

# ipicstory 数据库设计说明

---

用户信息表 USER

字段名称	类型	描述
idUSER	Bigint(20)PK AI	自增的 long 型主键
fansCount	Int(11)	粉丝数量
feedsCount	Int(11)	发布的消息数量
followerCount	Int(11)	关注的数量
UserName	Varchar(255)	用户名
UserPass	Varchar(255)	用户密码

用户信息关系表 USERRELATION

字段名称	类型	描述
Iduserrelation	Bigint(20)PK AI	自增的 long 型主键
followerId	Bigint(20)	被关注者编号
UserId	Bigint(20)	用户编号
type	Int(11)	关系类型 (0,粉丝;1,关注)

用户消息索引表 FEED

字段名称	类型	描述
IdFEEDS	Bigint(20)PK AI	自增 long 型主键
Ukey	Varchar(255)	系统为每一条消息生成的 key
timeLine	datetime	消息的时间戳
USER_idUSER	Bigint(20)FK	外键用户 id

消息与消息的关系表 FEEDRELATION

字段名称	类型	描述
idFEEDRELATION	Bigint(20)PK AI	自增 long 型主键
referFeedId	Bigint(20)	引用的消息编号
referUserId	Bigint(20)	引用的消息用户编号
referredFeedId	Bigint(20)	被引用的 (主) 消息编号
referredUserId	Bigint(20)	消息发布者编号
timeL	datetime	发布时间
type	Int(11)	操作类型 (0,转发;1,评论)

## 故事表 STORY

字段名称	类型	描述
IdSTORY	Bigint(20)PK AI	自增 long 型主键
commentCount	Int(11)	评论数量
Latitude	Double	发布的经度 lbs
Longitude	Double	发布的纬度 lbs
storyCode	Varchar(255)	故事的唯一编码
Storyname	Varchar(255)	故事名称
Storydesc	Varchar(255)	故事描述
TimeI	Datetime	发布时间
transferrCount	Int(11)	转发数量
Type	Int(11)	故事类型 (0 原创, 1 评论 2,转发)
FEED_idFEED	Bigint(20)FK	消息原 id
USER_idUSER	Bigint(20)FK	消息原所属用户 id

## 故事关注度表 STORYFOLLOWER

字段名称	类型	描述
IdSTORYFOLLOWER	Bigint(20)PK AI	自增 long 型主键
Followtime	Datetime	用户关注时间
userId	Bigint(20)	关注该故事的用户
STORY_idSTORY	Bigint(20)FK	外键故事 id

## 故事的瞬间表 MOMENT

字段名称	类型	描述
idMOMENT	Bigint(20)PK AI	自增 long 型主键
pMonDesc	Varchar(255)	对故事瞬间的描述
pMonIndex	Int(11)	瞬间在故事中的索引
STORY_idSTORY	Bigint(20)FK	外键故事 id

## 页面表（包括视频图片的表现）PAGE

字段名称	类型	描述
IdPAGE	Bigint(20)PK AI	自增 long 型主键
Content	Varchar(255)	页面的描述
mediaType	Int(11)	多媒体文件没类型 (0,图片 1 视频 2 音频)
mediaUrl	Varchar(255)	资源文件地址
Moment_idMoment	Bigint(20)FK	外键瞬间 id

## 评论表 COMMEND

字段名称	类型	描述
idCOMMEND	Bigint(20)PK AI	自增 long 型主键
commendText	Varchar(255)	评论的内容
FEEDRELATION_idFEEDRELATION	Bigint(20)PK	消息关系的 id

### 业务逻辑

#### 1. A 关注 B

在 USERRELATION 中添加

A	B	1
B	A	0

#### 2.原创发消息

- 1) 在 STORY 中添加这条原消息, type=0
- 2) 更新 USER 的 feedsCount+1
- 3) 在 FEED 中插入这条消息进入缓存
- \*4) 在 USERRELATION 中找出所有关注 A 的人.并将 FEED 中 A 的消息拉给关注者

#### 3.A 转发 B 的故事

- 1) 在 STORY 中添加这条原消息, type=2
- 2) 更新 USER 的 feedsCount+1
- 3) 在 FEED 中插入这条消息进入缓存
- 4) 在 STORY 中更新转发数量
- 5) 在 FEEDRELATION 添加 user\_a,feed\_a 与 user\_b,feed\_b 的转发关系

#### 4. A 评论 B 的故事

- 1) 在 STORY 中添加这条原消息, type=1
- 2) 更新 USER 的 feedsCount+1
- 3) 在 FEED 中插入这条消息进入缓存
- 4) 在 STORY 中更新评论数量
- 5) 在 FEEDRELATION 添加 user\_a,feed\_a 与 user\_b,feed\_b 的转发关系
- 6) 在 FEEDRELATION 中添加评论信息

#### 5.删除故事

- 1) 删除 STORY 原数据 story\_a
- 2) 删除对应的 FEED

3) 备注：如果 A 的 STORY 被别人评论或者引用，那么在对方查看评论或者转发的时候会提示“原消息已被作者删除”

#### 6.删除转发/评论的故事

- 1) 删除 STORY 原数据 story\_a
- 2) 删除对应的 FEED
- 3) 在 FEEDRELATION 找到转发关系并删除
- 4) 删除对应的 COMMENT
- 5) 故事中的转发-1 或评论-1

#### 7.拉取消息

- 1) 从 FEED 中拉取 USERID, FEEDID, TimeL 索引, 并以 TimeL 排序