



# 数据库开发规范文档

## 1.3

文档修改历史记录

日 期	版本号	更改人	更改内容
07-10-2012	1.2	陶卫	创建文档
12-09-2012	1.3	钟格致	更新

注：所有 SQL 皆需要 DBA 审核

## 建表

1. 库名、表名、列名统一小写，字符集 UTF8，分割符号统一下划线\_
2. 库名、表名、列名命名根据相关英文单词，禁止拼音，表及列必须加备注
3. 默认所有表存储引擎统一用 innodb
4. 所有表必须有个自增长 ID，非负加 UNSIGNED
5. 所有表尽量禁止物理删除(delete)，如有特殊需求
6. 时间统一 datetime 日期 DATE 年 YEAR, int 统一 int(11), 范围在 1970-01-01 00:00:00'-2037 年，用 TIMESTAMP，状态类字段统一 tinyint，长度统一 tinyint(3)，不要用 ENUM，ENUM 有三个问题：增减新值要 DDL，默认值(非法值插入 ENUM 以空字符串作为特殊错误值)，索引值(插入数字实际是插入索引对应的值)
7. 所有表关联采用 int 型 ID，禁止字符串的 code。比如：订单相关表统一存储订单 ID(goform\_id),而不是订单 code(goform\_code)
8. 统一长度的字符串采用 char，不要使用 varchar。例如：MD5，password 等函数生成的定长字符串
9. varchar char 的长度保持跟前端统一，比如：名称前段只允许 8 个字符，后端就是 varchar(8)，8 表示是字符数而不是字节数，因为 varchar(n)申请内存时候会使用 N 的长度，而不是实际长度
10. 精确浮点数必须使用 DECIMAL，而不是 FLOAT 和 DOUBLE，比如：金额类字段
11. 尽量不使用 TEXT、BLOB 类型，如有特殊需求 单独用一张表存储
12. 建议所有字段定义为 NOT NULL，不确定列给默认值
13. 禁止使用 VARBINARY、BLOB 存储图片文件等
14. 列过多拆分成多张表，比如：订单主表，订单细表。分表根据访问频繁度拆分，最频繁访问的字段放在一张表内
15. 尽量一次性把所有需要列考虑完整，避免后期的 DDL 造成大量锁表
16. 建表时尽量考虑后续 SQL，尽量做到对于频繁的 SQL 一次性，一张表，获取到所需数据

## 索引

- 1.命名统一小写,分割符号统一下划线\_ , 可以唯一的加唯一索引。唯一索引以 `uniq_` 开头, 非唯一所有以 `idx_` 开头
2. 单表索引数量控制在 5 个以内
3. 长度过长的 `VARCHAR` 字段建立索引时, 添加 `crc32` 或者 `MD5 Hash` 字段, 对 `Hash` 字段建立索引
4. 合理创建联合索引,避免冗余, `(a,b,c)` 相当于 `(a)` 、 `(a,b)` 、 `(a,b,c)`
5. `distinct` 值小于 `count` 很多的列不要建索引, 比如: 性别等状态类字段

## SQL

1. 所有 SQL 中不允许出现 `*` 只取必要列, 禁止 `group by order by 1` , 必须显示的给出列名
2. 禁止写存储过程, 函数, 视图
3. `UPDATE`、`DELETE` 语句不使用 `LIMIT` , 避免误操作
4. `where` 中禁止对列加函数, 函数应加在常量上。例如:禁止 `FROM_UNIXTIME(update_time) = 1354169905` , 应该 `update_time = UNIX_TIMESTAMP(1354169905)`
5. 禁止 `where in` 太多内容
6. `INSERT` 使用 `batch` 提交 (`INSERT INTO table VALUES(),(),().....`), `values` 的个数不能过多
7. 不使用 `ORDER BY RAND()`
8. 使用 `COUNT(*)`, 而不是 `COUNT(primary_key)` 或 `COUNT(1)`
9. 直接重要的接口, 可使用 `prepared statement`, 避免 SQL 注入
10. 尽量避免模糊查询
11. 避免对 `innodb` 大数据量表的频繁 `count(*)`操作
12. 对于大表分页查询 `SELECT * FROM table ORDER BY TIME DESC LIMIT 10000,10`; 此种分页方式会导致大量的 `io`, 因为 `MySQL` 使用的是提前读取策略, 读取 100010 行再丢弃 10000 行。推荐分页方式: `SELECT xxx FROM table WHERE TIME<last_TIME ORDER BY TIME DESC LIMIT 10` # `last_TIME` 有索引  
`SELECT xxx FROM table JOIN(SELECT id FROM table ORDER BY TIME LIMIT 10000,10) as t`

USING(id) # 是自增长主键

## 其他

1.推广活动，提前通知 DBA 和运维 进行评估。