Kaijun He
Part 2 for homework 4

1. Extensibility: Explain how your solution would support the following extensibility goal: The customer wishes to redesign the system to handle text files written in languages other than English (each file is one language). The design time modification must take less than one day. The ultimate solution must be configurable automatically at runtime.
   a. Answers: since I'm using the snowball lucene jar file and import into my own eclipse, they already support lots of different language, only different is that using different stemmer, like SpanishStemmer class for different language file, by now, it's difficult to stemmer all different languages in one file without known that which language is it for particular portion.
2. Response Time: Using the "King James version" text file, instrument your code and report on the response time of your application under conditions of no contention. Do not include infrastructure setup time (e.g. loading stopwords), but do include everything related to loading, processing, and saving results. Also evaluate the individual response times of each filter. Note that this is going to be a little tricky, because the runtime of each filter is impacted by the performance of its incoming buffer. You will have to figure out how to get around this problem. Some possible solutions include: (a) Determining the time needed to process a single element (i.e. read from incoming pipe, process, and output to outgoing pipe), or (b) Set up a driver component. Discuss your results in at least one paragraph. Identify bottlenecks (pipes, filters, etc.)
   a. Answers: there are lots of ways to deal with problem in eclipse, there is an API called System.nanoTime(), you can use this function by given starttime call System.nanoTIme() and endTime System.nanoTime(), and the difference for starttime and endTime is the response time of that part of program performance. For example, if we want to find out the time to process a single element, simply we could just get two system time when read the element from pipe one and the time when read the element from pipe two. There is another way to do the same things, we can use Stopwatch to deal with this problem as the System,nanoTime().

Kaijun He
Part 2 for homework 4

3. Based on your answer from #2 “Response Time” redesign your solution to improve response time while balancing the need for maintainability and reuse. Implement your solution and deliver a comparative graph to show how it improves performance over the original solution.
    a. Answers: by knowing the response time, if we want to improve the response time, we should let the whole pipe and filters communication as fast as we can, for example, in this homework, it has 4 filters, since the speed of 4 filters are not balanced, so some filter will be faster than other filters, to make sure we can receive correct data, some threads have to sleep to make sure when the thread is running, the next pipe already received all data needed. So if we know response time for each filter, we could simply set up a sleep time close to the time difference between the data in from pipe and out for another pipe. This will save some times, and we could also change the each filter’s algorithm to reduce performance time.