



知识点74：关系规范化理论



关系数据库规范化理论

- 如何设计一个适合的关系数据库系统，关键是关系数据库模式的设计，一个好的关系数据库模式应该包括多少关系模式，而每一个关系模式又应该包括哪些属性，又如何将这些相互关联的关系模式组建成一个适合的关系模型，这些工作属于数据库设计的问题，确切地讲是数据库逻辑设计的问题



关系数据库规范化理论

- 关系数据库的规范化理论最早是由关系数据库的创始人E. F. Codd提出的,
- 后经许多专家学者对关系数据库理论作了深入的研究和发展, 形成了一整套有关关系数据库设计的理论。



规范化理论内容

- 关系数据库的规范化理论主要包括三个方面的内容：
 - 函数信赖
 - 范式 (Normal Form)
 - 模式设计
- 其中，**函数信赖**起着核心的作用，是模式分解和模式设计的基础，范式是模式分解的标准。



关系模式的存储异常问题

- 数据库的逻辑设计为什么要遵循一定的规范化理论？
- 什么是好的关系模式？
- 某些不好的关系模式可能导致哪些问题？

- **教学管理数据库**，其关系模式SCD如下：

- SCD (SNO, SN, AGE, DEPT, MN, CNO, SCORE)

- 其中，**SNO**表示学生学号，**SN**表示学生姓名，**AGE**表示学生年龄，**DEPT**表示学生所在的系别，**MN**表示系主任姓名，**CNO**表示课程号，**SCORE**表示成绩。

- 根据实际情况，这些数据有如下语义规定：

- 1. 一个系有若干个学生，但一个学生只属于一个系；
 - 2. 一个系只有一名系主任，但一个系主任可以同时兼几个系的系主任；
 - 3. 一个学生可以选修多门功课，每门课程可有若干学生选修；
 - 4. 每个学生学习课程有一个成绩。

在此关系模式中填入一部分具体的数据，则可得到**SCD**关系模式的实例，即一个教学管理数据库，如图所示。

SNO	SN	AGE	DEPT	MN	CNO	SCORE
S1	赵亦	17	计算机	刘伟	C1	90
S1	赵亦	17	计算机	刘伟	C2	85
S5	钱尔	18	信息	王平	C5	57
S2	钱尔	18	信息	王平	C6	80
S2	钱尔	18	信息	王平	C7	70
S2	钱尔	18	信息	王平	C5	70
S3	孙珊	20	信息	王平	C1	0
S3	孙珊	20	信息	王平	C2	70
S3	孙珊	20	信息	王平	C4	85
S4	李思	男	自动化	刘向	C1	93

关系SCD



关系模式的存储异常问题

- (SNO, CNO) 是该关系模式的主关系键。
- 在进行数据库的操作时，会出现以下几方面的问题。
 - 1. 数据冗余。每个系名和系主任的名字存储的次数等于该系的学生人数乘以每个学生选修的课程门数，同时学生的姓名、年龄也都要重复存储多次，数据的冗余度很大，浪费了存储空间。



关系模式的存储异常问题

- **插入异常**。如果某个新系没有招生，尚无学生时，则系名和系主任的信息无法插入到数据库中。
 - 因为在这个关系模式中，**(SNO,CNO)**是主关系键。根据关系的实体完整性约束，主关系键的值不能为空，而这时没有学生，SNO和CNO均无值，因此不能进行插入操作。
 - 另外，当某个学生尚未选课，即CNO未知，实体完整性约束还规定，主关系键的值不能部分为空，同样不能进行插入操作。



关系模式的存储异常问题

■ 3. 删除异常。

- 某系学生全部毕业而没有招生时，删除全部学生的记录则系名、系主任也随之删除，而这个系依然存在，在数据库中却无法找到该系的信息。
- 另外，如果某个学生不再选修C1课程，本应该只删去C1，但C1是主关系键的一部分，为保证实体完整性，必须将整个元组一起删掉，这样，有关该学生的其它信息也随之丢失。



关系模式的存储异常问题

■ 4. 更新异常。

- 如果学生改名，则该学生的所有记录都要逐一修改SN；
- 又如某系更换系主任，则属于该系的学生记录都要修改MN的内容，稍有不慎，就有可能漏改某些记录，这就会造成数据的不一致性，破坏了数据的完整性。



