2.x版本中，HDFS架构解决了单点故障问题，即引入双NameNode架构，同时借助共享存储系统来进行元数据的同步，共享存储系统类型一般有几类，如：Shared NAS+NFS、BookKeeper、BackupNode 和 Quorum Journal Manager(QJM)

Hadoop 2.x元数据

Hadoop的元数据主要作用是维护HDFS文件系统中文件和目录相关信息。元数据的存储形式主要有3类：**内存镜像、磁盘镜像(FSImage)、日志(EditLog)。**

在Namenode启动时，会加载磁盘镜像到内存中以进行元数据的管理，存储在NameNode内存；
磁盘镜像是某一时刻HDFS的元数据信息的快照，包含所有相关Datanode节点文件块映射关系和命名空间(Namespace)信息，存储在NameNode本地文件系统；
日志文件记录client发起的每一次操作信息，即保存所有对文件系统的修改操作，用于定期和磁盘镜像合并成最新镜像，保证NameNode元数据信息的完整，存储在NameNode本地和共享存储系统(QJM)中

EditLog文件有两种状态： **inprocess和finalized**, inprocess表示正在写的日志文件，文件名形式:edits*inprocess*[start-txid]，finalized表示已经写完的日志文件,文件名形式：edits*[start-txid]*[end-txid]； FSImage文件也有两种状态, **finalized和checkpoint**， finalized表示已经持久化磁盘的文件，文件名形式: fsimage_[end-txid], checkpoint表示合并中的fsimage, 2.x版本checkpoint过程在Standby Namenode(SNN)上进行，SNN会定期将本地FSImage和从QJM上拉回的ANN的EditLog进行合并，合并完后再通过RPC传回ANN。

```
data/hbase/runtime/namespace
├── current
│   ├── VERSION
│   ├── edits_0000000003619794209-0000000003619813881
│   ├── edits_0000000003619813882-0000000003619831665
│   ├── edits_0000000003619831666-0000000003619852153
│   ├── edits_0000000003619852154-0000000003619871027
│   ├── edits_0000000003619871028-0000000003619880765
│   ├── edits_0000000003619880766-0000000003620060869
│   ├── edits_inprogress_0000000003620060870
│   ├── fsimage_0000000003618370058
│   ├── fsimage_0000000003618370058.md5
│   ├── fsimage_0000000003620060869
│   ├── fsimage_0000000003620060869.md5
│   └── seen_txid
└── in_use.lock
```

上面所示的还有一个很重要的文件就是seen_txid,保存的是一个事务ID，这个事务ID是EditLog最新的一个结束事务id，当NameNode重启时，会顺序遍历从edits_0000000000000000001到seen_txid所记录的txid所在的日志文件，进行元数据恢复，如果该文件丢失或记录的事务ID有问题，会造成数据块信息的丢失。**EidtLog保存最近更新，文件较小修改方便，FsImage包含某一时刻NameNode所有元数据镜像，文件较大，直接操作性能较差。**

参考：

https://www.cnblogs.com/qcloud1001/p/7693476.html