

数据库系统原理

第七章 数据库设计:属性-联系方法

第七章 数据库设计:属性-联系方法

- 数据依赖
- 模式分解
- 范式
- 规范化
- 大数据与反规范化

函数依赖

设 $S(A)$ 是属性集 A 上的关系模式, $A_i, A_j \subseteq A$, t 是 $S(A)$ 上的任意一个关系, 如果

对 $\forall r_m, r_n \in t$, 若 $r_m[A_i] = r_n[A_i]$, 则 $r_m[A_j] = r_n[A_j]$ 成立。

那么称“ A_i 函数决定 A_j ”, 或“ A_j 函数依赖于 A_i ”, 记作 $A_i \rightarrow A_j$

称 A_i 为决定因素。

如果 A_j 不函数依赖于 A_i , 则记作 $A_i \nrightarrow A_j$ 。

examiner (erid,ername, ersex,erage,dname)

✓ 有函数依赖:

- $\text{erid} \rightarrow \text{ername}$
- $\text{erid} \rightarrow \text{ersex}$
- $\text{erid} \rightarrow \text{erage}$
- $\text{erid} \rightarrow \text{dname}$

✓ 但是, 没有函数依赖:

- $\text{ername} \rightarrow \text{dname}$

逻辑蕴涵

如果给定关系模式S的函数依赖集D，可以证明其它一些函数依赖也成立，就称这些函数依赖被D逻辑蕴涵。

闭包

如果给定关系模式S的函数依赖集D，被D所逻辑蕴涵的所有函数依赖的集合称为D的闭包，记为 D^+ 。

Armstrong公理

- 反射律:

若 $A_j \subseteq A_i$, 则 $A_i \rightarrow A_j$ 。

- 增广律:

若 $A_i \rightarrow A_j$, 则 $A_i A_k \rightarrow A_j A_k$ 。

- 传递律:

若 $A_i \rightarrow A_j$, $A_j \rightarrow A_k$, 则 $A_i \rightarrow A_k$ 。

Armstrong公理推论

- 合成规则
- 分解规则
- 伪传递规则

合成规则： 若 $A_i \rightarrow A_j$, $A_i \rightarrow A_k$, 则 $A_i \rightarrow A_j A_k$ 。

证明：

$$\left. \begin{array}{l} A_i \rightarrow A_j \Rightarrow A_i \rightarrow A_i A_j \\ A_i \rightarrow A_k \Rightarrow A_i A_j \rightarrow A_j A_k \end{array} \right\} \Rightarrow A_i \rightarrow A_j A_k$$

分解规则： 若 $A_i \rightarrow A_j A_k$ ， 则 $A_i \rightarrow A_j$ ， $A_i \rightarrow A_k$ 。

证明：

$$\left. \begin{array}{l} A_j \subseteq A_j A_k \Rightarrow A_j A_k \rightarrow A_j \\ A_i \rightarrow A_j A_k \end{array} \right\} \Rightarrow A_i \rightarrow A_j$$

同理： $A_i \rightarrow A_k$

伪传递规则：若 $A_i \rightarrow A_j$, $A_j A_1 \rightarrow A_k$, 则 $A_i A_1 \rightarrow A_k$ 。

证明：

$$\begin{array}{c} A_i \rightarrow A_j \Rightarrow A_i A_1 \rightarrow A_j A_1 \\ A_j A_1 \rightarrow A_k \end{array} \Bigg\} \Rightarrow A_i A_1 \rightarrow A_k$$

- 设D为属性集A上的一组函数依赖, $\alpha \subseteq A$,
 $\alpha_D^+ = \{ \beta \mid \alpha \rightarrow \beta \text{ 能由D根据Armstrong公理导出} \}$,
 α_D^+ 称为属性集 α 关于函数依赖集D的闭包。

算法 对 $S\langle A, D \rangle$ ，求属性集 M ($M \subseteq A, M = \{A_1, A_2, \dots, A_k\}$) 关于 A 上的函数依赖集 D 的闭包 M_D^+ 。