关系模式的存储异常问题

- SCD是一个不好的关系模式。
- 产生上述问题的原因，直观地说，是因为关系中“包罗万象”，内容太杂了。
- 那么，怎样才能得到一个好的关系模式呢？
 - 分解
- 我们把关系模式SCD分解为下面三个结构简单的关系模式，如图所示。
 - 学生关系S (SNO, SN, AGE, DEPT)
 - 选课关系SC (SNO, CNO, SCORE)
 - 系关系D (DEPT, MN)

S

SNO	SN	AGE	DEPT
S1	赵亦	17	计算机
S2	钱尔	18	信息
S3	孙珊	20	信息
S4	李思	21	自动化

D

DEPT	MN
计算机	刘伟
信息	王平
自动化	刘向

SC

SNO	CNO	SCORE
S1	C1	90
S1	C2	85
S2	C5	57
S2	C6	80
S2	C7	
S2	C5	70
S3	C1	0
S3	C2	70
S3	C4	85
S4	C1	93

分解后的关系模式

- 在以上三个关系模式中，实现了信息的某种程度的分离，
 - S中存储学生基本信息，与所选课程及系主任无关；
 - D中存储系的有关信息，与学生无关；
 - SC中存储学生选课的信息，而与学生及系的有关信息无关。
- 与SCD相比，分解为三个关系模式后，数据的冗余度明显降低。
 - 当新插入一个系时，只要在关系D中添加一条记录。
 - 当某个学生尚未选课，只要在关系S中添加一条学生记录，而与选课关系无关，这就避免了**插入异常**。
 - 当一个系的学生全部毕业时，只需在S中删除该系的全部学生记录，而关系D中有关该系的信息仍然保留，从而不会引起**删除异常**。
 - 同时，由于数据冗余度的降低，数据没有重复存储，也不会引起**更新异常**。



关系模式的存储异常问题

- 分解后的关系模式是一个好的关系数据库模式。
- 一个好的关系模式应该具备以下四个条件：
 - 1. 尽可能少的数据冗余。
 - 2. 没有插入异常。
 - 3. 没有删除异常。
 - 4. 没有更新异常。



关系模式的存储异常问题

- 注意：一个好的关系模式并不是在任何情况下都是最优的，
 - 比如查询某个学生选修课程名及所在系的系主任时，要通过连接，而连接所需要的系统开销非常大，因此要以实际设计的目标出发进行设计
- 如何按照一定的规范设计关系模式，将结构复杂的关系分解成结构简单的关系，降低或消除数据库中冗余数据的过程，这就是**关系的规范化**。



知识点75：函数依赖



函数依赖的定义及性质

- 在关系模式SCD中，SNO与SN、AGE、DEPT之间都有一种依赖关系。
- 由于一个SNO只对应一个学生，而一个学生只能属于一个系，所以当SNO的值确定之后，SN，AGE，DEPT的值也随之被唯一的确定了。
- 这类似于变量之间的**单值函数关系**。设单值函数 $Y=F(X)$ ，自变量 X 的值可以决定一个唯一的函数值 Y 。
- 在这里，我们说SNO决定函数（SN，AGE，DEPT），或者说（SN，AGE，DEPT）函数依赖于SNO。



函数依赖的形式化定义

定义5 设关系模式 $R(U, F)$ ， U 是属性全集， F 是 U 上的函数依赖集， X 和 Y 是 U 的子集，如果对于 $R(U)$ 的任意一个可能的关系 r ，对于 X 的每一个具体值， Y 都有唯一的具體值与之对应，则称 X 决定函数 Y ，或 Y 函数依赖于 X ，记作 $X \rightarrow Y$ 。我们称 X 为决定因素， Y 为依赖因素。当 Y 不函数依赖于 X 时，记作： $X \nrightarrow Y$ 。

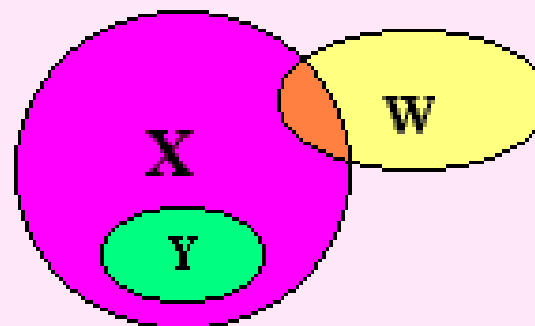
若 $X \rightarrow Y$ ，且 $Y \subseteq X$ ，则称 $X \rightarrow Y$ 是平凡的函数依赖

