

PostgreSQL 开发最佳实践

digoal

阿里云

目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

规约的重要性



目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

命名规约

- `quote_nullable` 使用有意义的对象名;
- `quote_ident` 使用小写字母, 下划线, 数字;
- `format (%L , %I)`

```
select c1 as 你好, c2 as 中国, .... from xxx where xxx;
```

```
create table "班级" ("学号" int, "姓名" name, ...);
```

```
comment on table is "市场营销";
```

```
create table xyz (...);
```

```
select c1 as ??, c2 as ??, .... from xxx where xxx;
```

```
comment on table is "????";
```

目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

设计规约

- 返回多行
- `select * from tbl where xxx=? order by yyy desc limit ?`
- 多行聚合，返回一行
- `select * from tbl where xxx=? and yyy=?`

- `select * from tbl where ? between x and y;`
- `select * from tbl where x_y @> ?; -- USE range and GIST idx.`

- `select * from tbl where x=? and y=1; -- use partial index.`

- `select * from tbl where ts_vector && ts_query? order by ? offset ? limit ?;`
- use rum idx instead gin;

设计规约

- `select * from t1 join t2 on (t1.xx=t2.xx);`
- `t1.xx text, t2.xx int`

- `create index idx on tbl (c1);`
- `insert into tbl (c1) values (very long text);`
- `select * from tbl where c1=?; -- hash value -> idx`

- `update tbl set c1=xx; -- 更新非常频繁`
- `HOT, fillfactor`

- 全球化、字符集的选择

- 线性相关、范围查询，索引AM的选择。

- 数据类型的选择。

- **big table** 问题(备份、恢复、创建索引、索引深度)，分区

目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

管理规约

- `begin; set local lock_timeout = '10s'; -- DDL query; end;`
- `begin; explain analyze query; rollback;`
- `create index CONCURRENTLY idx on tbl(id);`
- 密码复杂度、认证方法
- 业务系统，开发测试账号，不要使用数据库超级用户。
- 长连接relcache , syscache
- 数据隔离(cluster -> database -> schema)
- 主备切换(rewarm)

目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

稳定性与性能建议

- 避免长事务、2PC。
- 绑定变量。
- hash index, unlogged table使用注意
- AD LOCK, 秒杀、消除并发数据冲突
- count(*) 是否存在数据? perform limit 1(FOUND)
- 短连接, 连接池
- 频繁创建、删除临时表(10.0 ROADMAP)
- explain评估结果集, instead count(*) (分页数)
- 分页优化cursor **instead** order by offset
- statement_timeout 防止雪崩
- truncate **instead** delete all data

稳定性与性能建议

- ddl in xact;
- ts_vector or array instead string(a,b,c,d,...) . xx like '%a%'
- 递归查询一定要有跳出递归的条件
- 估值计算，HLL代替count(distinct xx)
- 频繁更新的字段与不频繁更新的字段建议拆成多个表，提高更新效率。
- 业务层避免死锁
- 根据业务需求调整表级autovacuum相关参数
- 内存足够时，UDF中尽量使用数组代替临时表
- JOIN的固化、子查询的固化。（plan hint OR query写法+parameter）
- GIN 索引 UPDATE 加速

目录

- 规约的重要性
- 命名规约
- 设计规约
- 管理规约
- 稳定性与性能建议
- 云数据库使用建议

云数据库使用建议

- 冷热分离
- 链路选择
- UDF & RT
- 尽量避免公网访问
- 可用区的选择

GIT

- https://github.com/digoal/blog/blob/master/201609/20160926_01.md
- 36条强制规约
- 78条推荐规约