作业_模式分解-数据库设计_2023211603_王何佳

姓名	学号	班级	专业
王何佳	2023211603	2023211804	网络空间安全

Teaching(Course, Teacher, Time, Room, Student, Grade)

函数依赖

- Course->Teacher
- Time, Room -> Course
- Time, Teacher -> Room
- Time, Student -> Room
- Course, Student -> Grade

1、是否是BCNF或3NF？

2、BCNF分解

3、分解后的模式集是否保持函数依赖？

1.是否是BCNF或3NF?

该关系模式既不属于 3NF，也不属于 BCNF。
唯一候选码是 `{Time, Student}`
函数依赖 `Course -> Teacher` 违反了BCNF（因为`Course`不是超码），也违反了3NF（因为`Course`不是超码，且`Teacher`不是主属性）

2.BCNF分解

分解后的关系模式集：

- `R1(Course, Teacher)` (主键: `{Course}`)
- `R3(Course, Student, Grade)` (主键: `{Course, Student}`)
- `R5(Time, Room, Course)` (主键: `{Time, Room}`)
- `R6(Time, Student, Room)` (主键: `{Time, Student}`)

分解过程如下：

1. **初始关系**: `R(Course, Teacher, Time, Room, Student, Grade)`
   - 违反 BCNF 的依赖包括 `Course -> Teacher`, `Time, Room -> Course`, `Time, Teacher -> Room`, `Course, Student -> Grade`。

2. **选择一个违反依赖进行分解**，例如 `Course -> Teacher`。
   - 分解为：
     - `R1(Course, Teacher)`: 这个关系满足 `Course -> Teacher`， `Course` 是其主键，符合 BCNF。
     - `R2(Course, Time, Room, Student, Grade)`: 剩下的属性，并保留决定因素 `Course`。
3. **检查新关系** `R2`。 `R2` 的候选码仍然是 `{Time, Student}`。
   - `R2` 中的函数依赖有 `Time, Room -> Course`, `Course, Student -> Grade`, `Time, Student -> Room`。
   - 其中 `Time, Room -> Course` 和 `Course, Student -> Grade` 仍然违反 BCNF。
4. **继续分解** `R2`，例如使用 `Course, Student -> Grade`。
   - `R2` 分解为：
     - `R3(Course, Student, Grade)`: 这个关系满足 `Course, Student -> Grade`， `{Course, Student}` 是其主键，符合 BCNF。
     - `R4(Course, Time, Room, Student)`: 剩下的属性，并保留决定因素 `{Course, Student}`。
5. **检查新关系** `R4`。 `R4` 的候选码是 `{Time, Student}`。
   - `R4` 中的函数依赖有 `Time, Room -> Course` 和 `Time, Student -> Room`。
   - `Time, Room -> Course` 仍然违反 BCNF (因为 `{Time, Room}` 不是 `R4` 的超码)。
6. **最后分解** `R4`，使用 `Time, Room -> Course`。
   - `R4` 分解为：
     - `R5(Time, Room, Course)`: 满足 `Time, Room -> Course`， `{Time, Room}` 是其主键，符合 BCNF。
     - `R6(Time, Student, Room)`: 剩下的属性，并保留决定因素 `{Time, Room}`。注意， `Time, Student -> Room` 这个依赖的所有属性都在这个关系中，其决定因素 `{Time, Student}` 是该关系的主键，所以它也符合 BCNF。

3.分解后的模式集是否保持函数依赖?

不保持。原始的函数依赖 `Time, Teacher -> Room` 丢失了，因为它的所有属性（`Time`, `Teacher`, `Room`）没有出现在分解后的任何一个关系中。

