



链滴

性能优化（一）初识性能优化

作者：Heayan

原文链接：<https://ld246.com/article/1520930606801>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

前言

性能问题是在项目最后阶段，必须要面对的核心问题。性能优化工作，不会出现在项目前期、中期阶段。在这些阶段，项目还没有稳定下来，进行项目优化并没有有效的切入点、侧重点，也不能确定哪些地方需要优化，哪些地方不需要优化。如果产品设计的合理变动，可能导致前期优化工作的浪费，从而导致重复性工作。所以，性能优化一般放在项目最后阶段，由测试工程师协助发现突出的性能问题，针对性解决。

性能优化一般针对服务器性能、程序性能、数据库性能三个方面进行优化。

程序性能

程序性能优化的核心是，降低代码量、降低IO频率、降低特殊功能开销、降低内存使用。

1. 算法优化。

主要目的就是降低代码量。使用合适的算法，解决工作中问题，能够有效的提升性能。

如：使用快速排序算法代替冒泡排序算法，来降低程序执行时间。

2. 降低SQL使用频率。

主要目的是，降低SQL使用量，尽量避免在循环中使用SQL，避免复杂SQL出现。因为每次SQL执行需要IO、SQL解析、SQL执行，SQL使用频次会增加IO和SQL解析时间，而这两个方面耗时是非常巨大的。平均每次SQL执行需要3ms，循环使用SQL耗时会成倍增加。

3. 异常优化。

异常的开销很大，不要有空异常驳回，避免使用异常处理业务异常。

4. 线程优化。

线程的创建销毁开销很大，尽量避免频繁创建销毁线程。如：Java提供的线程池，可以重复利用工作。但缺点是，任务线程依然在频繁创建销毁；优点是，Java提供的线程池具有通用性，把业务和线隔离开来，管理方便。如果线程已经影响到性能，最好封装针对业务的线程池。

5. 缓存使用。

使用缓存来保存部分数据，减少数据库访问，从而调高性能。缓存分为应用缓存，在系统内部实现；布式缓存，通过Memcached、Redis实现。

数据库性能

数据库优化的核心是，减低IO频率、避免复杂SQL、降低特殊功能开销、降低内存使用。因为数据库时主要体现在IO传输、SQL解析、SQL执行。

SQL优化

1. 判断条件尽量在SQL语句中提现。

Oracle解析引擎，在解析筛选条件时，先从最后一个条件开始，所以尽量将可以过滤大量数据的条件在where最后。

2. 返回结果集尽量小

3. 尽量减少级联

索引

在百万级、千万级及以下数据量时，索引的使用，还是能够非常好的提升性能。索引特点是以空间换时间，过多的索引会增加INSERT、DELETE、UPDATE的负担，不必要的索引尽量删除，存在空值的段尽量不要创建索引。

特别注意，SQL使用时，在下面条件下会导致索引失效。可以通过执行计划，来查看SQL执行时，索引是否生效。

以下情况容易导致引擎放弃使用索引而进行全表扫描：

- 对索引字段进行函数操作、算术运算或其他表达式运算
- 在索引列上出现数据类型转换
- 在索引列上使用<>、!=
- 在索引列上使用like '%%'(不能前置百分号)
- B-tree索引is null不会走 is not null 会走；位图索引is null,is not null都会走

服务器方面

暂无

常见问题

1. Oracle中in超过1000报错

可以使用in or in...解决，也可以通过临时表（推荐）解决。

2. Oracle实现树的遍历

```
select * from BD_SPFL start with bm = '4' connect by sjbm = prior bm;
```

3. 很多时候用 exists 代替 in 是一个好的选择

```
select num from a where num in(select num from b)
```

用下面的语句替换：

```
select num from a where exists(select 1 from b where num=a.num)
```

版权所有，转载必须标识原文地址