（PPT1）

大家好，在本节中，我们将向同学们介绍HDFS的数据读写过程。

（PPT2）

客户端读写文件时，首先要创建文件输入流DFSInputStream和文件输出流DFSOutputStream，当读写结束要关闭这些流。由于HDFS的文件是分块存储到不同的数据节点上，所以客户端要和数据节点交互进行读写数据块操作，当要读数据块时要在这个数据节点上建立一个数据块输入流对象FSDataInputStream，并调用read（），读完要调用close（）方法关闭该数据节点的数据块输入流；当要写数据块时要在这个数据节点上建立一个数据块输出流对象FSDataOutputStream，并调用write（），写完后要调用close（）方法关闭该数据节点的数据块输出流。

客户端读写文件，都要把自己的需求提交给名称节点，名称节点会挑选合适的数据节点介绍给客户端来读或写数据块。客户端和名称节点的交互是通过ClientProtocol协议接口提供的getBlockLocation（）方法，从名称节点获取文件的每个数据块的每个副本所在的数据节点地址，返回给客户端。

（PPT3）

和普通文件读操作一样，读操作分为三部曲进行，open（）创建文件输入流DFSInputStream、read（）读文件、close（） 关闭文件输入流。

由于HDFS的文件是分块存储到不同的数据节点上，因此读文件又分为三步：与某个数据节点建立数据块输入流FSDataInputStream，然后read（）数据块，最后close（）该数据节点上的输入流。

具体过程如下：

1、客户端通过FileSystem.open()，获取要读的这个文件对应的输入流DFSInputStream。

2、DFSInputStream通过ClientProtocol协议远程调用名称节点，获得中此文件对应的数据块保存位置，包括每个数据块副本的所在数据节点的地址，同时根据距离客户端的远近对数据节点进行排序。为了和数据节点进行交互，实例化了FSDataInputStream，返回给客户端，同时返回数据块的数据节点地址。

3、客户端开始和数据节点交互，通过FSDataInputStream的对象调用read（）方法，选择离客户端的远近对数据节点读数据。

4、数据块读取完毕，通过FSDataInputStream的对象调用close（）方法，关闭与该数据节点的连接。

5、重复第2步操作，查找下一个数据块的地址。

6、重复第3步操作，读取数据。

7、当文件中所有数据块读取完毕，调用文件输入流DFSInputStream的close（），关闭文件输入流。

（PPT4）

客户端如果需要向HDFS写一个文件，比如一个300M的文件需要写入HDFS，client是不知道要怎么拆分，存到哪些DataNode上的，因此需要求助于NameNode，NameNode会根据各个数据节点上存储的情况，以及当前文件的大小，计算出一份合理的存储方案，告诉client应该拆分为几个block，分别存在哪几个DataNode。然后client会首先找到一个最近的DataNode，写入一个block，然后这个block会平移复制到其他分配的DataNode，完成一个block块的写入，剩余的block也是进行同样的操作。具体过程如下：

1、客户端发起写文件请求FileSystem.create（），创建文件输出流DFSOutputStream；

2、DFSOutputStream通过ClientProtocol协议远程调用名称节点，名称节点首先要进行一些检查，例如文件是否存在存在、客户端是否有权限创建文件等，当检查通过之后，名称节点才会创建文件元数据，并向客户端返回对应的所有数据块的多副本的存放数据节点列表。

3、当客户端获取了名称节点返回的所有数据块的多副本的存放数据节点列表后，开始向这些数据节点写数据块，因此要创建数据块输出流FSDataOutputStream，调用write（）方法向数据节点写入数据块。

（PPT5）

4、在确定了要写入的数据块block和数据节点DataNode位置后就可以开始写数据了。

首先客户端（Client）数据块分成4byte的一个个包（packet）放入文件输出流DFSOutputStream队列中dataQueue，然后数据输出流FSDataOutputStream会像名称节点申请保存数据块副本的数据节点地址，形成一个数据流管道pipeline，再将dataQueue的packet发往pipeline中DataNode01，DataNode01将packet发往DataNode02，DataNode02将packet发往DataNode03。

5、为了保证发送到数据节点的packet是正确的，接受packet的数据节点要发送者逆向返回ack确认包，如果返回SUCESS，则ackqueue中的镜像packate就会删除，否则会从ackqueue取出对应packet到dataqueue尝试重新发送。当block中所有的数据按照上面流程写完后，会发送一个空的packate代表写完了，关闭当前block的pipeline，其他的block写入流程类似，重复3-5步即可。

6、客户端调用DFSOutputStream的close方法关闭文件输出流，同时客户端使用ClientProtocol协议接口的complete（）方法RPC通知NameNode的名称节点关闭文件，完成本次写操作。