## 5.1 MySQL数据库介绍

数据库介绍+范式基础+如何实现高效的数据库设计

数据库是一个存储数据集合的独立应用程序。 每个数据库都有一个或多个不同的API，用于创建，访问，管理，搜索和复制它所拥有的数据。也可以使用其他类型的数据存储，例如文件系统上的文件或内存中的大型哈希表，但是对于这些类型的系统，数据获取和写入不会那么快速和容易。

如今，我们使用关系数据库管理系统（RDBMS）来存储和管理大量数据。 这称为关系数据库，因为所有数据都存储在不同的表中，并且使用主键或称为外键的其他键建立关系。

关系数据库管理系统（RDBMS）是一种软件使您能够实现包含表，列和索引的数据库、保证各种表行之间的参照完整性、自动更新索引、解释SQL查询并组合来自各种表的信息。

MySQL是一种快速，易于使用的RDBMS，可用于许多小型和大型企业。 MySQL由瑞典公司MySQL AB开发，营销和支持。由于很多很好的理由，MySQL变得如此受欢迎MySQL是在开源许可下发布的。因此无需支付任何费用。

MySQL本身就是一个非常强大的程序。它处理最昂贵和功能最强大的数据库包的大部分功能。MySQL使用着名的SQL数据语言的标准形式。MySQL可以在许多操作系统上运行，并且可以使用许多语言，包括PHP，PERL，C，C ++，JAVA等。MySQL工作非常快，即使对于大型数据集也能正常工作。MySQL对PHP非常友好，PHP是用于Web开发的最受欢迎的语言。MySQL支持大型数据库，表中包含多达5000万行或更多行。表的默认文件大小限制为4GB，但您可以将此值（如果您的操作系统可以处理它）增加到理论上的800万TB（TB）。MySQL是可定制的。开源GPL许可证允许程序员修改MySQL软件以适应他们自己的特定环境。

## 5.2 NOSQL及Redis数据库介绍

几乎每个应用程序都必须存储数据，并且通常需要大量快速变化的数据。直到最近，大多数应用程序使用关系数据库管理系统（简称RDBMS）存储数据，如Oracle，MySQL或PostgreSQL。然而，最近，由于需要以更有效的方式存储无模式数据NoSQL，因此出现了一种新的数据存储模式。选择是使用SQL还是NoSQL通常是设计成功应用程序的重要第一步。在选择是否使用SQL或NoSQL来存储数据时，有两件重要的事情要考虑：它的性质和使用模式。某些数据非常适合关系存储引擎，而其他数据则受益于Nois引擎（如Redis或其替代品）的无模式特性。

在Web上工作时，数据和数据模型可能会随着增加的功能和业务更新而不断变化。在关系数据库中发展模式以支持这些更改是一个痛苦的过程， 特别适合非关联存储的数据示例包括事务详细信息，历史数据和服务器日志。这些通常是高度动态的，经常变化，并且它们的存储趋于快速增长，进一步加剧了调整模式以存储它们的问题。它们通常也不会感觉到“关系” 。也就是说，它们中的数据不会在与其他类型数据的关系中散开。这是一个很好的迹象，表明他们可以使用RDBMS以外的东西。

衡量NoSQL适用性的另一种方法是查看是否因性能原因而发现自己的数据非规范化，并且不再受益于关系系统的某些优势，例如一致性和冗余检查。NoSQL数据库通常不提供ACID（原子性，一致性，隔离性，持久性），或者仅部分地执行它。这允许他们做出一些其他情况下无法实现的权衡。 Redis通过设计提供部分ACID合规性，因为它是单线程（保证一致性和隔离），并且如果始终配置appendfsync，则提供完全合规性，同时提供持久性。表现也可能是一个关键因素。 NoSQL数据库通常更快，特别是对于写操作，使它们非常适合大量写入的应用程序。