输入: A, M, D

输出: M_D^+

步骤:

(1) 令闭包为 N ，将 N 初始化: $N = \{A_1, A_2, \dots, A_k\}$ 。

(2) 在 D 中反复寻找这样的函数依赖: $B \rightarrow C$ 且 $B \subseteq N$ 但 $C \not\subseteq N$ ，若找到则

$N = C \cup N$ ，并重复这个过程。

(3) 当不能再添加任何属性时，集合 N 就是 M_D^+ 。

对 $S\langle A, D \rangle$, $A = \{a, b, c, g, h, i\}$, $D = \{a \rightarrow b, a \rightarrow c, cg \rightarrow h, cg \rightarrow i, b \rightarrow h\}$ 。

计算 $(ag)_D^+$

所用依赖

初始

$a \rightarrow b$

$a \rightarrow c$

$cg \rightarrow h$

$cg \rightarrow i$

$(ag)_D^+$

ag

agb

agbc

agbch

agbchi

$(ag)_D^+ = agbchi$

如果 $G^+=H^+$ ，则称H与G等价。

如果函数依赖集D满足下列条件，则称D为一个极小函数依赖集。亦称为最小依赖集。

- (1) D中任一函数依赖的右部仅含有一个属性。
- (2) D中不存在这样的函数依赖 $M \rightarrow N$ ，M有真子集Q使得 $D - \{M \rightarrow N\} \cup \{Q \rightarrow N\}$ 与D等价。
- (3) D中不存在这样的函数依赖 $M \rightarrow N$ ，使得 $D - \{M \rightarrow N\}$ 与D等价。

- (1) 逐一检查 D 中各函数依赖 $D_i: M \rightarrow N$, 若 $N = A_1 A_2 \dots A_k$, $k > 2$, 则用 $\{ M \rightarrow A_j | j=1, 2, \dots, k \}$ 来取代 $M \rightarrow N$ 。
- (2) 逐一取出 D 中各函数依赖 $D_i: M \rightarrow N$, 设 $M = B_1 B_2 \dots B_m$, 逐一考查 B_i ($i=1, 2, \dots, m$),
- 若 $N \in (M - B_i)_D^+$, 则以 $M - B_i$ 取代 M 。
- (3) 逐一检查 D 中各函数依赖 $D_i: M \rightarrow N$, 令 $G = D - \{M \rightarrow N\}$, 若 $N \in M_G^+$, 则从 D 中去掉此函数依赖。

$$D = \{a \rightarrow b, \cancel{b \rightarrow a}, b \rightarrow c, \cancel{a \rightarrow c}, c \rightarrow a\}$$

$$D_{m1} = \{a \rightarrow b, b \rightarrow c, c \rightarrow a\}$$

$$D = \{a \rightarrow b, \quad b \rightarrow a, \quad \cancel{b \rightarrow c}, \quad a \rightarrow c, \quad c \rightarrow a\}$$

$$D_{m2} = \{a \rightarrow b, \quad b \rightarrow a, \quad a \rightarrow c, \quad c \rightarrow a\}$$

一个函数依赖集的极小依赖集不唯一

一个函数依赖集与其极小依赖集以及其闭包都是等价的

- 在关系模式 $S(A)$ 中, 对于 A 的子集 A_i 和 A_j , 如果 $A_i \rightarrow A_j$, 但 $A_j \not\subseteq A_i$, 则称 $A_i \rightarrow A_j$ 是非平凡的函数依赖;
- 若 $A_i \rightarrow A_j$, 但 $A_j \subseteq A_i$, 则称 $A_i \rightarrow A_j$ 是平凡的函数依赖。
- 除非必要或特别声明, 我们总是只考虑非平凡的函数依赖。

