Atitit 性能原因与优化解决总结

性能问题原因与解决对策

目录

[1. 遗漏的索引 1](#_Toc25667)

[1.1. 全面排查条件索引 1](#_Toc17327)

[1.2. 适当使用强制索引force index 1](#_Toc17107)

[1.3. 增加适当的函数索引机制 2](#_Toc22110)

[1.4. 适当使用不同类型索引bree hash fulltext 2](#_Toc27977)

[2. Join过多 字母表分拆过细导致的查询慢 2](#_Toc26257)

[2.1. join\_buffer\_size配置增大（简单） 2](#_Toc26766)

[2.2. 增加mysql cache系列的配置值 2](#_Toc24907)

[2.3. 1 用小表做驱动表。 2 一定要用上被驱动表的索引 2](#_Toc25193)

[2.4. 物化视图法（简单） 2](#_Toc9916)

[2.5. 内存表模式（简单）） 2](#_Toc9647)

[2.6. 多语句sql，适当利用临时表join 2](#_Toc7234)

[2.7. 增加主表字段，减少join，子母表触发器同步 2](#_Toc6324)

[2.8. 无结构模式 Json字段法 3](#_Toc5001)

[3. 子查询过多 3](#_Toc24700)

[3.1. 使用物化视图或适当字段冗余解决 3](#_Toc22402)

[3.2. 函数索引 3](#_Toc22077)

[4. 初始阶段 表设计考虑性能的设计 3](#_Toc30290)

[4.1. 大力使用json存储 3](#_Toc10804)

[4.2. 匹配ui的冗余字段表 3](#_Toc22367)

[5. 其他 3](#_Toc9689)

[5.1. 自己实现了连接引擎 3](#_Toc26409)

[5.2. Redis 3](#_Toc20722)

[5.3. 固态硬盘 3](#_Toc11925)

# 遗漏的索引

## 全面排查条件索引

## 适当使用强制索引force index

如果遇到走不来索引的话

## 增加适当的函数索引机制

## 适当使用不同类型索引bree hash fulltext

# Join过多 字母表分拆过细导致的查询慢

## join\_buffer\_size配置增大（简单）

## 增加mysql cache系列的配置值

## 1 用小表做驱动表。 2 一定要用上被驱动表的索引

## 物化视图法（简单）

定时刷新走event机制，批量insert from刷入数据

## 内存表模式（简单））

## 多语句sql，适当利用临时表join

## 增加主表字段，减少join，子母表触发器同步

匹配ui与表设计

## 无结构模式 Json字段法

# 子查询过多

## 使用物化视图或适当字段冗余解决

## 函数索引

# 初始阶段 表设计考虑性能的设计

## 大力使用json存储

## 匹配ui的冗余字段表

# 其他

## 自己实现了连接引擎

## Redis

## 固态硬盘