若 $X \rightarrow Y$ ，但 $Y \not\subseteq X$ ，则称 $X \rightarrow Y$ 是非平凡的函数依赖，若不特别声明，总是讨论非平凡的函数依赖

当 $X \rightarrow Y$ 且 $Y \rightarrow X$ 时，则记作： $X \leftrightarrow Y$



例： $X \rightarrow Y$ 为平凡函数依赖
 $X \rightarrow W$, $W \rightarrow Y$ 为非平凡函数依赖





- 例：在关系SC(Sno, Cno, Grade)中
- 非平凡函数依赖： $(Sno, Cno) \rightarrow Grade$
- 平凡函数依赖：
 - $(Sno, Cno) \rightarrow Sno$
 - $(Sno, Cno) \rightarrow Cno$
- 说明：对于任一关系模式，平凡函数依赖都是必然成立的，它不反映新的语义，因此若不特别声明，我们总是讨论非平凡函数依赖。



例如:

- 对于关系模式SCD

- $U = \{SNO, SN, AGE, DEPT, MN, CNO, SCORE\}$

- $F = \{SNO \rightarrow SN, SNO \rightarrow AGE, SNO \rightarrow DEPT\}$

- 一个SNO有多个SCORE的值与其对应，因此SCORE不能唯一地确定，即SCORE不能函数依赖于SNO，所以有： $SNO \not\rightarrow SCORE$ 。

- 但是SCORE可以被 (SNO, CNO) 唯一地确定。所以可表示为： $(SNO, CNO) \rightarrow SCORE$ 。



函数依赖（续）

例: Student(Sno, Sname, Ssex, Sage, Sdept)

假设不允许重名, 则有:

$Sno \rightarrow Ssex, \quad Sno \rightarrow Sage, \quad Sno \rightarrow Sdept,$

$Sno \longleftrightarrow Sname, \quad Sname \rightarrow Ssex, \quad Sname \rightarrow Sage$

$Sname \rightarrow Sdept$

但 $Ssex \not\rightarrow Sage$

若 $X \rightarrow Y$, 并且 $Y \rightarrow X$, 则记为 $X \longleftrightarrow Y$ 。

若 Y 不函数依赖于 X , 则记为 $X \not\rightarrow Y$ 。



有关函数依赖的说明:

1. 函数依赖是语义范畴的概念。

- 只能根据语义来确定一个函数依赖，而不能按照其形式化定义来证明一个函数依赖是否成立。
- 例如，对于关系模式S，当学生不存在重名的情况下，可以得到：
 - $SN \rightarrow AGE$
 - $SN \rightarrow DEPT$
- 这种函数依赖关系，必须是在没有重名的学生条件下才成立的，否则就不存在函数依赖了。
- 所以函数依赖反映了一种语义完整性约束。



有关函数依赖的说明

2. 函数依赖与属性之间的联系类型有关。

- (1) 在一个关系模式中，如果属性X与Y有1:1联系时，则存在函数依赖 $X \rightarrow Y$ ， $Y \rightarrow X$ ，即 $X \leftrightarrow Y$ 。
■ 例如，当学生无重名时， $SNO \leftrightarrow SN$ 。
- (2) 如果属性X与Y有m:1的联系时，则只存在函数依赖 $X \rightarrow Y$ 。
■ 例如，SNO与AGE，DEPT之间均为m:1联系，所以有 $SNO \rightarrow AGE$ ， $SNO \rightarrow DEPT$ 。
- (3) 如果属性X与Y有m: n的联系时，则X与Y之间不存在任何函数依赖关系。
■ 例如，一个学生可以选修多门课程，一门课程又可以为多个学生选修，所以SNO与CNO之间不存在函数依赖关系
- 可从属性间的联系类型来分析属性间的函数依赖



有关函数依赖的说明:

3. 函数依赖关系的存在与时间无关。

- 因为函数依赖是指关系中的所有元组应该满足的约束条件，而不是指关系中某个或某些元组所满足的约束条件。
- 当关系中的元组增加、删除或更新后都不能破坏这种函数依赖。
- 因此，必须根据语义来确定属性之间的函数依赖，而不能单凭某一时刻关系中的实际数据值来判断。
 - 例如，对于关系模式**S**，假设没有给出无重名的学生这种语义规定，则即使当前关系中没有重名的记录，也只能存在函数依赖**SNO→SN**，而不能存在函数依赖**SN→SNO**，因为如果新增加一个重名的学生，函数依赖**SN→SNO**必然不成立。



知识点76：函数依赖类型



函数依赖类型1

- 定义6.2 在 $R(U)$ 中，如果 $X \rightarrow Y$ ，并且对于 X 的任何一个真子集 X' ，都有 $X' \not\rightarrow Y$ ，则称 Y 对 X 完全函数依赖，记作 $X \xrightarrow{f} Y$ 。

若 $X \rightarrow Y$ ，但 Y 不完全函数依赖于 X ，则称 Y 对 X 部分函数依赖，记作 $X \xrightarrow{p} Y$ 。



SNO	SN	AGE	DEPT	MN	CNO	SCORE
S1	赵亦	17	计算机	刘伟	C1	90
S1	赵亦	17	计算机	刘伟	C2	85
S5	钱尔	18	信息	王平	C5	57
S2	钱尔	18	信息	王平	C6	80
S2	钱尔	18	信息	王平	C7	70
S2	钱尔	18	信息	王平	C5	70
S3	孙珊	20	信息	王平	C1	0
S3	孙珊	20	信息	王平	C2	70
S3	孙珊	20	信息	王平	C4	85
S4	李思	男	自动化	刘向	C1	93



- 在一个关系模式中，当存在非主属性对候选键的部分依赖时，就会产生数据冗余和更新异常。若非主属性对候选键完全函数依赖，不会出现以上问题。



■ 由上可知：

- 只有当决定因素是组合属性时，讨论部分函数依赖才有意义
- 当决定因素是单属性时，只能是完全函数依赖。
 - 例如，在关系模式S (SNO, SN, AGE, DEPT)，决定因素为单属性SNO，有 $SNO \rightarrow (SN, AGE, DEPT)$ ，不存在部分函数依赖。



函数依赖类型2

- **传递函数依赖**：在 $R(U)$ 中，如果 $X \rightarrow Y$ ， $Y \rightarrow Z$ ，且 $Y \not\subseteq X$ ， $Y \not\rightarrow X$ ，则称 Z 对 X 传递函数依赖。
- 如果 $Y \rightarrow X$ ，则 $X \leftrightarrow Y$ ，这时称 Z 对 X 直接函数依赖，而不是传递函数依赖。
- 例如，在关系模式SCD中， $SNO \rightarrow DEPT$ ，但 $DEPT \not\rightarrow SNO$ ，而 $DEPT \rightarrow MN$ ，则有 $SNO \xrightarrow{t} MN$ 。
- 当学生不存在重名的情况下，有 $SNO \rightarrow SN$ ， $SN \rightarrow SNO$ ， $SNO \leftrightarrow SN$ ， $SNO \rightarrow DEPT$ ，则 $SN \rightarrow DEPT$ ，这时 $DEPT$ 对 SN 是直接函数依赖，而不是传递函数依赖。



函数依赖类型

- 综上所述，函数依赖分为**完全函数依赖**、**部分函数依赖**和**传递函数依赖**三类，它们是规范化理论的依据和规范化程度的准则
- 从函数依赖的角度看，关系模式中存在各属性（包括非主属性和主属性）对候选键的部分依赖和传递依赖是产生数据冗余和更新异常的根源。



知识点77：范式



范式

- 规范化的**基本思想**是消除关系模式中的数据冗余，消除数据依赖中的不合适的部分，解决数据插入、删除时发生异常现象。
- 这就要求关系数据库设计出来的关系模式要满足一定的条件。
- 关系数据库的规范化过程中为不同程度的规范化要求设立的不同标准称为**范式**（Normal Form）。
- 由于规范化的程度不同，就产生了**不同的范式**。
- 满足最基本规范化要求的模式叫**第一范式**，
- 在第一范式中进一步满足一些要求为**第二范式**，
- 以此类推就产生了**第三范式**等概念。

