**HDFS的数据完整性**

hdfs会对写入的数据计算校验和（CRC-32），会对读取的数据和对校验和

datanode在收到数据时也会计算校验和

客户端在读取datanode的数据时会计算校验和，用它和datanode的校验和进行比较。如果不正确，会向namenode报告错误和正在读取的datanode。Namenode会将标记这个快出现错误，它安排一个数据快的副本复制到datanode。

LocalFileSystem执行客户端的校验和和验证

LocalFileSystem通过ChecksumFileSystem来完成校验和相关的操作

**压缩**

减少存储文件的压缩空间

加速数据在网络和磁盘中的传输速度

CompressionCodec可以对数据流进行压缩和解压缩

CompressionCodecFactory用于找到文件扩展名对应的codec(不同扩展名需要用不同的codec进行解压,CompressionCodec就是其中的一种)

使用原生类库来实现压缩和解压缩效率比较高

CodecPool支持使用反复压缩和解压缩

在把压缩文件当成MapReduce输入流时，需要考虑压缩格式是否支持分片。一般来说，hdfs会把文件分片，每一个片分配一个map，这些map很可能会并发执行。

可以对MapReduce或map的输出进行压缩，提高传输效率

**序列化**

系列化可以把对象转换成字节流，方便在网络上传输或永久存储在硬盘

反序列化就是把字节流转换成对象的过程

序列化可用于进程间的通信和永久存储

Hadoop中的进程通信是通过PRC协议进行的，PRC协议是通过把消息序列化为二进制流发送到远程节点，远程节点再进行反序列化成原来的消息

Hadoop的Writable可以实现序列化和反序列化

Avro框架是用来序列化和反序列化的，比使用writable的效果更好

**基于文件的数据结构**

SequenceFile

适合处理纯文本

可以做为小文件的处理容器，提高处理效率

MapFile是经过排序的SequenceFile，可以通过键值对查找