

《数据库系统原理》课程设计

系统设计报告

题目名称：综合聊天系统

学号及姓名：13061210，何涛

13061192，赵东方

2015 年 11 月 26 日

目录

- 一、需求分析.....4
 - 1. 需求描述.....4
 - 2. 数据流图.....5
 - 3. 数据元素表.....7
- 二、数据库概念模式设计.....14
 - 1. 系统初步 E-R 图.....15
 - 2. 系统基本 E-R 图.....17
- 三、数据库逻辑模式设计.....18
 - 1. 数据库关系模式.....18
 - 2. 关系模式范式等级的判定与规范化.....18
 - 3. 数据库设计优化.....22
- 四、最终版修改说明.....23

组内同学承担任务说明

	赵东方	何涛
系统设计阶段	<ul style="list-style-type: none">✓ 需求描述✓ 数据库关系模式✓ 关系模式范式等级的判定与规范化✓ 数据库设计优化	<ul style="list-style-type: none">✓ 数据流图✓ 数据元素表✓ 系统初步 E-R 图✓ 系统基本 E-R 图
系统实现阶段	<ul style="list-style-type: none">✓ 定义触发器和存储过程✓ 通过 Servlet 实现用户、群组和聊天室管理功能✓ 通过 WebSocket 实现聊天功能的后台逻辑✓ 实现 Servlet 到具体 URL 的映射	<ul style="list-style-type: none">✓ 建立数据库和表✓ 实现基本界面和交互✓ 实现聊天功能的前端部分
系统报告撰写	<ul style="list-style-type: none">✓ 系统功能需求分析✓ 数据库基本表的定义✓ 触发器和存储过程定义与实现✓ 项目总结	<ul style="list-style-type: none">✓ 系统功能结构设计✓ 系统实现结果✓ 项目总结

一. 需求分析

1. 需求描述

通过调查相关的聊天软件，发现用户对于聊天系统的需求包括：用户与用户之间能够进行实时通信，群组中每个用户相互之间的实时通信。我们的综合聊天系统在包括了这些需求的基础之上，增加了聊天室功能，改变以往的群组模式，使得陌生人之间也能对同一个话题进行沟通。由于聊天室的信息只能被当时在聊天室中的人接收到，所以我们为每个聊天室设置了留言板，方便通知一些信息给关注某个聊天室的用户。在用户与用户聊天时，往往想知道对方是否已经读过自己的消息，来确定对方是否有回应的意图，我们增加了显示消息状态的功能，消息被查看，则会被标记为已读。下面将对系统在信息要求、处理要求 and 安全性要求上进行进一步分析。

(1) 信息要求

- 1) 用户注册，用户将注册信息填好，这些信息将存入数据库中。
- 2) 用户登陆，用户通过注册时的提交的用户 ID 和密码来登陆综合聊天系统。判断是否有该用户，以及密码是否正确，需要从数据库中进行查找。
- 3) 用户资料修改，用户可以修改自己的个人信息，包括昵称，密码，邮箱等。修改后会将新的信息存入数据库中。
- 4) 查看他人资料，通过他人用户 ID，来在数据库中进行查找并显示。
- 5) 加好友以及加群组功能，用户可以申请与其他用户加为好友，或是加入某些群组。建立联系后的信息会被存在数据库中。
- 6) 查看好友列表和群组列表，需要在数据库中对好友列表和群组列表进行查询。
- 7) 发送消息，允许用户给自己的好友，自己所在的群组以及聊天室发送消息。每条消息将会被储存在数据库中，包括消息内容，消息发送时间，消息状态以及消息发送者等一些信息。
- 8) 查看聊天记录，用户可以查看自己与好友以及自己所在群组中的聊天记录，这些信息可以在数据库中进行查找。

