

wanfangdata / guide

Q

Type 7 to search

<> Code

Issues 1

Pull requests 1

Actions

Projects

Security

Insights

master

guide / 数据库规范 / Mysql数据库结构设计及开发规范.md

Q

Go to file

t

...

Lvyn0112

添加MySQL数据库规范文档的目录

422541e · 9 years ago

History

Preview

Code

Blame

194 lines (166 loc) · 6.38 KB

Raw

#MySQL数据库结构设计规范

1.适用范围

2.知识体系与参考文档

3.一般要求

• 3.1.理解需求，合理设计

• 3.2.命名规范、一致

4.数据库结构设计规范

• 4.1.数据库

• 4.2.引擎及版本选择

• 4.3.用户

• 4.4.表

• 4.5.索引

• 4.6.存储过程/函数/触发器

• 4.7.约束

• 4.8.视图

• 4.9.表结构变更

• 4.10.库名、表名、字段名必须使用小写字母

• 4.11.尽可能不使用TEXT、BLOB类型

• 4.12.表字符集选择UTF8

• [4.13.建议字段定义为NOT NULL](#413建议字段定义为NOT NULL)

• 4.14.EXPLAIN语句

• 4.15.避免MySQL进行隐式类型转化

• 4.16.MySQL中分页

1.适用范围

规范数据库逻辑模型与物理模型的设计，提高系统可维护性和可扩展性，避免性能问题。 适用对象与范围：本规范适用于数据库设计人员，要求设计人员理解关系数据库的基本概念，掌握MySQL数据库的基本开发技能。

2.知识体系与参考文档

开发设计人员应具备的知识体系：

• 数据库概念与体系结构

• 数据库逻辑与物理模型的设计方法

• 数据库开发技能

• 数据库应用优化技能

• 常见数据库系统故障分析方法

3.一般要求

良好的数据库结构设计在满足功能需求外，还应满足非功能需求，体现在系统的可维护性、扩展性，保证数据质量，消除系统性能瓶颈。

3.1.理解需求，合理设计

深入理解需求，需要关注：

• 实体基本属性

• 实体关系

• 唯一性

• 列是否可空

• 数据量

• 数据清理逻辑

- 数据访问方式（INSERT/UPDATE/DELETE/SELECT分别以哪些列为条件进行查询）
- 数据访问频率（INSERT/UPDATE/DELETE/SELECT）
- 关联关系

3.2.命名规范、一致

系统内对象的命名应该规范、一致，包括但不限于以下对象：

- 表
- 列
- 索引
- 存储过程/函数/触发器
- 约束
- 临时表
- 视图

4.数据库结构设计规范

4.1.数据库

数据库名称使用含有业务含义的英文小写字母，“\_”分割，数据库结构设计工具建议使用PowerDesigner。

4.2.引擎及版本选择

存储引擎根据实际的场景，结合各存储引擎的特点，选择合适的使用，如需要支持事务，选择Innodb，如cluster版本，则选择Ndbcluster。根据目前版本的稳定性，建议使用MySQL5.6及以上版本。

4.3.用户

每个用户模式一般表示一个子系统，使用子系统英文小写进行命名，“\_”分割。

4.4.表

表命名规则：

- 表名称应简洁，准确反映表内数据的业务含义
- 交易表名称以trans\_开头
- 日志表名称以\_log结尾
- 临时表名称应以tmp开头，如tmp\_abc
- 备份表名称应以bak开头，如bak\_abc
- 不能使用系统保留字

列命名规则：

- 列名称应简洁，准确反映列内数据的业务含义
- 应参考已存在同样业务含义的列进行命名
- 不能使用系统保留字

表及字段设计建议：

- 便于分布式设计，允许适度冗余，为了容量扩展允许适度开销
- 数据保留周期，历史数据清理逻辑
- 整形定义中不添加长度，比如使用INT，而不是INT(4)
- 使用短数据类型，比如取值范围为0-80时，使用TINYINT UNSIGNED
- 尽可能不使用TEXT、BLOB类型
- VARCHAR(N)，N表示的是字符数不是字节数，比如VARCHAR(255)，可以最大可存储255个汉字，需要根据实际的宽度来选择N
- 存储年使用YEAR类型。存储日期使用DATE类型。 存储时间（精确到秒）使用TIMESTAMP类型或INT
- 禁止在数据库中存储明文密码
- 表字符集选择UTF8
- 禁止在数据库中使用VARBINARY、BLOB存储图片、文件等
- 除应用程序不访问的日志表外，均应创建主键