第七章:数据库设计:属性-联系方法

- 数据依赖
- 模式分解
- 范式
- 规范化
- 大数据与反规范化

设关系模式 $S\langle A, D \rangle$ ，属性全集为 A ，函数依赖集为 D 。

而 $S_1\langle A_1, D_1 \rangle$ ， $S_2\langle A_2, D_2 \rangle$ ， \dots ， $S_n\langle A_n, D_n \rangle$ ，其中 $A=A_1A_2\dots A_n$ ，且不存在 $A_i\subseteq A_j$ ，

$D_i=\{\alpha\rightarrow\beta|\alpha\rightarrow\beta\in D^+\wedge\alpha\beta\subseteq A_i\}$ ($i,j=1,2,\dots,n$)，

对于 S 的任意关系 t ， $t_i=\pi_{S_i}(t)$ 。

关系模式 S_1 ， \dots ， S_n 的集合用 ρ 表示， $\rho=\{S_1, \dots, S_n\}$ 。

用 ρ 代替 S 的过程称为关系模式 S 的模式分解。

erdepa (erid, dname, dsum)

$D = \{\text{erid} \rightarrow \text{dname}, \text{dname} \rightarrow \text{dsum}\}$

erdepa

erid	dname	dsum
2009040	历史学院	100
1990122	教育学部	300
1998039	文学院	100
2007033	心理学部	300

idsum

erid	dsum
2009040	100
1990122	300
1998039	100
2007033	300

dnamesum

dname	dsum
历史学院	100
教育学部	300
文学院	100
心理学部	300

```
SELECT *  
FROM idsum NATURAL JOIN dnamesum;
```

erdepa

erid	dname	dsum
2009040	历史学院	100
1990122	教育学部	300
1998039	文学院	100
2007033	心理学部	300

idsum

erid	dsum
2009040	100
1990122	300
1998039	100
2007033	300

idname

erid	dname
2009040	历史学院
1990122	教育学部
1998039	文学院
2007033	心理学部

```
SELECT *  
FROM idsum NATURAL JOIN idname;
```

✓ 设 S 为关系模式，关系模式集 $\{S_1, S_2, \dots, S_n\}$ 为 S 的一个分解，即： $S = S_1 \cup S_2 \cup \dots \cup S_n$ 。令 t 是模式 S 上的关系，即 $t(S)$ ，而 $t_i = \Pi_{S_i}(t)$ ，于是总有：

$$\bullet t \subseteq t_1 \bowtie t_2 \bowtie \dots \bowtie t_n$$

✓ 如果对模式 S 上所有合法关系 t ，均有：

$$\bullet t = \Pi_{S_1}(t) \bowtie \Pi_{S_2}(t) \bowtie \dots \bowtie \Pi_{S_n}(t)$$

✓ 那么， $\{S_1, S_2, \dots, S_n\}$ 就是关系模式 S 的一个无损连接分解。

输入：关系模式 $S(A_1, A_2, \dots, A_n)$ ，它的函数依赖集 D ，以及分解 $\rho=\{S_1, S_2, \dots, S_k\}$

输出：确定 ρ 是否具有无损连接性

方法：

- (1) 构造一个 k 行 n 列的表，第 i 行对应于关系模式 S_i ，第 j 列对应于属性 A_j 。如果 $A_j \in S_i$ ，则 在第 i 行第 j 列上放符号 x_j ，否则放符号 y_{ij} 。
- (2) 重复考察 D 中的每一个函数依赖，并修改表中的元素。方法是这样的：取 D 中一个 $M \rightarrow A_j$ ，在 M 的分量中寻找相同的行，然后将这些行中 A_j 的分量改为相同的符号，如果 其中有 x_j ，则将 y_{ij} 改为 x_j ；若其中无 x_j ，则全部改为 y_{ij} （ i 是这些行的行号最小值）。
- (3) 如果发现表中某一行变成了 x_1, x_2, \dots, x_n ，则分解 ρ 具有无损连接性；如果 D 中所有函数依赖都不能再修改表中的内容，且没有发现这样的行，则分解 ρ 不具有无损连接性。