9) 留言, 用户可以在聊天室的留言板上留言, 留言将会被存在数据库中。

(2) 处理要求

1) 用户在聊天的过程中, 要保证消息立即被接收者所接收。

2) 在用户查看他人资料与查看聊天记录时, 响应速度尽量控制在用户可忍受的等待时间内。

(3) 安全性要求

1) 用户的密码用 MD5 进行加密处理, 避免个人信息以及个人聊天记录的泄露。

2) 对用户进行权限分类, 来限制在系统中的否写操作, 避免数据被破坏。

2. 数据流图

系统的数据流全部是用户和聊天系统进行交互过程中的数据流, 顶层数据流图表现了本系统的基本功能。

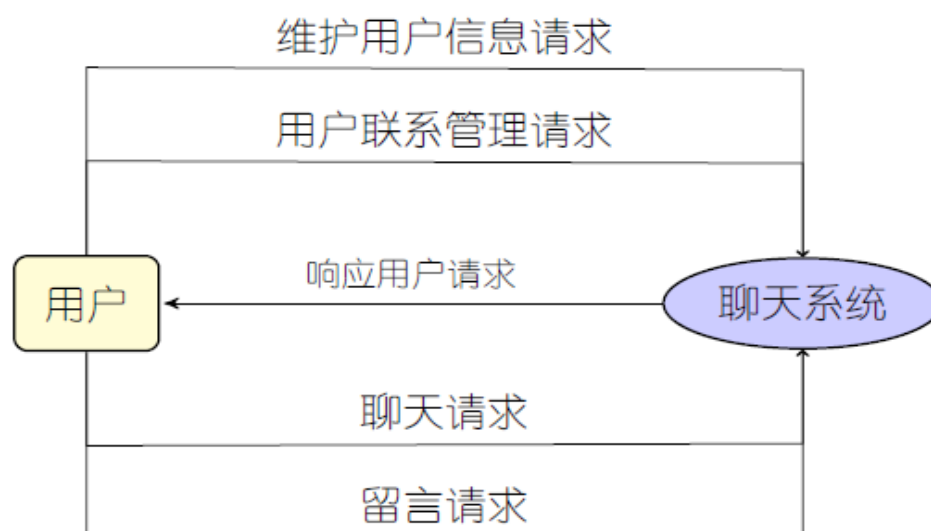


图 1.2.1: 顶层数据流图

0 层数据流图细化了聊天系统的作用, 从用户信息管理、用户关系管理和聊天与留言三部分进行详细阐释。

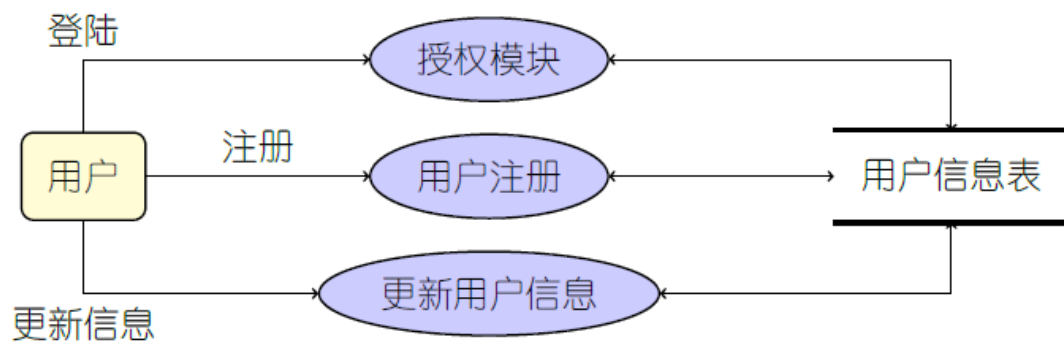


图 1.2.2：0 层数据流图第一部分：用户信息管理

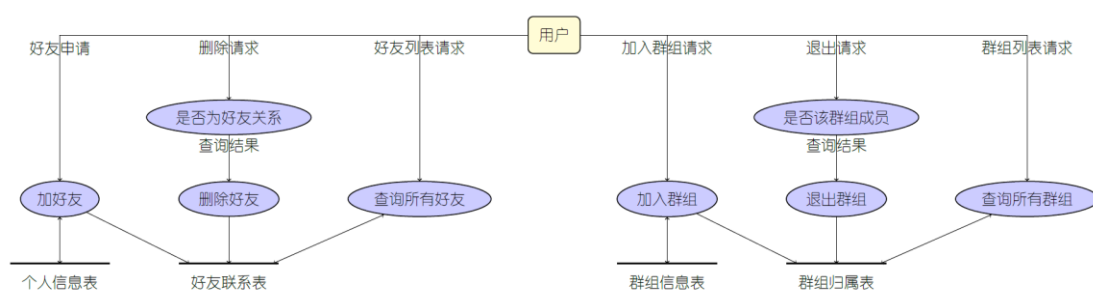


图 1.2.3：0 层数据流图第二部分：好友关系管理

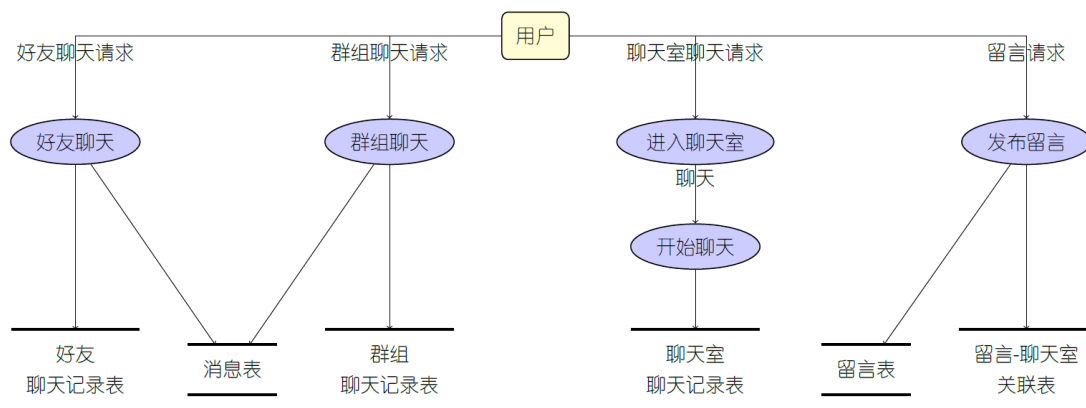


图 1.2.4：0 层数据流图第三部分：聊天与留言

1 层数据流图针对加好友、加群组、好友聊天这三部分较为复杂的流程进行了细化，进一步阐释系统的功能和数据流方面的特点。

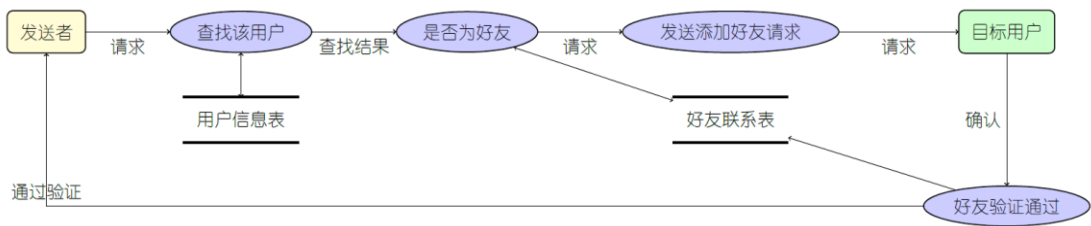


图 1.2.5：1 层数据流图：申请加为好友

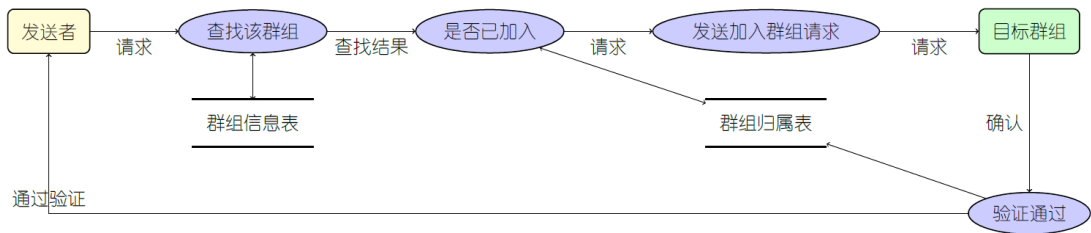


图 1.2.6：1 层数据流图：申请加入群组

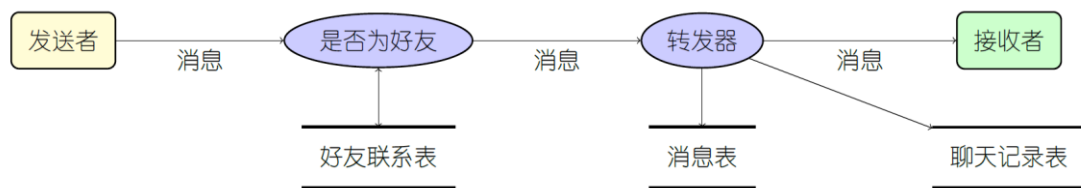


图 1.2.7：1 层数据流图：好友间聊天