Redis是一个开源（BSD许可），内存数据结构存储，用作数据库，缓存和消息代理。它支持数据结构，如字符串，散列，列表，集合，带有范围查询的排序集，位图，超级日志，具有半径查询和流的地理空间索引。 Redis具有内置复制，Lua脚本，LRU驱逐，事务和不同级别的磁盘持久性，并通过Redis Sentinel提供高可用性并使用Redis Cluster自动分区。

您可以对这些类型运行原子操作，例如附加到字符串;递增哈希值;将元素推送到列表中;计算集合交集，并集和差异;或者在排序集中获得排名最高的成员。

为了实现其出色的性能，Redis使用内存数据集。根据您的使用情况，您可以通过每隔一段时间将数据集转储到磁盘或通过将每个命令附加到日志来保留它。如果您只需要功能丰富的网络内存缓存，则可以选择禁用持久性。

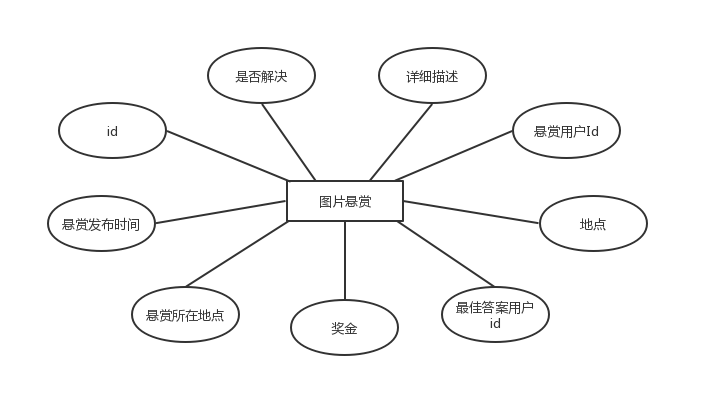
Redis还支持简单到设置的主从异步复制，具有非常快速的非阻塞第一次同步，自动重新连接以及在网络分割上的部分重新同步。其他功能包括：交易、发布/订阅、Lua脚本、钥匙的生存时间有限、LRU逐出钥匙、自动故障转移

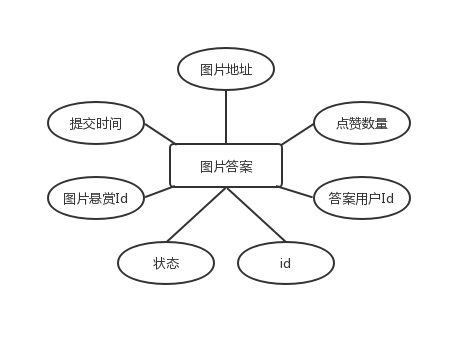
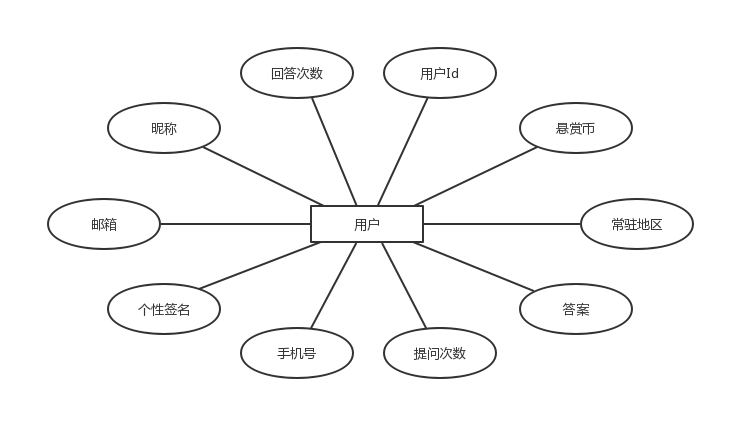
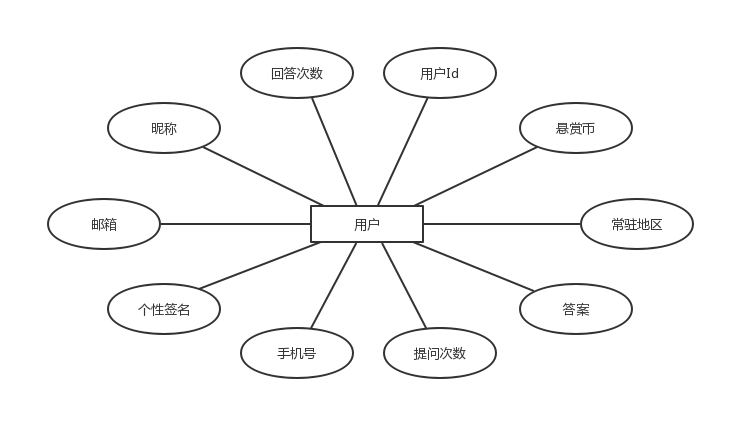
Redis是用ANSI C编写的，适用于大多数POSIX系统，如Linux，\* BSD，OS X，没有外部依赖性。 Linux和OS X是Redis开发和测试的两个操作系统，我们建议使用Linux进行部署。 Redis可能在Solaris衍生系统（如SmartOS）中工作，但支持是最好的努力。 Windows版本没有官方支持，但Microsoft开发并维护了Redis的Win-64端口。