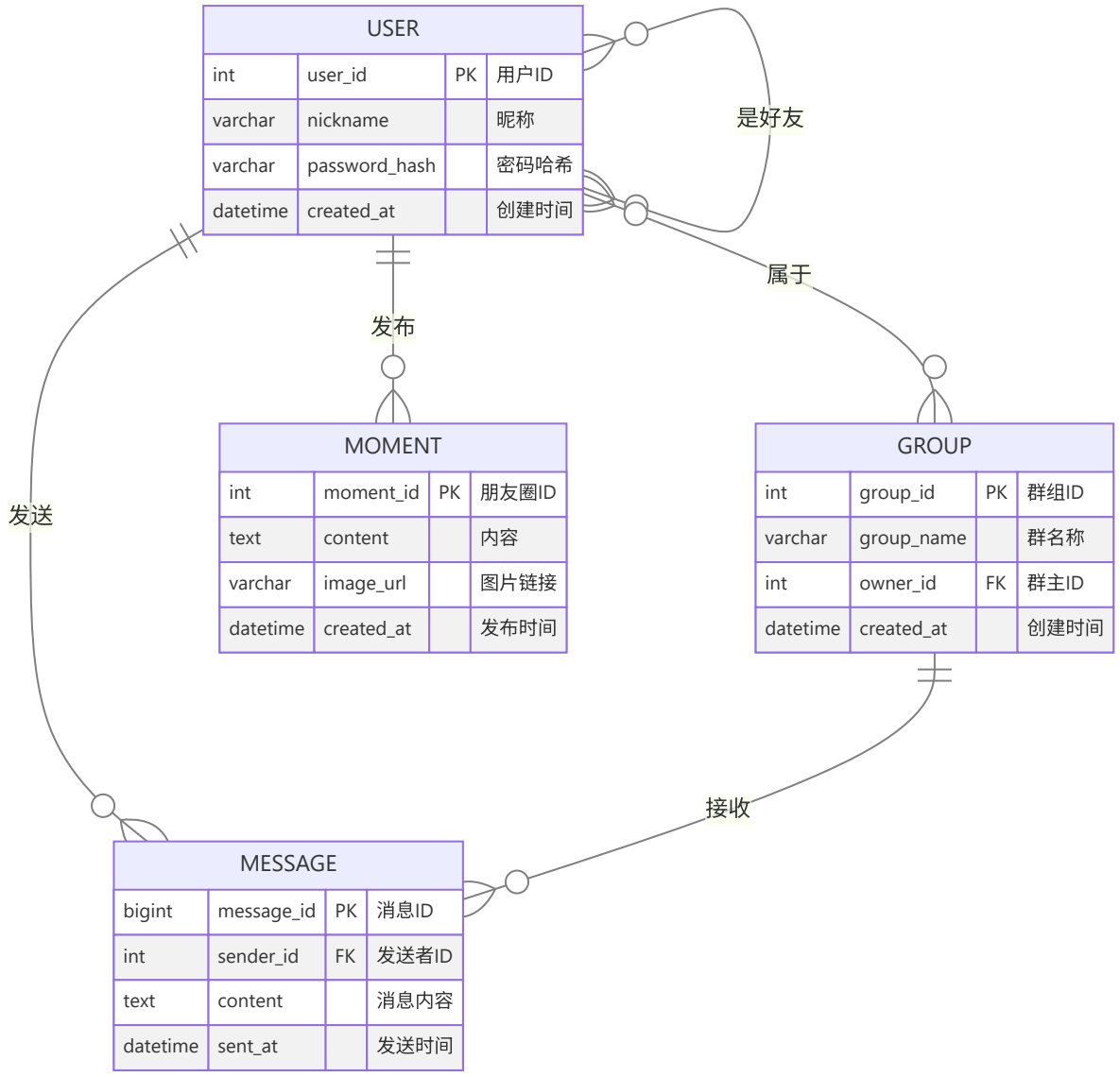
设计微信数据库核心模式

1.概念结构设计

核心实体与联系分析
微信的核心功能是社交，主要围绕以下几点展开：

- **实体** (Entities):
  - **用户** (User): 微信的基础，每个操作都离不开用户。
  - **群组** (Group): 多人聊天的载体。
  - **消息** (Message): 用户之间沟通的内容。

    - **朋友圈 (Moment)**: 用户分享生活动态的功能。
- **联系 (Relationships)**:
    - 用户与用户之间是 **好友关系** (多对多, m:n)。一个用户可以有多个好友，反之亦然。
    - 用户与群组之间是 **成员关系** (多对多, m:n)。一个用户可以加入多个群组，一个群组可以包含多个用户。
    - 用户与消息之间是 **发送关系**。这可以看作是用户之间或用户与群组之间的联系。
    - 用户与朋友圈之间是 **发布关系** (一对多, 1:n)。一个用户可以发布多条朋友圈。
    - 朋友圈与评论之间是 **评论关系** (一对多, 1:n)。一条朋友圈可以有多条评论。



- `PK` 表示主键，`FK` 表示外键。
- `||--o{` 表示一对多关系
- `}o--o{` 表示多对多关系

2.逻辑结构设计

将E-R图转换为关系模式：

用户表 (Users)

字段名	数据类型	约束	描述
`user_id`	INT	主键 (PK)	唯一用户标识
`nickname`	VARCHAR(50)	NOT NULL	用户昵称
`password_hash`	VARCHAR(255)	NOT NULL	加密后的密码
`created_at`	DATETIME		注册时间

好友关系表 (Contacts)

字段名	数据类型	约束	描述
`user_id_1`	INT	**复合主键, 外键 (FK)**	用户1的ID
`user_id_2`	INT	**复合主键, 外键 (FK)**	用户2的ID
`status`	VARCHAR(20)		关系状态 (如 'friend', 'blocked')
`created_at`	DATETIME		成为好友的时间

群组表 (Groups)

字段名	数据类型	约束	描述
`group_id`	INT	**主键 (PK)**	唯一群组标识
`group_name`	VARCHAR(50)	NOT NULL	群名称
`owner_id`	INT	外键 (FK)	群主的 `user_id`
`created_at`	DATETIME		创建时间

群成员表 (GroupMembers)

字段名	数据类型	约束	描述
`group_id`	INT	**复合主键, 外键 (FK)**	群组ID
`user_id`	INT	**复合主键, 外键 (FK)**	用户ID
`joined_at`	DATETIME		加入时间
`role`	VARCHAR(20)		角色 (如 'member', 'admin')

消息表 (Messages)

字段名	数据类型	约束	描述
`message_id`	BIGINT	**主键 (PK)**	唯一消息标识

字段名	数据类型	约束	描述
`sender_id`	INT	外键 (FK)	发送者的`user_id`
`receiver_id`	INT	外键 (FK), NULLABLE	接收者的`user_id` (私聊)
`group_id`	INT	外键 (FK), NULLABLE	接收群的`group_id` (群聊)
`content`	TEXT		消息内容
`sent_at`	DATETIME		发送时间

朋友圈表 (Moments)

字段名	数据类型	约束	描述
`moment_id`	INT	**主键 (PK)**	唯一动态标识
`user_id`	INT	外键 (FK)	发布者的`user_id`
`content`	TEXT		文字内容
`image_urls`	TEXT	NULLABLE	图片链接 (可存JSON格式)
`created_at`	DATETIME		发布时间