

心累的时候不妨停下来休息一下，好好收拾一下情绪在继续前进

# Federation架构设计

## 1. NameNode架构的局限性

### (1) Namespace（命名空间）的限制

由于NameNode在内存中存储所有的元数据（metadata），因此单个NameNode所能存储的对象（文件+块）数目受到NameNode所在JVM的heap size的限制。50G的heap能够存储20亿（200million）个对象，这20亿个对象支持4000个DataNode，12PB的存储（假设文件平均大小为40MB）。随着数据的飞速增长，存储的需求也随之增长。单个DataNode从4T增长到36T，集群的尺寸增长到8000个DataNode。存储的需求从12PB增长到大于100PB。

### (2) 隔离问题

由于HDFS仅有一个NameNode，无法隔离各个程序，因此HDFS上的一个实验程序就很有可能影响整个HDFS上运行的程序。

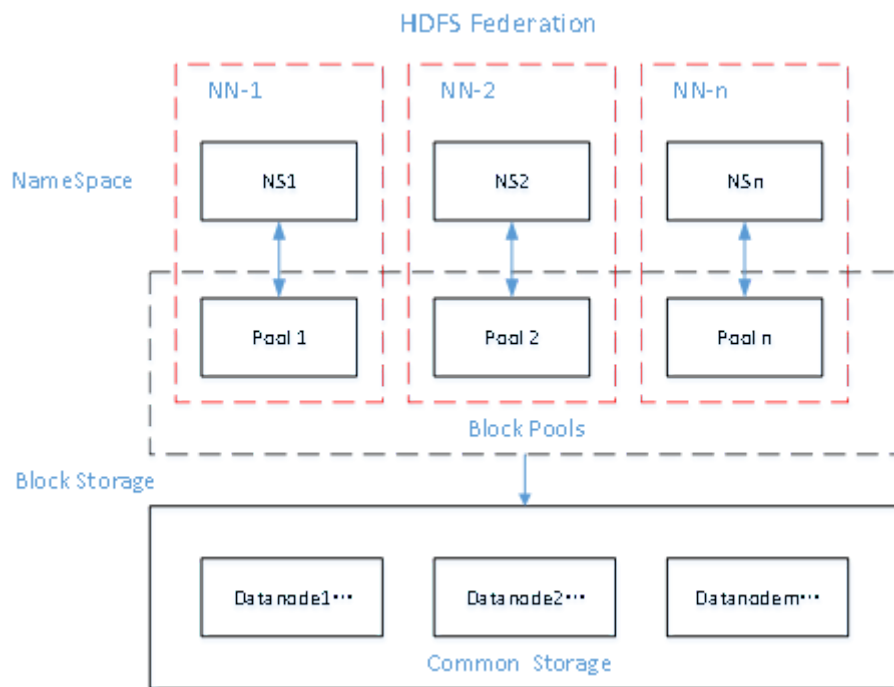
### (3) 性能的瓶颈

由于是单个NameNode的HDFS架构，因此整个HDFS文件系统的吞吐量受限于单个NameNode的吞吐量。

## 2.HDFS Federation架构设计

能不能有多个NameNode

| NameNode | NameNode | NameNode  |
|----------|----------|-----------|
| 元数据      | 元数据      | 元数据       |
| Log      | machine  | 电商数据/话单数据 |



### 3.HDFS Federation应用思考

不同应用可以使用不同NameNode进行数据管理，图片业务、爬虫业务、日志审计业务

Hadoop生态系统中，不同的框架使用不同的NameNode进行管理NameSpace。（隔离性）

但是呢，仅限于超大型公司，超大型数据量使用，中小型公司不用考虑这样的架构，毕竟服务器开销很大