所有这些都说，尽管NoSQL感觉更灵活，但在RDBMS中存储关系数据也有很好的论据。如果您具有非常适合规范化的可预测数据，则可以获得使用关系数据存储引擎的好处。在做出决定之前，请始终查看数据。像Redis这样的NoSQL数据库速度快，容易扩展，非常适合许多现代问题。一旦加权了所有选项，在SQL（对于稳定的，可预测的，关系数据）和NoSQL（对于临时的，高度动态的数据）之间进行选择应该是一件容易的事。

## 5.3 数据库结构设计

本论坛中分为人员信息分为普通用户和悬赏审核管理员。普通用户和悬赏审核管理员在实际操作中具有的权限不同。普通用户具有增加、修改、删除悬赏信息、答案信息、个人信息的权限。管理员具有修改答案状态的权限。本文绘制了Mysql数据库的相应E-R图。





## 5.4 数据库表设计

数据库的设计原则上要按照第三范式进行设计，一边进行后续的拓展使用，本论坛在数据库设计过程中本着第三范式的原则，在一定程度上提高了整个数据库应用的整体性能。

在本章节中我们会将设计的数据库表进行展示。

1、用户信息表

用户信息表记录了每个普通用户的基本信息，用户表user\_mes相关字段的定义见表5.1。

表 5.1 用户表user\_mes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| 用户id | user\_id | varchar(32) | 主键 |
| 邮箱 | email | varchar(64) |  |
| 个性签名 | signature | text |  |
| 手机 | phone | varchar(16) |  |
| 悬赏币 | money | decimal |  |
| 常驻地区 | region | varchar(64) |  |
| 提问个数 | ask | int |  |
| 回答个数 | answer | int |  |
| 密码 | password | varchar(32) |  |
| 昵称 | Nickname | varchar(32) |  |

2、管理员表

管理员表记录了每个管理员的id及密码信息，管理员表admin相关字段的定义见表5.2。

表 5.2 管理员表admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| 管理员id | admin\_id | varchar(32) | 主键 |
| 密码 | password | varchar(32) |  |

3、账单表

账单表记录了每个账单的相关信息，账单表bill\_schedule相关字段的定义见表5.3。

表 5.3 账单表bill\_schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| 账单id | bill\_id | int | 主键 |
| 金额 | money | decimal |  |
| 账单类型 | type | int |  |
| 创建账单的时间 | ctime | timestamp |  |
| 悬赏id | app\_id | int |  |

4、等级信息表

等级信息表记录了每个等级对应的起始经验和最高经验，由普通用户在线时长及答案分数表中查询计算得到普通用户的总经验值之后通过与该表比对，得到该用户的等级。等级信息表level相关字段的定义见表5.4。

表 5.4 等级信息表level

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| 等级 | level | int | 主键 |
| 起始经验 | start\_score | int |  |
| 最高经验 | end\_score | int |  |

