sparkstreaming集成kafka时的maven的pom依赖：

<dependency>

<groupId>org.apache.spark</groupId>

<artifactId>spark-streaming-kafka-0-8\_2.11</artifactId>

<version>0.8.2.1/version>

</dependency>

使用SparkStreaming集成kafka时有几个比较重要的参数：

（1）spark.streaming.stopGracefullyOnShutdown （true / false）默认fasle

确保在kill任务时，能够处理完最后一批数据，再关闭程序，不会发生强制kill导致数据处理中断，没处理完的数据丢失

（2）spark.streaming.backpressure.enabled （true / false） 默认false

开启后spark自动根据系统负载选择最优消费速率

（3）spark.streaming.backpressure.initialRate （整数） 默认直接读取所有

在（2）开启的情况下，限制第一次批处理应该消费的数据，因为程序冷启动 队列里面有大量积压，防止第一次全部读取，造成系统阻塞

（4）spark.streaming.kafka.maxRatePerPartition （整数） 默认直接读取所有

限制每秒每个消费线程读取每个kafka分区最大的数据量

注意：

只有（4）激活的时候，每次消费的最大数据量，就是设置的数据量，如果不足这个数，就有多少读多少，如果超过这个数字，就读取这个数字的设置的值

只有（2）+（4）激活的时候，每次消费读取的数量最大会等于（4）设置的值，最小是spark根据系统负载自动推断的值，消费的数据量会在这两个范围之内变化根据系统情况，但第一次启动会有多少读多少数据。此后按（2）+（4）设置规则运行

（2）+（3）+（4）同时激活的时候，跟上一个消费情况基本一样，但第一次消费会得到限制，因为我们设置第一次消费的频率了。

除此之外，还应该考虑程序容错性，这个跟checkpoint有关系散仙在前面的文章已经描述过具体请参