3. 数据元素表

(1) 用户表

特征	用户号	用户昵称	用户密码	用户邮箱
	E01	E02	E03	E04
数据类型	int	varchar	varchar	varchar

数据宽度	10	30	6-18	20
允许空值否	否	否	否	是
说明	用户编号	用户自定义的昵称	用户密码	用户邮箱

(2) 群组表

特征	群组号	群组名称	群组管理员	群组说明
	E01	E02	E03	E04
数据类型	int	varchar	varchar	varchar
数据宽度	8	30	6-18	50
允许空值否	否	否	否	是
说明	群组编号	群组的名称	群组管理员编号	用户的特点、性质等说明

(3) 消息表

特征	消息编号	发出时间	消息内容	消息状态
	E01	E02	E03	E04
数据类型	int	data.time	varchar	bool
数据宽度	10		100	
允许空值否	否	否	否	否

说明	消息编号	消息发出的时间	消息的内容	接受者是否已经阅读（是表示已读，否表示未读）
----	------	---------	-------	------------------------

（4）留言表

特征	留言号	留言时间	留言内容
	E01	E02	E03
数据类型	int	data.time	varchar
数据宽度	10		100
允许空值否	否	否	否
说明	留言编号	留言的时间	留言的内容

（5）聊天室表

特征	聊天室号	聊天室名称	聊天室介绍
	E01	E02	E03
数据类型	int	varchar	varchar
数据宽度	10	30	50
允许空值否	否	否	是
说明	聊天室编号	聊天室名称	对聊天室的特点、性质等的介绍

(6) 留言板号

特征	留言板号
	E01
数据类型	int
数据宽度	10
允许空值否	否
说明	留言板编号

(7) 好友关系表

特征	用户 A	用户 B	验证状态
	E01	E02	E03
数据类型	int	int	bool