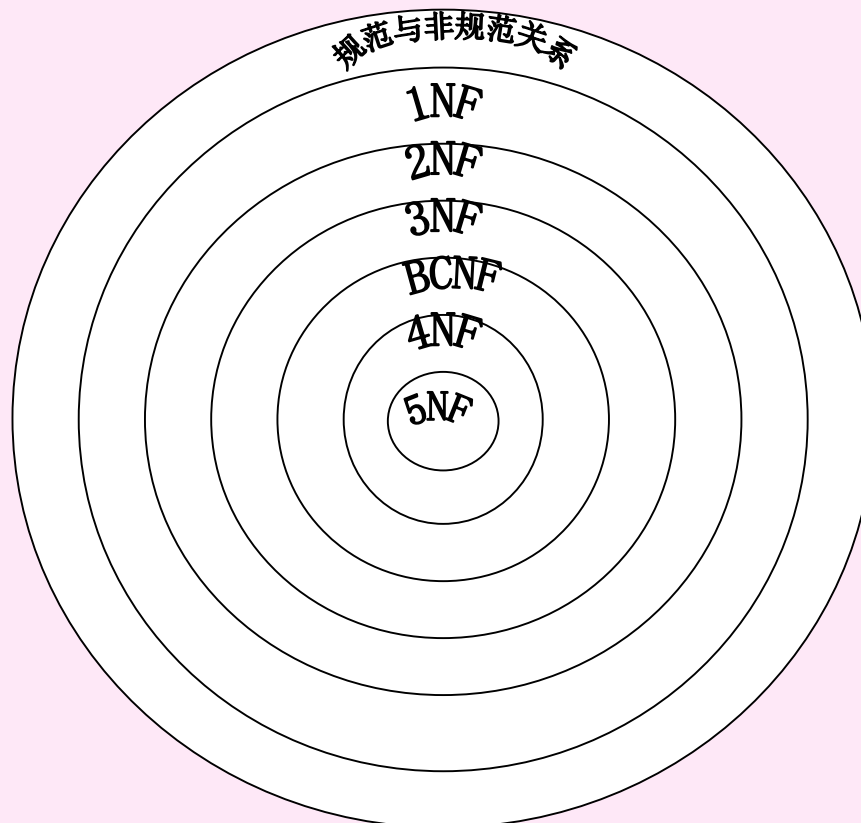


- 范式的概念最早由E. F. Codd提出。
- 从1971年起，Codd相继提出了关系的三级规范化形式，即第一范式（1NF）、第二范式（2NF）、第三范式（3NF）。
- 1974年，Codd和Boyce以共同提出了一个新的范式的概念，即Boyce-Codd范式，简称BC范式。
- 1976年Fagin提出了第四范式，后来又有人定义了第五范式。
- 至此在关系数据库规范中建立了一个范式系列：1NF, 2NF, 3NF, BCNF, 4NF, 5NF, 一级比一级有更严格的要求。



范式之间的关系

- 各个范式之间的联系可以表示为：
 - $5NF \subset 4NF \subset BCNF \subset 3NF \subset 2NF \subset 1NF$





- 某一关系模式R为第n范式，可简记为 $R \in nNF$ 。
- 一个低一级范式的关系模式，通过模式分解可以转换为若干个高一级范式的关系模式的集合，这种过程就叫规范化

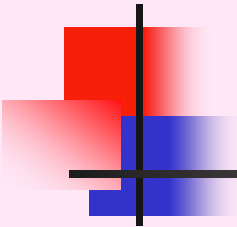


第一范式

- 第一范式（**First Normal Form**）是最基本的规范形式，即关系中每个属性都是不可再分的简单项。
- **定义8** 如果关系模式R，其所有的属性均为简单属性，即每个属性是不可再分的，则称R属于第一范式，简称**1NF**，记作 $R \in 1NF$ 。
- 在非规范化的关系中去掉组合项就能化成规范化的关系。**每个规范化的关系都属于1NF**，这也是它之所以称为“第一”的原因。