5、普通用户在线时长及答案分数表

普通用户在线时长及答案分数表记录了每个普通用户的在线时长及回答悬赏所得到的分数，此表内的信息用于计算该用户的等级。用户表level\_score相关字段的定义见表5.1。

表 5.1 消息表level\_score

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 在线时长 | login\_time | int |  |
| 回答悬赏得分 | answer\_score | int |  |

6、消息表

消息表记录了要提示给用户的相关消息，其中包括回答悬赏提示、答案审核通过通知、答案中标通知、答案涉嫌违规通知、充值通知、发布悬赏通知等。账单表message相关字段的定义见表5.6。

表 5.6 用户表message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 消息内容 | comment | text |  |
| 消息发布时间 | time | timestamp |  |
| 消息类型 | type | int |  |
| 消息详情 | link | varchar(128) |  |
| 消息阅读状态 | state | int |  |

7、图片答案表

图片答案表包含了回复图片悬赏的答案的相关信息。图片答案表pic\_answer相关字段的定义见表5.7。

表 5.7 图片答案表pic\_answer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 图片悬赏id | picapp\_id | int |  |
| 上传图片地址 | pic\_adress | varchar(256) |  |
| 回答时间 | answer\_time | time\_stamp |  |
| 答案状态 | state | int |  |
| 点赞量 | awesome | int |  |

7、图片悬赏表

图片悬赏含了回复图片悬赏的相关信息。图片悬赏表pic\_app\_mes相关字段的定义见表5.7。

表 5.7 图片答案表pic\_answer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 悬赏城市 | city | varchar(64) |  |
| 悬赏简介 | introduction | text |  |
| 悬赏发布时间 | app\_time | time\_stamp |  |
| 悬赏金额 | money | decimal |  |
| 是否解决 | solve | int |  |
| 正确答案用户id | right\_user\_id | varchar(32) |  |
| 所属地区 | region | varchar(64) |  |

8、图片点赞表

图片点赞表包含了图片悬赏答案点赞的相关信息。图片点赞表pic\_awsome相关字段的定义见表5.7。

表 5.8 图片点赞表pic\_awsome

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 悬赏答案id | answeri\_id | int |  |
| 点赞用户id | user\_id | varchar(32) |  |

9、在线时间统计表

图片统计表包含了回复用户本次登录的在线时间的相关信息，用于计算每次用户登录的在线时长。在线时间统计表time\_satistic相关字段的定义见表5.7。

表 5.9 在线时间统计表time\_satistic

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 登录时间 | login\_time | timestamp |  |
| 最后操作时间 | latestOptime | timestamp |  |

10、视频答案表

视频答案表包含了回复视频悬赏的答案的相关信息。图片答案表video\_answer相关字段的定义见表5.7。

表 5.10 视频答案表video\_answer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 视频悬赏id | videoapp\_id | int |  |
| 上传图片地址 | video\_adress | varchar(256) |  |
| 回答时间 | answer\_time | time\_stamp |  |
| 答案状态 | state | int |  |
| 点赞量 | awesome | int |  |

11、视频悬赏表

视频悬赏含了回复视频悬赏的答案的相关信息。视频悬赏表video\_app\_mes相关字段的定义见表5.7。

表 5.11 视频答案表video\_answer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 用户id | user\_id | varchar32) |  |
| 悬赏城市 | city | varchar(64) |  |
| 悬赏简介 | introduction | text |  |
| 悬赏发布时间 | app\_time | time\_stamp |  |
| 悬赏金额 | money | decimal |  |
| 是否解决 | solve | int |  |
| 正确答案用户id | right\_user\_id | varchar(32) |  |
| 所属地区 | region | varchar(64) |  |

12、视频点赞表

视频点赞表包含了视频悬赏答案点赞的相关信息。视频点赞表video\_awsome相关字段的定义见表5.7。

表 5.8 图片点赞表video\_awsome

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段说明 | 字段名 | 属性 | 备注 | |
| id | id | int | 主键 |
| 悬赏答案id | answeri\_id | int |  |
| 点赞用户id | user\_id | varchar(32) |  |