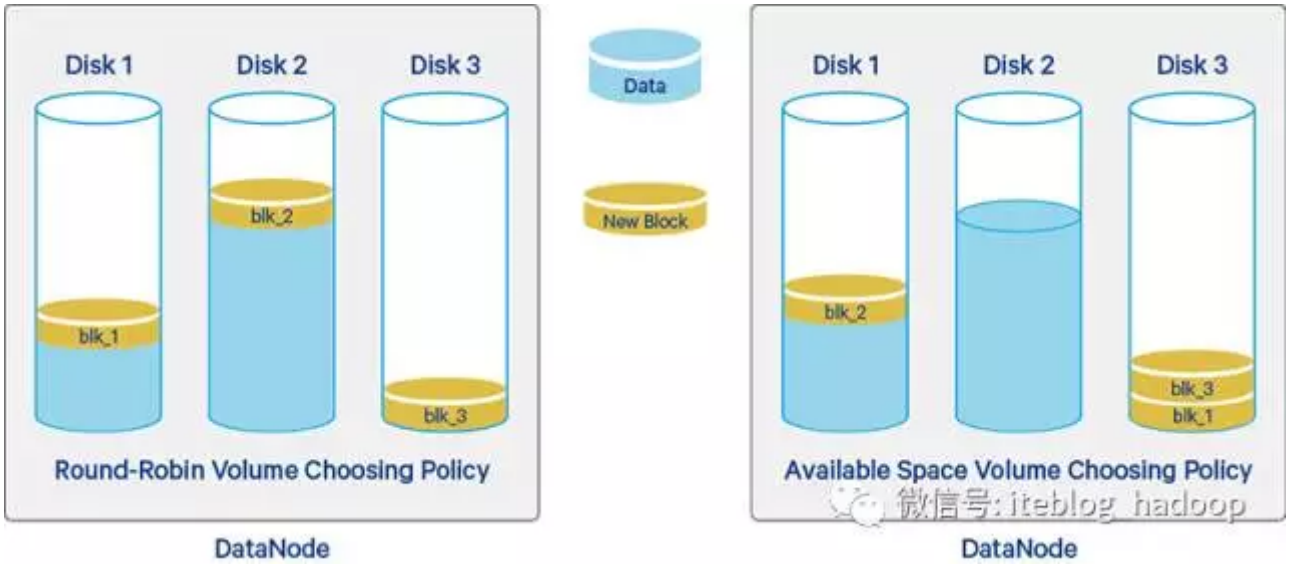
Hadoop 3.0磁盘均衡器(diskbalancer)新功能及使用介绍

原创 2017-09-26 iteblog Hadoop技术博文

在HDFS中，DataNode 将数据块存储到本地文件系统目录中，具体的目录可以通过配置 `hdfs-site.xml` 里面的 `dfs.datanode.data.dir` 参数。在典型的安装配置中，一般都会配置多个目录，并且把这些目录分别配置到不同的设备上，比如分别配置到不同的HDD（HDD的全称是Hard Disk Drive）和SSD（全称Solid State Drives，就是我们熟悉的固态硬盘）上。

当我们往HDFS上写入新的数据块，DataNode 将会使用volume选择策略来为这个块选择存储的地方。目前Hadoop支持两种volume选择策略：round-robin 和 available space（详情参见：HDFS-1804），我们可以通过 `dfs.datanode.fsdataset.volume.choosing.policy` 参数来设置。

循环（round-robin）策略将新块均匀分布在可用磁盘上；而可用空间（ available-space ）策略优先将数据写入具有最大可用空间的磁盘（通过百分比计算的）。正如下图所示：



如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：**iteblog_hadoop**

默认情况下，DataNode 是使用基于round-robin策略来写入新的数据块。然而在一个长时间运行的集群中，由于HDFS中的大规模文件删除或者通过往DataNode 中添加新的磁盘仍然会导致同一个DataNode中的不同磁盘存储的数据很不均衡。即使你使用的是基于可用空间的策略，卷（volume）不平衡仍可导致较低效率的磁盘I/O。比如所有新增的数据块都会往新增的磁盘上写，在此期间，其他的磁盘会处于空闲状态，这样新的磁盘将会是整个系统的瓶颈。

最近，Apache Hadoop community开发了好几个离线的脚本（可以参见 HDFS-1312 或者 hadoop-balancer ）以缓解数据不平衡问题。然而这些脚本都是在HDFS代码库之外，在执行这些脚本往不同磁盘之间移动数据的时候，需要要求DataNode处于关闭状态。结果，HDFS-1312 还引入了一个在线磁盘均衡器，旨在根据各种指标重新平衡正在运行DataNode上的磁盘数据。和现有的HDFS均衡器类似，HDFS 磁盘均衡器在DataNode中以线程的形式运行，并在相同

