操作系统级别对Hadoop性能优化

由于Hadoop自身的一些特点，它只适合用于将Linux作为操作系统的生产环境。在实际应用场景中，管理员适当对Linux内核参数进行调优，可在一定程度上提高作业的运行效率，比较有用的调整选项如下。

一、增大同时打开的文件描述符和网络连接上限

在Hadoop集群中，由于涉及的作业和任务数目非常多，对于某个节点，由于操作系统内核在文件描述符和网络连接数目等方面的限制，大量的文件读写操作和网络连接可能导致作业运行失败，因此，管理员在启动Hadoop集群时，应使用ulimit命令将允许同时打开的文件描述符数目上限增大至一个合适的值，同时调整内核参数net.core.somaxconn至一个足够大的值。

此外，Hadoop RPC采用了epoll作为高并发库，如果你使用的Linux内核版本在2.6.28以上，你需要适当调整epoll的文件描述符上限。

二、关闭swap分区

在Linux中，如果一个进程的内存空间不足，那么，它会将内存中的部分数据暂时写到磁盘上，当需要时，再将磁盘上的数据动态置换到内存中，通常而言，这种行为会大大降低进程的执行效率。在MapReduce分布式计算环境中，用户完全可以通过控制每个作业处理的数据量和每个任务运行过程中用到的各种缓冲区大小，避免使用swap分区。

具体方法是调整/etc/sysctl.conf文件中的vm.swappiness参数。vm.swappiness有效范围是0~100，值越高表明内核应该更积极将应用程序的数据交换到磁盘，较低的值表示将延迟这种行为，而不是强制丢弃文件系统的缓冲区。

三、设置合理的预读取缓冲区大小

磁盘I/O性能的发展远远滞后于CPU和内存，因而成为现代计算机系统的一个主要瓶颈。预读可以有效地减少磁盘的寻道次数和应用程序的I/O等待时间，是改进磁盘读I/O性能的重要优化手段之一。管理员可使用Linux命令blockdev设置预读取缓冲区的大小，以提高Hadoop中大文件顺序读的性能。当然，也可以只为Hadoop系统本身增加预读缓冲区大小。

四、文件系统选择与配置

Hadoop的I/O性能很大程度上依赖于Linux本地文件系统的读写性能。Linux中有多种文件系统可供选择，比如ext3和ext4，不同的文件系统性能有一定的差别。如果公司内部有自主研发的更高效的文件系统，也鼓励使用。

在Linux文件系统中，当未启用noatime属性时，每个文件读操作会触发一个额外的文件写操作以记录文件最近访问时间。该日志操作可通过将其添加到mount属性中避免。

五、I/O调度器选择

主流的Linux发行版自带了很多可供选择的I/O调度器。在数据密集型应用中，不同的I/O调度器性能表现差别较大，管理员可根据自己的应用特点启用最合适的I/O调度器。

六、vm.overcommit_memory设置

进程通常调用malloc()函数来分配内存，内存决定是否有足够的可用内存，并允许或拒绝内存分配的请求。Linux支持超量分配内存，以允许分配比可用RAM加上交换内存的请求。

vm.overcommit_memory参数有三种可能的配置：

0 表示检查是否有足够的内存可用，如果是，允许分配；如果内存不够，拒绝该请求，并返回一个错误给应用程序。

1 表示根据vm.overcommit_ratio定义的值，允许分配超出物理内存加上交换内存的请求。vm.overcommit_ratio参数是一个百分比，加上内存量决定内存可以超量分配多少内存。例如，vm.overcommit_ratio值为50，而内存有1GB，那么这意味着在内存分配请求失败前，加上交换内存，内存将允许高达1.5GB的内存分配请求。

2 表示内核总是返回true。

除了以上几个常见的Linux内核调优方法外，还有一些其他的方法，管理员可根据需要进行适当调整。