设 $S(a,b,c,d,e)$, 分解 $\rho=\{ad,ab,be,cde,ae\}$

$D=\{a \rightarrow c, b \rightarrow c, c \rightarrow d, de \rightarrow c, ce \rightarrow a\},$

① ② ③ ④ ⑤

判断 ρ 是否具无损连接性。

	a	b	c	d	e
ad	$\left\{ \begin{array}{l} x_1 \end{array} \right.$	y_{12}	$\left\{ \begin{array}{l} y_{13} \end{array} \right.$	x_4	y_{15}
ab	$\left\{ \begin{array}{l} x_1 \end{array} \right.$	$\left\{ \begin{array}{l} x_2 \end{array} \right.$	$\left\{ \begin{array}{l} y_{13} \end{array} \right.$	x_4	y_{25}
be	x_1	$\left\{ \begin{array}{l} x_2 \end{array} \right.$	$\left\{ \begin{array}{l} x_3 \end{array} \right.$	x_4	$\left\{ \begin{array}{l} x_5 \end{array} \right.$
cde	x_1	y_{42}	x_3	x_4	x_5
ae	$\blacklozenge x_1$	y_{52}	$\blacklozenge x_3$	x_4	x_5

定理：设 $\rho=(S_1, S_2)$ 是 S 的一个分解， D 是 S 上的函数依赖集，分解 ρ 具有无损连接性的充分必要条件是：

$$S_1 \cap S_2 \rightarrow (S_1 - S_2) \in D^+$$

或 $S_1 \cap S_2 \rightarrow (S_2 - S_1) \in D^+$

设关系模式 $S\langle A, D \rangle$ 被分解为 n 个关系模式

$$S_1\langle A_1, D_1 \rangle, S_2\langle A_2, D_2 \rangle, \dots, S_n\langle A_n, D_n \rangle$$

其中 $A = A_1 A_2 \dots A_n$, 且不存在 $A_i \subseteq A_j$, D_i 为 D 在 A_i 上投影, 若 D 等价 \cup

则称关系模式 S 的这个分解是保持函数依赖的。

erdepa(erid, dname, dsum)

$D_{\text{erdepa}} = \{\text{erid} \rightarrow \text{dname}, \text{dname} \rightarrow \text{dsum}\}$

idsum(erid, dsum)

$D_{\text{idsum}} = \{\text{erid} \rightarrow \text{dsum}\}$

dnamesum(dname, dsum)

$D_{\text{dnamesum}} = \{\text{dname} \rightarrow \text{dsum}\}$

$$(D_{\text{idsum}} \cup D_{\text{dnamesum}})^+ \neq D_{\text{erdepa}}^+$$

idname(erid, dname)

$D_{\text{idname}} = \{\text{erid} \rightarrow \text{dname}\}$

dnamesum(dname, dsum)

$D_{\text{dnamesum}} = \{\text{dname} \rightarrow \text{dsum}\}$

$$(D_{\text{idname}} \cup D_{\text{dnamesum}})^+ = D_{\text{erdepa}}^+,$$

模式的分解具有无损连接性和分解保持函数依赖是两个互相独立的标准。具有无损连接性的分解不一定能够保持函数依赖；同样，保持函数依赖的分解也不一定具有无损连接性。

- 不具有无损连接性，也未保持函数依赖。
- 保持函数依赖，但不具有无损连接性。
- 具有无损连接性，但未持函数依赖。
- 既具有无损连接性，又保持了函数依赖。

第七章:数据库设计:属性-联系方法

- 数据依赖
- 模式分解
- 范式
- 规范化
- 大数据与反规范化

- 属性之间往往亲疏不同

- 在S中, 如果 $A_i \rightarrow A_j$, 并且对于 A_i 的任何一个真子集 A_i' , 都没有 $A_i' \rightarrow A_j$, 则称 A_j 对 A_i 完全函数依赖, 记作 $A_i \xrightarrow{F} A_j$ 。

