删除

```
Delete student from class where  
student.classid=class.classid and class.department='CS'
```