- 第一范式是对关系模式的最起码的要求。不满足第一范式的数据库模式不能称为关系数据库
- 但是满足第一范式的关系模式并不一定是一个好的关系模式



职工号	姓名	工资		
		基本工资	职务工资	工龄工资

职工号	姓名	基本工资	职务工资	工龄工资

- SCD (SNO, SN, AGE, DEPT, MN, CNO, SCORE)
- 关系模式SCD属于第一范式，但其具有大量的数据冗余，具有插入异常、删除异常、更新异常等弊端。
- 为什么会存在这种问题呢？
- 让我们分析一下SCD中的函数依赖关系，它的主关系键是 (SNO, CNO) 的属性组合，所以有：
 - $(SNO, CNO) \xrightarrow{f} SCORE$
 - $SNO \rightarrow SN, (SNO, CNO) \xrightarrow{p} SN$
 - $SNO \rightarrow AGE, (SNO, CNO) \xrightarrow{p} AGE$
 - $SNO \rightarrow DEPT, (SNO, CNO) \xrightarrow{p} DEPT$
 - $SNO \longrightarrow MN, (SNO, CNO) \xrightarrow{p} MN$



- 在SCD中，既存在完全函数依赖，又存在部分函数依赖和传递函数依赖。这种情况往往在数据库中是不允许的，也正是由于关系中存在着复杂的函数依赖，才导致数据操作中出现了种种弊端。



第二范式

- **定义9** 如果关系模式 $R \in 1NF$ ，且每个非主属性都完全函数依赖于 R 的每个关系键，则称 R 属于**第二范式**（**Second Normal Form**），简称**2NF**，记作 $R \in 2NF$ 。
- 在关系模式SCD中， SNO ， CNO 为主属性， AGE ， $DEPT$ ， MN ， MN ， $SCORE$ 均为非主属性，经上述分析，存在非主属性对关系键的部分函数依赖，所以 $SCD \notin 2NF$ 。

S

SNO	SN	AGE	DEPT
S1	赵亦	17	计算机
S2	钱尔	18	信息
S3	孙珊	20	信息
S4	李思	21	自动化

D

DEPT	MN
计算机	刘伟
信息	王平
自动化	刘向

SC

SNO	CNO	SCORE
S1	C1	90
S1	C2	85
S2	C5	57
S2	C6	80
S2	C7	
S2	C5	70
S3	C1	0
S3	C2	70
S3	C4	85
S4	C1	93

- S的关系键为SNO，D的关系键为DEPT，都是单属性，不可能存在部分函数依赖。
- 而对于SC， $(SNO, CNO) \xrightarrow{f} SCORE$ 。所以SCD分解后，消除了非主属性对关系键的部分函数依赖，S，D，SC均属于2NF。



第二范式

- 关系模式TCS (T, C, S) ,
 - 一个教师可以讲授多门课程，一门课程可以为多个教师讲授，
 - 同样一个学生可以选听多门课程，一门课程可以为多个学生选听，
 - (T, C, S)三个属性的组合是关系键，T, C, S都是主属性，而无非主属性，所以也就不可能存在非主属性对关系键的部分函数依赖， $TCS \in 2NF$ 。



第二范式

■ 结论:

- 1. 从1NF关系中消除非主属性对关系键的部分函数依赖, 则可得到2NF关系。
- 2. 如果R的关系键为单属性, 或R的全体属性均为主属性, 则 $R \in 2NF$ 。



2NF规范化

- 2NF规范化是指把1NF关系模式通过投影分解转换成2NF关系模式的集合。
- 分解时遵循的基本原则就是“**一事一地**”，让一个关系只描述一个实体或者实体间的联系。如果多于一个实体或联系，则进行投影分解。
- 下面以关系模式SCD为例，来说明2NF规范化的过程

例 将SCD (SNO, SN, AGE, DEPT, MN, CNO, SCORE) 规范到2NF。

- 由 $SNO \rightarrow SN$, $SNO \rightarrow AGE$, $SNO \rightarrow DEPT$, $(SNO, CNO) \xrightarrow{f} SCORE$, 可以判断，关系SCD至少描述了两个实体，
- 一个为学生实体，属性有SNO、SN、AGE、DEPT、MN；
- 另一个是学生与课程的联系（选课），属性有SNO、CNO和SCORE。



2NF规范化

- 2NF规范化是指把1NF关系模式通过投影分解转换成2NF关系模式的集合。
- 分解时遵循的基本原则就是“**一事一地**”，让一个关系只描述一个实体或者实体间的联系。如果多于一个实体或联系，则进行投影分解。