- $Erid \xrightarrow{F} ername$ 、 $erid \xrightarrow{F} ersex$

- 在S中, 如果 $A_i \rightarrow A_j$, 并且存在 A_i 的一个真子集 A_i' 有 $A_i' \rightarrow A_j$, 则称 A_j 对 A_i 部分函数依赖, 记作 $A_i \xrightarrow{P} A_j$ 。
- $(\text{erid}, \text{ername}) \xrightarrow{P} \text{erage}$ 、 $(\text{erid}, \text{ername}) \xrightarrow{P} \text{ersex}$

- 如果 $A_i \rightarrow A_j$, $A_j \rightarrow A_k$ 且 $A_j \nrightarrow A_i$, $A_j \not\subseteq A_i$, $A_k \not\subseteq A_j$, 那么称 $A_i \rightarrow A_k$ 是传递依赖, 或 A_k 传递依赖于 A_i , 记作 $A_i \xrightarrow{T} A_k$ 。

- 考官院系(考官号、考官院系名、考官院系办公地点)

- 考官号 \xrightarrow{T} 考官院系办公地点

- 如果 $A_i \rightarrow A_k$, 不存在 A_j 有 $A_j \rightarrow A_i$ 但 $A_i \rightarrow A_j$ 、 $A_j \not\subseteq A_i$ 、 $A_j \rightarrow A_k$ 、 $A_k \not\subseteq A_j$, 称 A_k 直接依赖于 A_i ,
记作 $A_i \xrightarrow{D} A_k$ 。

- 属性在关系模式中有不同的地位

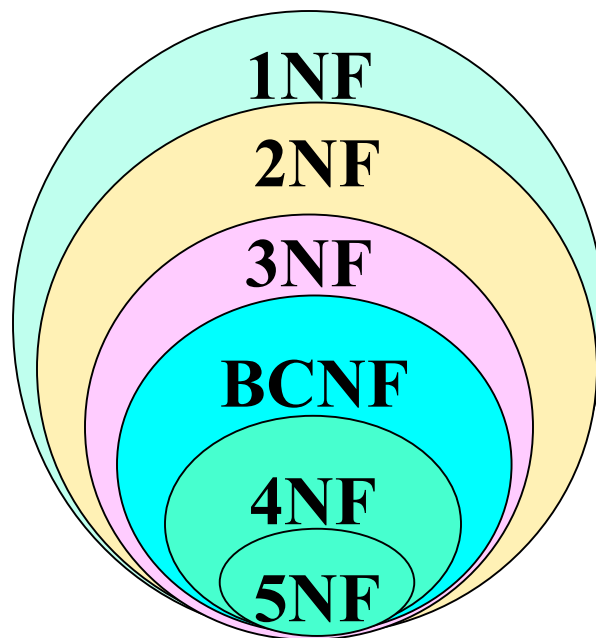
超键： 设K为S< A , D >的属性或属性组，若 $K \rightarrow A$ ，则称K为S的超键。

候选键： 设K为S< A , D >的超键，若 $K \xrightarrow{F} A$ ，则称K为S的候选键。

主键： 若S<A,D>有多个候选键，则可以从中选定一个作为S的主键。

- **主属性**：候选键中的各个属性，称作主属性
- **非主属性**：不包含在任何候选键中的各个属性称为非主属性或非键属性

- 范式
- $1NF \supset 2NF \supset 3NF \supset BCNF \supset 4NF \supset 5NF$



- 如果关系模式S的每个关系的每个属性值都是不可分的原子值，那么称S是第一范式的模式。

- 如果关系模式 $S \in 1NF$ ，且每一个非主属性都不部分依赖于 S 的任何候选键，则 $S \in 2NF$ 。
- 报考(报考号,试卷号,姓名,试卷名,分数)
- 报考号 \rightarrow 姓名 (报考号,试卷号) \rightarrow 姓名
- 报考 $\in 1NF$ 报考 $\notin 2NF$
- 考生(报考号,姓名)、试卷(试卷号,试卷名)、成绩(报考号,试卷号,分数)，都属于 $2NF$

