

`hdfs.batchSize` 控制了每个事务写出事务的最大数量（提供在 Channel 中有足够的事件可用——如果 Channel 中有更少的事件，会认为批处理立即完成）。每次批处理利用 Channel 提交一个事务。

让我们看一些 HDFS 配置的真实例子。下面的配置将事件写到 10 分钟 bucket，使用基于时间的分桶和向下取整，Snappy 格式。如果没有配置文件后缀，文件将会自动添加后缀 *.snappy*。每 2 分钟或者当 100000 个事件写到一个文件，无论哪个条件先达到，Sink 会关闭文件，如果文件已经打开 30 秒没有写入，且使用闲置超时就关闭文件。

Sink 使用内置的 TEXT 序列化器以纯文本的格式写事件。在任何时候当 100 个文件打开，Sink 也会关闭最早写入的文件。它使用 Flume 主体来做 Kerberos 认证信息登录，以 UsingFlume 用户写数据。Flume 用户必须在 HDFS 配置中被授权可以模拟 UsingFlume 用户。当写入到该 Sink 的文件使用“.”前缀使之成为隐藏文件，文件的后缀名是 *.temporary*。一旦文件关闭，文件会被重命名为最终的文件名。

```
agent.sinks = hdfsSink
agent.channels = memoryChannel

agent.channels.memoryChannel.type = memory
agent.channels.memoryChannel.capacity = 10000

agent.sinks.hdfsSink.type = hdfs
agent.sinks.hdfsSink.channel = memory
agent.sinks.hdfsSink.hdfs.path = /Data/UsingFlume/${topic}/${Y}/${m}/${d}/${H}/${M}
agent.sinks.hdfsSink.hdfs.filePrefix = UsingFlumeData
agent.sinks.hdfsSink.hdfs.inUsePrefix = .
agent.sinks.hdfsSink.hdfs.inUseSuffix = .temporary
agent.sinks.hdfsSink.hdfs.fileType = CompressedStream
agent.sinks.hdfsSink.hdfs.codec = snappy
agent.sinks.hdfsSink.hdfs.rollSize = 128000000
agent.sinks.hdfsSink.hdfs.rollCount = 100000
agent.sinks.hdfsSink.hdfs.rollInterval = 120
agent.sinks.hdfsSink.hdfs.idleTimeout = 30
agent.sinks.hdfsSink.hdfs.maxOpenFiles = 100
agent.sinks.hdfsSink.hdfs.round = true
agent.sinks.hdfsSink.hdfs.roundUnit = minute
agent.sinks.hdfsSink.hdfs.roundValue = 10
agent.sinks.hdfsSink.hdfs.kerberosPrincipal = flume/_HOST@OREILLY.COM
agent.sinks.hdfsSink.hdfs.kerberosKeytab = /etc/flume/conf/UsingFlume.keytab
agent.sinks.hdfsSink.hdfs.proxyUser = UsingFlume
```