2NF规范化

- 下面以关系模式SCD为例，来说明2NF规范化的过程

例 将SCD (SNO, SN, AGE, DEPT, MN, CNO, SCORE) 规范到2NF。

- 由 $SNO \rightarrow SN$, $SNO \rightarrow AGE$, $SNO \rightarrow DEPT$, $(SNO, CNO) \xrightarrow{f} SCORE$, 可以判断，关系SCD至少描述了两个实体，
- 一个为学生实体，属性有SNO、SN、AGE、DEPT、MN；
- 另一个是学生与课程的联系（选课），属性有SNO、CNO和SCORE。

- SD (SNO, SN, AGE, DEPT, MN), 描述学生实体;
- SC (SNO, CNO, SCORE), 描述学生与课程的联系。

■ SD

SNO	SN	AGE	DEPT	MN
S1	赵亦	17	计算机	刘伟
S2	钱尔	18	信息	王平
S3	孙珊	20	信息	王平
S4	李思	21	自动化	刘向

SC

SNO	CNO	SCORE
S1	C1	90
S1	C2	85
S2	C5	57
S2	C6	80
S2	C7	
S2	C5	70
S3	C1	0
S3	C2	70
S3	C4	85
S4	C1	93

关系SD和SC

- 分解后的两个关系SD和SC，主键分别为SNO和（SNO，CNO），非主属性对主键完全函数依赖。因此， $SD \in 2NF$ ， $SC \in 2NF$
- 分解后，SD和SC的函数依赖分别如图1和2所示。

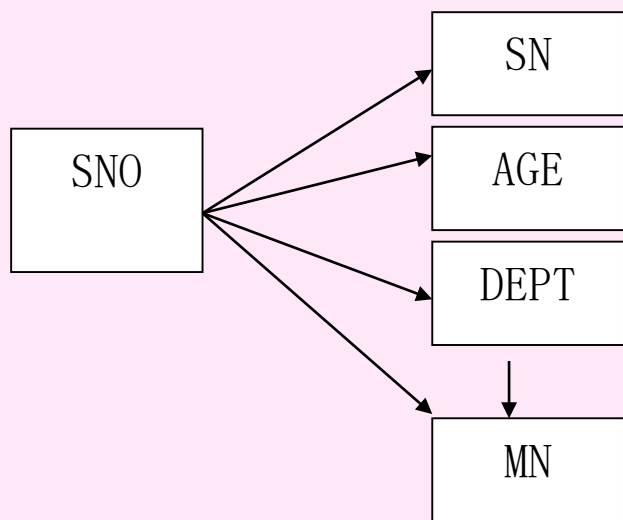


图1 SD中的函数依赖关系

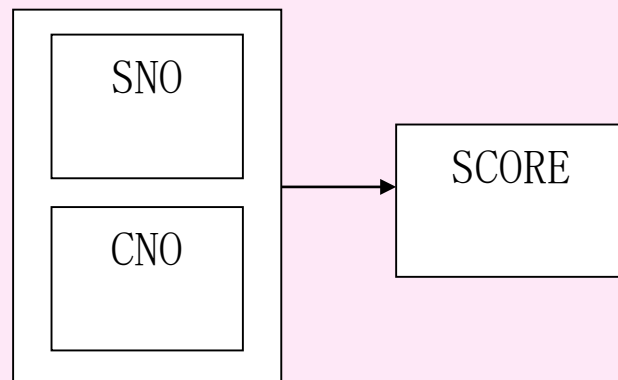


图2 SC中的函数依赖关系

2NF的缺点

■ 2NF的关系模式在进行数据操作时，仍然存在着一些问题：

- 1. 数据冗余。每个系名和系主任的名字存储的次数等于该系的学生人数。
- 2. 插入异常。当一个新系没有招生时，有关该系的信息无法插入。
- 3. 删除异常。某系学生全部毕业而没有招生时，删除全部学生的记录也随之删除了该系的有关信息。
- 4. 更新异常。更换系主任时，仍需改动较多的学生记录。

■ 之所以存在这些问题，是由于在SCD中存在着非主属性对主键的传递依赖。

■ 分析SCD中的函数依赖关系， $SNO \rightarrow SN$ ， $SNO \rightarrow AGE$ ， $SNO \rightarrow DEPT$ ， $DEPT \rightarrow MN$ ， $SNO \xrightarrow{t} MN$ ，非主属性MN对主键SNO传递依赖。