数据宽度	10	10	
允许空值否	否	否	否
说明	好友关系中用户 A 的用户号	好友关系中用户 B 的用户号	好友关系验证状态

(8) 留言发布记录表

特征	用户号	留言号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	用户编号	留言编号

(9) 聊天室状态表

特征	聊天室号	用户号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	聊天室编号	在聊天室中的 用户的编号

(10) 聊天记录表

特征	用户 A	用户 B	消息号
	E01	E02	E03
数据类型	int	int	int
数据宽度	10	10	10
允许空值否	否	否	否
说明	用户编号	留言编号	聊天过程中的 的消息的编号

(11) 消息发送记录表

特征	用户号	消息号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	用户编号	用户发的消息 编号

(12) 群组归属表

特征	群组号	用户号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	群组编号	用户编号

（13）群组聊天记录表

特征	群组号	消息号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	群组编号	群组聊天记录中的消息编号

（14）留言归属表

特征	留言板号	留言号
	E01	E02
数据类型	int	int
数据宽度	10	10
允许空值否	否	否
说明	留言板编号	留言编号

(15) 留言板归属表

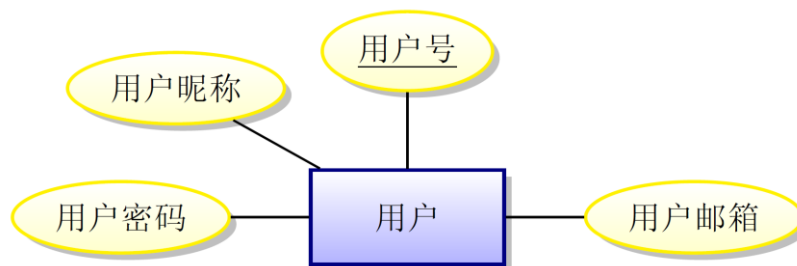
特征	留言板号	聊天室号
	E01	E02
数据类型	int	char
数据宽度	10	30
允许空值否	否	否
说明	留言板编号	留言板所属的聊天室的编号