- 如果关系模式S<A, D>是1NF, 且每个非主属性都既不部分也不传递依赖于S的任何候选键, 那么称S是第三范式(3NF)的模式。
- 考官院系(考官号,姓名,院系名,院系总人数)
- 考官号→院系名、院系名→院系总人数
- 考官号 \xrightarrow{T} 院系总人数
- 考官院系 \in 2NF 考官院系 \notin 3NF
- 考官(考官号,姓名,院系名)、院系(院系名,院系总人数), 都属于3NF

- 如果关系模式 $S\langle A, D \rangle$ 是第三范式，它的任何一个主属性都既不部分也不传递依赖于 S 的任何候选键，则称 $S \in \text{BCNF}$ 。
- 关系模式 $S\langle A, D \rangle$ ，它的任何一个(非主、主)属性都既不部分也不传递依赖于任何候选键，则称 $S \in \text{BCNF}$ 。
- 关系模式 $S\langle A, D \rangle \in 1\text{NF}$ ，其 D 中任意一个非平凡函数依赖的决定因素都包含键，则 $S \in \text{BCNF}$ 。

- 研究生导师(研究生号,导师号,院系名)
- 每个导师可以指导多名研究生但只能在一个院系工作; 每个研究生可以有几位导师, 但一个院系只能有一个
- (研究生号,导师号)→院系名、 导师号→院系名、 (研究生号,院系名)→导师号
- 候选键: (研究生号,导师号)、(研究生号,院系名)
- 导师号→院系名 (研究生号,导师号)→院系名
- 研究生导师∈3NF 研究生导师∉BC范式
- 导师院系(导师号, 院系名)、 师生(研究生号,导师号), 都属于BCNF

第七章:数据库设计:属性-联系方法

- 数据依赖
- 模式分解
- 范式
- 规范化
- 大数据与反规范化

- 规范化

我的账户

转账

贷款

缴费

信用卡

理财

家庭管家

得利宝

信托

基金超市

保险

黄金

外汇

国债证券期货

年金

网银设置

您当前所在位置：金融服务 -- 账户明细查询

说明

业务导航

明细查询

卡号/账号：6222 6208 1000 8714 269

日期：2016-12-01 至 2018-01-25

查询

高级查询

交易时间▲

打印

批量下载

批量记账

交易时间	交易方式	币种	支出金额	收入金额	余额	交易地点	操作
2016-12-20 00:00:00	存款利息	人民币	—	0.01	1.33	批处理	查看详情 导入账本
2018-01-23 11:40:35	代发其他款项	人民币	—	10.00	11.33	企业网银	查看详情 导入账本

保存文件格式：☒ excel ☐ txt

保存

页面编号:AC-10020100-02

- 若仅仅要求分解具有无损连接性，那么模式分解一定能够达到BCNF
- 若要求分解既具有无损连接性，又具有保持依赖性，则模式分解一定能够达到3NF，但不一定能够达到BCNF

- 达到BCNF无损连接分解算法：

给定关系模式 $S\langle A, D \rangle$,

① 令 $\rho = \{S\langle A, D \rangle\}$

② 检查 ρ 中各关系模式是否属于BCNF, 若是, 则算法终止; 否则, 继续执行③

③ 设 ρ 中 $S_i\langle A_i, D_i \rangle$ 不属于BCNF,

则必定存在非平凡函数依赖 $\alpha \rightarrow \beta \in D_i$, $\alpha \cap \beta = \emptyset$ 且 α 不包含 S_i 的候选键,