注释规则：

- 表、字段都要有注释，说明业务含义
- 对于枚举型列，应在注释中列举。

4.5.索引

命名规则：

- 应包含表名、列名
- 普通索引命名格式为idx\_tabname\_col1
- 唯一索引命名格式为uidx\_tabname\_col1
- 主键命名格式为pk\_tabname\_col1
- 位图索引命名格式为bidx\_tabname\_col1

创建索引时，应考虑索引存储与维护成本，关注：

- 创建索引目的明确（实现唯一约束，加快查询、排序速度）
- 单张表的索引数量控制在5个以内
- 唯一键不和主键重复
- UPDATE、DELETE语句需要根据where条件添加索引
- 使用EXPLAIN判断SQL语句是否合理使用索引，尽量避免extra列出现：Using File Sort，Using Temporary。
- 不建议使用%前缀模糊查询，例如LIKE “%weibo”, 会导致全表扫描

#### 4.6.存储过程/函数/触发器

命名规则：

- 函数命名以f\_开头，后接函数的功能
- 存储过程以prc\_开头，后接功能描述
- 触发器名称格式，tri\_开头，后接功能描述

建议：

- 合理利用自治事务
- 合理利用调用者权限和使用者权限
- 避免存储过程之间存在复杂依赖关系
- 一般情况下，不得使用触发器

#### 4.7.约束

命名规则：

- 应包含表名、列名
- 检查约束命名规则，chk\_tabname\_col1\_col2
- 主键应以pk\_tabname
- 外键格式应为fk\_tabname\_col1\_r\_tab2

关注：

- 对于需要确保数据一致性的表，创建约束（非空、唯一、外键、检查）
- 针对外键约束，如果父表的记录需要修改或删除，子表必须创建索引，避免死锁
- 合理利用约束，提高数据质量

#### 4.8.视图

命名规则：v\_表名

规则：

- 应明确指定列，不应使用SELECT \* 方式定义视图
- 涉及多表时，需要使用表别名限定列所属表

#### 4.9.表结构变更

表结构变更这里主要指增加列、删除列、调整类型，不包括存储属性的变更。 针对表结构变更，关注：

- 新增列是否有缺省值（如有缺省值，需要考虑时间窗口）
- 是否需要数据进行迁移
- 主键/索引是否需要调整
- 应用调整
- 对其他业务领域的影响（原则上要求不能影响其他业务条线）

#### 4.10.库名、表名、字段名必须使用小写字母

- MySQL有配置参数lower\_case\_table\_names，不可动态更改，Linux系统默认为0，即库表名以实际情况存储，大小写敏感。如果是1，以小写存储，大小写不敏感。如果是2，以实际情况存储，但以小写比较
- 如果大小写混合用，可能存在abc,Abc,ABC等多个表共存，容易导致混乱
- 字段名显示区分大小写，但实际使用不区分，即不可以建立两个名字一样但大小写不一样的字段
- 为了统一规范， 库名、表名、字段名使用小写字母



#### 4.11.尽可能不使用TEXT、BLOB类型

- 索引排序问题，只能使用max\_sort\_length的长度或者手工指定ORDER BY SUBSTRING(column, length)的长度来排序
- Memory引擎不支持text,blog类型，会在磁盘上生成临时表
- 可能浪费更多的空间
- 导致使用WHERE没有索引的语句变慢

#### 4.12.表字符集选择UTF8

- 使用utf8字符集，如果是汉字，占3个字节，但ASCII码字符还是1个字节
- 统一，不会有转换产生乱码风险
- 其他地区的用户（美国、印度、台湾）无需安装简体中文支持，就能正常看您的文字，并且不会出现乱码

#### 4.13.建议字段定义为NOT NULL

- 如果null字段被索引，需要额外的1字节
- 可用0，"代替
- 如果是索引字段，一定要定义为not null

#### 4.14.EXPLAIN语句

EXPLAIN 语句（在MySQL客户端中执行）可以获得MySQL如何执行SELECT语句的信息。通过对SELECT语句执行EXPLAIN，可以知晓MySQL执行该SELECT语句时是否使用了索引、全表扫描、临时表、排序等信息。尽量避免MySQL进行全表扫描、使用临时表、排序等。

#### 4.15.避免MySQL进行隐式类型转化

因为MySQL进行隐式类型转化之后，可能会将索引字段类型转化成=号右边值的类型，导致使用不到索引，原因和避免在索引字段中使用函数是类似的。

#### 4.16.MySQL中分页

假如有类似下面分页语句：SELECT \* FROM table ORDER BY TIME DESC LIMIT 10000,10; 这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。推荐分页方式：SELECT \* FROM table INNER JOIN(SELECT id FROM table ORDER BY TIME LIMIT 10000,10) as t USING(id);