存储类型的卷（volumes）之间移动数据。我们要注意，本文介绍的HDFS 磁盘均衡器是在同一个DataNode中的不同磁盘之间移动数据，而之前的HDFS均衡器是在不同的DataNode之间移动数据。

在下面的文章中，我将介绍如何使用这个新功能。

让我们通过一个例子逐步探讨这个有用的功能。首先，确保所有DataNode上的 dfs.disk.balancer.enabled 参数设置成true。本例子中，我们的DataNode已经挂载了一个磁盘（ /mnt/disk1 ），现在我们往这个DataNode上挂载新的磁盘（ /mnt/disk2 ），我们使用df命令来显示磁盘的使用率：

```
# df -h
….
/var/disk1    5.8G    3.6G    1.9G    66% /mnt/disk1
/var/disk2    5.8G      13M    5.5G     1% /mnt/disk2
```

从上面的输出可以看出，两个磁盘的使用率很不均衡，所以我们来将这两个磁盘的数据均衡一下。

典型的磁盘平衡器任务涉及三个步骤（通过HDFS的diskbalancer 命令）：plan, execute 和 query。第一步，HDFS客户端从NameNode上读取指定DataNode的的必要信息以生成执行计划：

```
# hdfs diskbalancer -plan lei-dn-3.example.org
16/08/19 18:04:01 INFO planner.GreedyPlanner: Starting plan for Node : lei-dn-3.exam
ple.org:20001
16/08/19 18:04:01 INFO planner.GreedyPlanner: Disk Volume set 03922eb1-63af-4a16-baf
e-fde772aee2fa Type : DISK plan completed.
16/08/19 18:04:01 INFO planner.GreedyPlanner: Compute Plan for Node : lei-dn-3.examp
le.org:20001 took 5 ms
16/08/19 18:04:01 INFO command.Command: Writing plan to : /system/diskbalancer/2016-
Aug-19-18-04-01
```

从上面的输出可以看出，HDFS磁盘平衡器通过使用DataNode报告给NameNode的磁盘使用信息并结合计划程序来计算指定DataNode上数据移动计划的步骤，每个步骤指定要移动数据的源卷和目标卷，以及预计移动的数据量。

截止到撰写本文的时候，HDFS仅仅支持 GreedyPlanner ，其不断地将数据从最常用的设备移动到最少使用的设备，直到所有数据均匀地分布在所有设备上。用户还可以在使用 plan 命令的时候指定空间利用阀值，也就是说，如果空间利用率的差异低于此阀值，planner 则认为此磁盘已经达到了平衡。当然，我们还可以通过使用 --bandwidth 参数来限制磁盘数据移动时的I/O。

磁盘平衡执行计划生成的文件内容格式是Json的，并且存储在HDFS之上。在默认情况下，这些文件是存储在 /system/diskbalancer 目录下面：

```
# hdfs dfs -ls /system/diskbalancer/2016-Aug-19-18-04-01
Found 2 items
-rw-r--r--   3 hdfs supergroup          1955 2016-08-19 18:04 /system/diskbala
ncer/2016-Aug-19-18-04-01/lei-dn-3.example.org.before.json
-rw-r--r--   3 hdfs supergroup           908 2016-08-19 18:04 /system/diskbal
ancer/2016-Aug-19-18-04-01/lei-dn-3.example.org.plan.json
```

可以通过下面的命令在DataNode上执行这个生成的计划：

```
$ hdfs diskbalancer -execute /system/diskbalancer/2016-Aug-17-17-03-56/172.26.10.1
6.plan.json
16/08/17 17:22:08 INFO command.Command: Executing "execute plan" command
```

这个命令将JSON里面的计划提交给DataNode，而DataNode会启动一个名为BlockMover的线程中执行这个计划。我们可以使用 query 命令来查询DataNode上diskbalancer任务的状态：

```
# hdfs diskbalancer -query lei-dn-3:20001
16/08/19 21:08:04 INFO command.Command: Executing "query plan" command.
Plan File: /system/diskbalancer/2016-Aug-19-18-04-01/lei-dn-3.example.org.plan.json
Plan ID: ff735b410579b2bbe15352a14bf001396f22344f7ed5fe24481ac133ce6de65fe5d721e223
b08a861245be033a82469d2ce943aac84d9a111b542e6c63b40e75
Result: PLAN_DONE
```

上面结果输出的 PLAN_DONE 表示disk-balancing task已经执行完成。为了验证磁盘平衡器的有效性，我们可以使用df -h 命令来查看各个磁盘的空间使用率：

```
# df -h
Filesystem           Size   Used Avail Use% Mounted on
....
/var/disk1    5.8G   2.1G   3.5G   37% /mnt/disk1
/var/disk2    5.8G   1.6G   4.0G   29% /mnt/disk2
```

上面的结果证明，磁盘平衡器成功地将 /var/disk1 和 /var/disk2 空间使用率的差异降低到10%以下，说明任务完成！