将 S_i 分解为 $\sigma = \{\alpha \cup \beta, A_i - \beta\}$,

以 σ 代替 S_i , 返回到②。

- 研究生导师(研究生号,导师号,院系名)
- (研究生号,导师号)→院系名、导师号→院系名 、(研究生号,院系名)→导师号
- 候选键为：(研究生号,导师号)、(研究生号,院系名)
- 导师院系(导师号,院系名)、师生(研究生号,导师号)

- 无损分解且保持依赖地分解成3NF的算法：

给定关系模式 $S\langle A, D \rangle$,

- ① 先求出D的极小依赖集，然后再把极小依赖集中那些左部相同的函数依赖用合并律合并起来
- ② 每个函数依赖 $\alpha \rightarrow \beta$ 去构成一个模式 $\alpha \sqcup \beta$
- ③ 在构成的模式集中，如果每个模式都不包含S的候选键，那么把候选键作为一个模式放入模式集中


- 考官院系(考官号,姓名,院系名,院系总人数)
- {考官号→姓名, 考官号→院系名, 院系名→院系总人数}
- {考官号→(姓名,院系名), 院系名→院系总人数}
- 考官(考官号,姓名,院系名)、院系(院系名,院系总人数)

- 报考(报考号,姓名,试卷号,试卷名)
- {报考号→姓名, 试卷号→试卷名}
- 考生(报考号,姓名)、试卷(试卷号,试卷名)、报考(报考号,试卷号)

第七章:数据库设计:属性-联系方法

- 数据依赖
- 模式分解
- 范式
- 规范化
- 大数据与反规范化

- 什么时候应该选择更高的范式、什么时候应该选择更低的范式呢？

- 
- 更新复杂。
 - 插入异常。
 - 删除异常。

eeid	eid	ename	erid
------	-----	-------	------

218811011013	0210000001	大学外语	2010019
--------------	------------	------	---------

---	0102000001	马克思主义 基本原理	2001002
-----	------------	---------------	---------

eeid	eid
218811011013	0205000002
218811011013	0210000001

eid	ename	erid
0205000002	中国近现代史 纲要	2009040
0210000001	大学外语	2010019
0102000001	马克思主义基本 原理	2001002

- 更新复杂无。
- 插入异常无。
- 删除异常无。



➤ 直接查询。

eeid	eid	ename	erid
------	-----	-------	------

218811011013	0210000001	大学外语	2010019
--------------	------------	------	---------

---	0102000001	马克思主义 基本原理	2001002
-----	------------	---------------	---------

eeid	eid
218811011013	0205000002
218811011013	0210000001
218811011116	0210000001

eid	ename	erid
0205000002	中国近现代史 纲要	2009040
0210000001	大学外语	2010019
0102000001	马克思主义基本 原理	2001002

➤先联接后查询。

- 规范化程度越高
 - 数据冗余越少，插入异常、删除异常、修改复杂等问题越少
 - 查询效率越低
- 规范化程度越低
 - 减少查询所要联接表的个数，减少了I/O和CPU时间，加速了查询
 - 插入异常、删除异常、修改复杂

• 写中心

- 证券交易、银行、售票系统等事务型应用：规范化，3NF或BCNF

序号	代码	名称	最新价	涨幅	换手率	成交量
1	600000	浦发银行	13.57	5.19%	1.35%	3796536
2	600001	邯郸钢铁				
3	600002	齐鲁退市				
4	600003	ST东北高				
5	600004	白云机场	16.75	1.58%	1.06%	219051

序号	代码	名称	最新价	涨幅	换手率	成交量
1	600000	浦发银行	13.57	5.19%	1.35%	3796536
2	600001	邯郸钢铁				
3	600002	齐鲁退市				
4	600003	ST东北高				
5	600004	白云机场	16.75	1.58%	1.06%	219051

• 读中心

- 数据分析型应用：低范式，非关系



- 借鉴利用关系模式分析方法来指导大数据非关系的模式设计

- 在设计数据库模式结构时，必须对现实世界的实际情况和用户应用需求作进一步分析，确定一个合适的冗余和规范化的折中处理。