第三范式

定义10 如果关系模式 $R \in 2NF$ ，且每个非主属性都不传递依赖于 R 的每个关系键，则称 R 属于第三范式（Third Normal Form），简称3NF，记作 $R \in 3NF$ 。

■ 第三范式具有如下性质：

1. 如果 $R \in 3NF$ ，则 R 也是2NF。
2. 如果 $R \in 2NF$ ，则 R 不一定是3NF。
 - 例如，我们前面由关系模式SCD分解而得到的SD和SC都为2NF，其中， $SC \in 3NF$ ，但在SD中存在着非主属性MN对主键SNO传递依赖， $SD \notin 3NF$ 。对于SD，应该进一步进行分解，使其转换成3NF。



- 判断下列模式分别属于哪个范式(最高范式)并说明理由
 - $R(\{S\#, SD, SL, SN\}, \{S\# \rightarrow SD, S\# \rightarrow SN, SD \rightarrow SL\})$



3NF规范化

- **3NF规范化**是指把2NF关系模式通过投影分解转换成3NF关系模式的集合。
- 和2NF的规范化时遵循的原则相同，即“**一事一地**”，让一个关系只描述一个实体或者实体间的联系。
- **3NF规范化是指把2NF关系模式通过投影分解转换成3NF关系模式的集合，因此要消除非主属性对候选键的传递依赖。分解的方法很简单，即以构成传递链的两个基本依赖为基础形成两个新的模式。**



- 将SD (SNO, SN, AGE, DEPT, MN) 规范到3NF。
 - 分析SD的属性组成，可以判断，关系SD实际上描述了两个实体：
 - 一个为学生实体，属性有SNO, SN, AGE, DEPT;
 - 另一个是系的实体，其属性DEPT和MN。

S

SNO	SN	AGE	DEPT
S1	赵亦	17	计算机
S2	钱尔	18	信息
S3	孙珊	20	信息
S4	李思	21	自动化

D

DEPT	MN
计算机	刘伟
信息	王平
自动化	刘向

关系S和D

对于分解后的两个关系S和D，主键分别为SNO和DEPT，不存在非主属性对主键的传递函数依赖。因此， $S \in 3NF$ ， $D \in 3NF$ 。



关系模式的规范化

- 一个低一级范式的关系模式，通过模式分解转化为若干个高一级范式的关系模式的集合，这种分解过程叫作关系模式的**规范化**（Normalization）。

关系模式规范化的目的和原则

- 一个关系只要其分量都是不可分的数据项，就可称作规范化的关系，但这只是最基本的规范化。
- 这样的关系模式是合法的。
- 但人们发现有些关系模式存在插入、删除、修改异常、数据冗余等弊病。
- 规范化的目的就是使结构合理，消除存储异常，使数据冗余尽量小，便于插入、删除和更新。



规范化的基本原则

- 规范化的**基本原则**就是遵从概念单一化“一事一地”的原则，即一个关系只描述一个实体或者实体间的联系。
 - 若多于一个实体，就把它“分离”出来。
 - 因此，所谓规范化，实质上是概念的单一化，即一个关系表示一个实体。
- SCD (SNO, SN, AGE, DEPT, MN, CNO, SCORE)
 - 学生关系S (SNO, SN, AGE, DEPT)
 - 选课关系SC (SNO, CNO, SCORE)
 - 系关系D (DEPT, MN)



关系模式规范化的步骤

- 规范化就是对原关系进行投影，消除决定属性不是候选键的任何函数依赖。具体可以分为以下几步：
 - 1. 对1NF关系进行投影，消除原关系中非主属性对键的**部分函数依赖**，将1NF关系转换成若干个2NF关系。
 - 2. 对2NF关系进行投影，消除原关系中非主属性对键的**传递函数依赖**，将2NF关系转换成若干个3NF关系。
 - 3. 对3NF关系进行投影，消除原关系中**主属性对键的部分函数依赖和传递函数依赖**，也就是说使决定因素都包含一个候选键。得到一组BCNF关系。



消除决定属性
不是候选键的
非平凡的函数
依赖

1NF

2NF

3NF

BCNF

消除非主属性对键的部分函数依赖

消除非主属性对键的传递函数依赖

消除主属性对键的部分和传递函数依赖