

缓存的作用

2021年10月6日 23:19

为什么需要缓存

计算机在不同层次上处理时间是不一样的

数据传输时间

1. 在计算机中，最快的是一级缓存和二级缓存

读取CPU的一级缓存只需要0.5ns，如果其中包含了转移和分支预测，则需要5ns

读取CPU的二级缓存需要7ns

CPU的一级缓存和二级缓存是最快的部分，也是容量最小的部分，我们一般不会把数据存在一级二级缓存中。一二级缓存中主要存储的是代码，以及当前正在计算的数据

2. 数据主要存储在内存中

内存中读取1MB数据，大约需要25,000ns

内存也是有限的，更多数据是放在网络上或硬盘中

3. 在1G的网络（千兆网）上读取/发送1MB的数据，大约需1000,000ns

4. SSD磁盘读取速度较快，但写、擦除等寿命有限

机械磁盘读取速度较慢，但寿命较长

所以我们存储数据更多使用的是机械磁盘

机械磁盘的时间主要分为磁盘的寻道时间、磁盘的读出写入时间

机械磁盘的寻道时间，大约为1000,000ns

从机械磁盘读出1MB数据，大约需要3000,000ns

数据库通常是部署在一台独立的服务器上的，需要通过网络去访问，而它又包含大量的磁盘操作

当整个系统的访问量较大时，数据库往往成为系统的瓶颈

我们可以采用加大最大连接数、并发数等来提升性能，但提升有上限，最终还要受到硬件条件的限制

所以当单台数据库服务器达到瓶颈时，我们会采用高并发、负载均衡、分布式数据等技术来把单台数据库服务器变成多台数据库服务器，或者进行分库分表的操作，但这些都会消耗人力财力，增加成本

所以我们往往采用**缓存技术**在上述技术之前来减轻数据库的负载

分层缓存

我们的系统通常分为以下几个部分：

1. 前端客户端
基于网页/基于APP/基于微信小程序
2. Web服务器（接入服务器）
基于网页的通常会有一个Web服务器，一些静态的数据、网页、图片会存储在Web服务器上
3. 应用服务器
用于运行Java程序，执行业务逻辑
4. 数据库服务器

我们在上述的每一层都可以做一些缓存，为后面的服务器去减轻压力：

1. 前端服务器
基于网页的可以用JavaScript把一些常用的只读数据存储在jsp的对象中，这样就不需要频繁的向后端发起请求，去读取数据
基于APP的，一些本身就是用java写的，可以把一些只读数据存储在前端
2. Web服务器
我们通常用**NGINX服务器**或者**Vanish服务器**来做Web服务器
其中NGINX服务器主要是用来处理静态资源的，也可以把我们访问的数据当做静态资源存储在服务器的磁盘上，我们称之为静态缓存
Vanish服务器与NGINX服务器大致类似，不同在于vanish的缓存不是存在磁盘上，而是存储在内存中，所以Vanish做缓存会比NGINX快（内存有限，Vanish会频繁进行导入导出）
我们往往会**用NGINX去处理静态资源，用Vanish去处理动态数据**
在Web服务器上的缓存通常是一些比较固定的信息，如枚举值、前端状态值
3. 应用服务器
应用服务器是做缓存的主力，应用服务器做缓存主要有以下三种方式：
 - **应用服务器本身**可以做缓存，如Spring框架中可以做一个Singleton Bean，其中存储了一些数据，之后要访问的时候可以直接用
 - **MyBatis、Hibernate**提供的一级二级缓存
 - **Redis分布式缓存技术**：
在有多台应用服务器的情况下，每台应用服务器都有自己的缓存，这些缓存占据各个应用服务器的内存，并且这些缓存如果要更新，会很不方便
Redis的分布缓存技术会为所有的应用服务器提供一个统一的缓存服务器，缓存的数据会统一的放到Redis服务器上去，这个是用内存和网络去访问的

缓存的依据

1. **只读的数据、很少改动**的数据，会尽量**往前端放**
即会把这些只读性较强的数据放在前端的缓存上，或者放到Web服务器的缓存中
越往前靠的缓存，对系统整体性能的提升越明显

2. 越是**可能修改**的数据，尽量**往后放**
如Redis缓存放一些有可能会被修改的数据
3. 频繁修改的数据，我们**不做缓存**，只能依靠数据库进行处理