二. 数据库概念模式设计

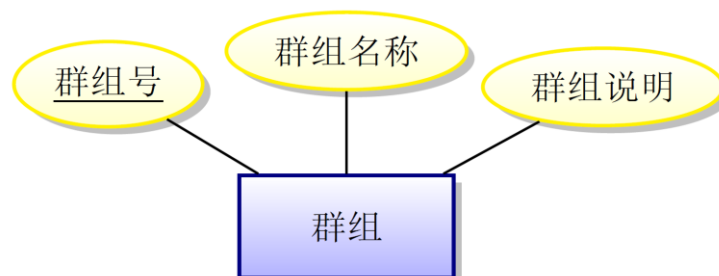
1. 系统初步E-R 图

(1) 各实体属性信息

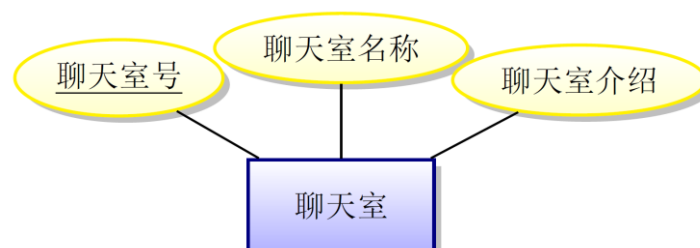
- 用户:



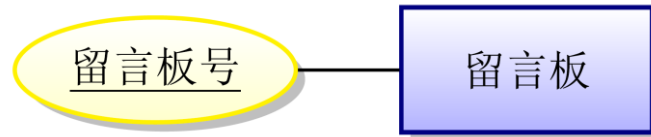
- 群组:



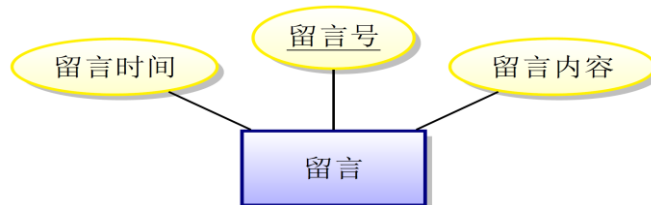
- 聊天室:



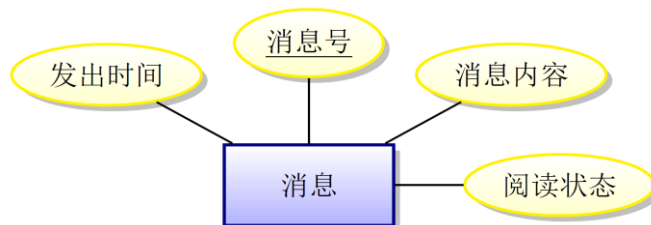
- 留言板:



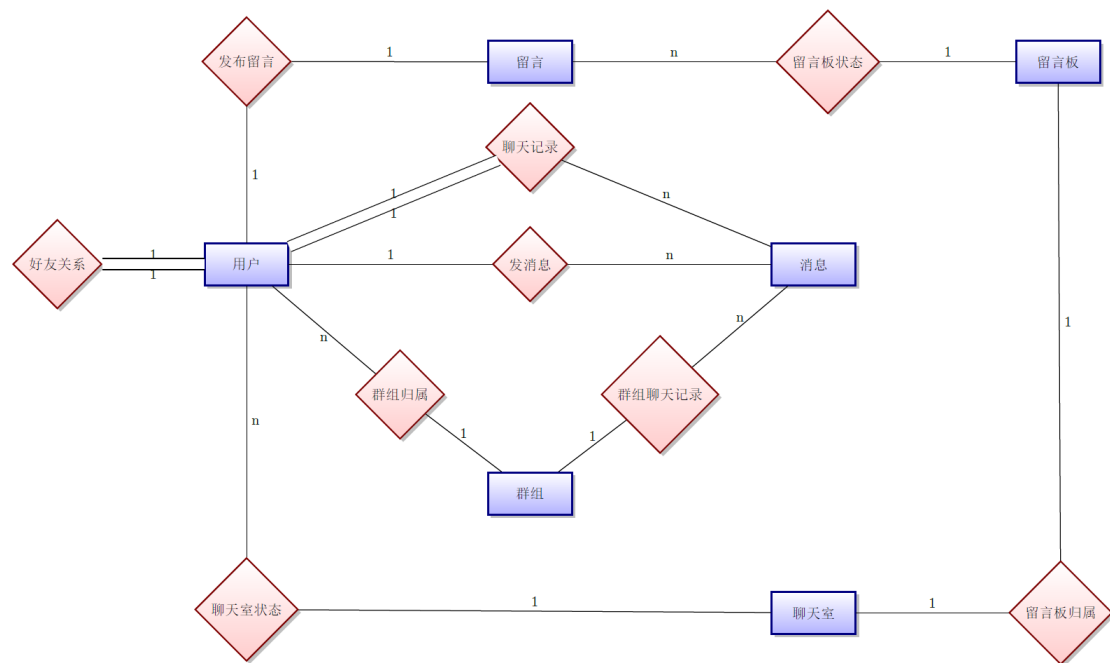
- 留言:



- 消息:

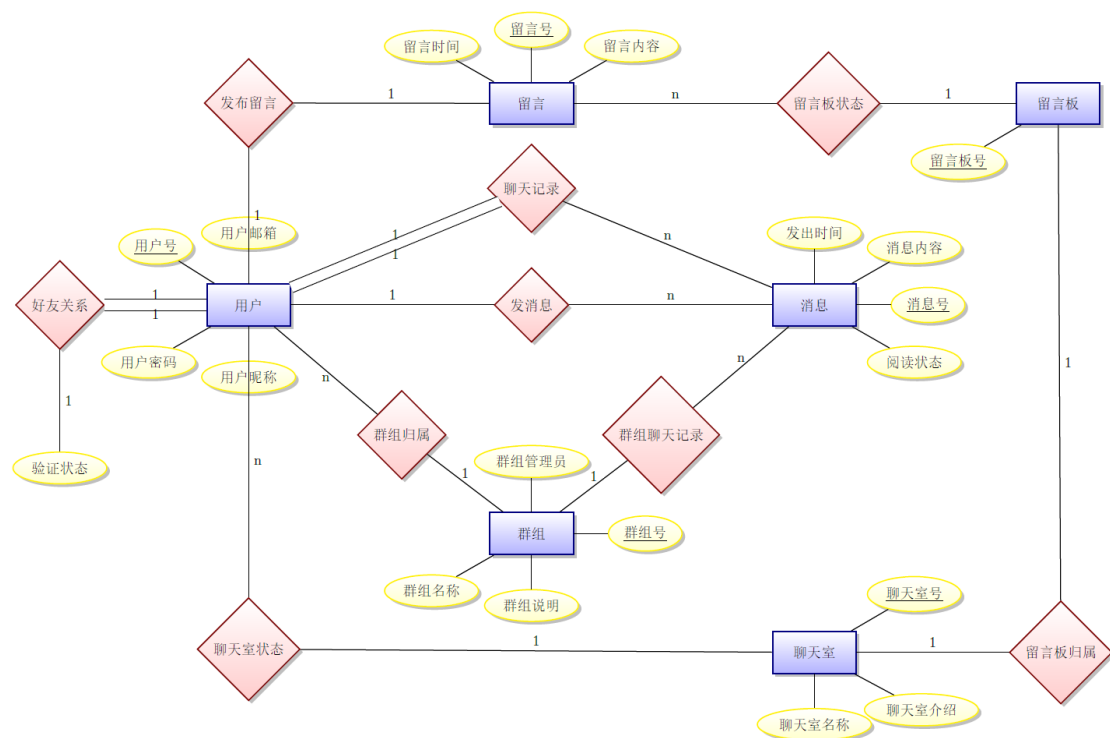


(2) 将上述各实体按照实体间的联系合并:



2. 系统基本 E-R 图

本聊天系统的系统基本 E-R 图如下所示：



三、数据库逻辑模式设计

1. 数据库关系模式

- 用户(用户号, 用户昵称, 用户密码, 用户邮箱)
- 群组(群组号, 群组名称, 群组说明, 用户号)
- 聊天室(聊天室号, 聊天室名称, 聊天室介绍)
- 留言板(留言版号)
- 留言(留言号, 留言内容, 留言时间)
- 消息(消息号, 消息内容, 发出时间, 阅读状态)
- 好友关系(用户号 A, 用户号 B, 验证状态)
- 群组归属(群组号, 用户号)
- 聊天室现状(聊天室号, 用户号)
- 留言板状态(留言号, 留言板号)
- 聊天记录(消息号, 用户号 A, 用户号 B)
- 群组聊天记录(消息号, 群组号)
- 发消息(消息号, 用户号)
- 发布留言(留言号, 用户号)
- 留言板归属(留言板号, 聊天室号)

2. 关系模式范式等级的判定与规范化

- 用户(用户号, 用户昵称, 用户密码, 用户邮箱)

函数依赖:

用户号→用户昵称

用户号→用户密码

用户号→用户邮箱

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- **群组(群组号, 群组名称, 群组说明, 用户号)**

函数依赖:

群组号→群组名称

群组号→群组说明

群组号→用户号

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- **聊天室(聊天室号, 聊天室名称, 聊天室介绍)**

函数依赖:

聊天室号→聊天室名称

聊天室号→聊天室介绍

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- **留言板(留言版号)**

全码满足 3NF

- **留言(留言号, 留言内容, 留言时间)**

函数依赖:

留言号→留言内容

留言号→留言时间

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 消息(消息号, 消息内容, 发出时间, 阅读状态)

函数依赖:

消息号→消息内容

消息号→发出时间

消息号→阅读状态

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 好友关系(用户号 A, 用户号 B, 验证状态)

函数依赖:

(用户号 A, 用户号 B) → 验证状态

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 群组归属(群组号, 用户号)

函数依赖:

群组号→用户号

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 聊天室现状(聊天室号, 用户号)

函数依赖:

聊天室号→用户号

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 留言板状态(留言号, 留言板号)

函数依赖:

留言号→留言板号

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 聊天记录(消息号, 用户号 A, 用户号 B)

函数依赖:

消息号→用户号 A

消息号→用户号 B

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 群组聊天记录(消息号, 群组号)

函数依赖:

消息号→群组号

从上面的信息可以看出, 每一个非主属性完全依赖于码, 且不传递依赖于码, 因此属于 3NF

- 发消息(消息号, 用户号)

函数依赖:

消息号→用户号

从上面的信息可以看出，每一个非主属性完全依赖于码，且不传递依赖于码，因此属于 3NF

- **发布留言(留言号, 用户号)**

函数依赖:

留言号→聊天室号

从上面的信息可以看出，每一个非主属性完全依赖于码，且不传递依赖于码，因此属于 3NF

- **留言板归属(留言板号, 聊天室号)**

函数依赖:

留言板号→聊天室号

聊天室号→留言板号

从上面的信息可以看出，没有非主属性，因此属于 3NF

综上所述，关系模式范式等级为 3NF

3. 数据库设计优化

(1) 水平分解

将消息分别和聊天记录，群组聊天记录和聊天室聊天记录进行合并，然后进行水平分解。

对于聊天记录，关于用户号 A 和用户号 B，进行分组；对于群组聊天记录，关于群组号进行分组；对于聊天室聊天记录，关于聊天室号进行分组。这样就可以省下很多查询时间。

当用户在聊天的时候，往往只会看到之前的 10 条左右的聊天记录信息，而不用每次都把所有的聊天记录载入出来。在上面水平分解的基础上，继续进行水平分解，按发出时间，分解出最近二十条信息。

(2) 合并

1) 将消息(消息号，消息内容，发出时间，阅读状态)和发消息(消息号，用户号)进行合并，因为对一个消息进行处理时，往往要知道是谁发出的信息。同理，也将发布留言留言(留言号，留言内容，留言时间)和(留言号，用户号)

2) 将留言板(留言板号)和留言板归属(留言板号，聊天室号)合并，因为留言板和聊天室是一一对应的。

四、最终版修改说明

1. 将聊天室记录删除，因为聊天室本身不应该有记录。
2. 完善小组分工。