Log File Processing System

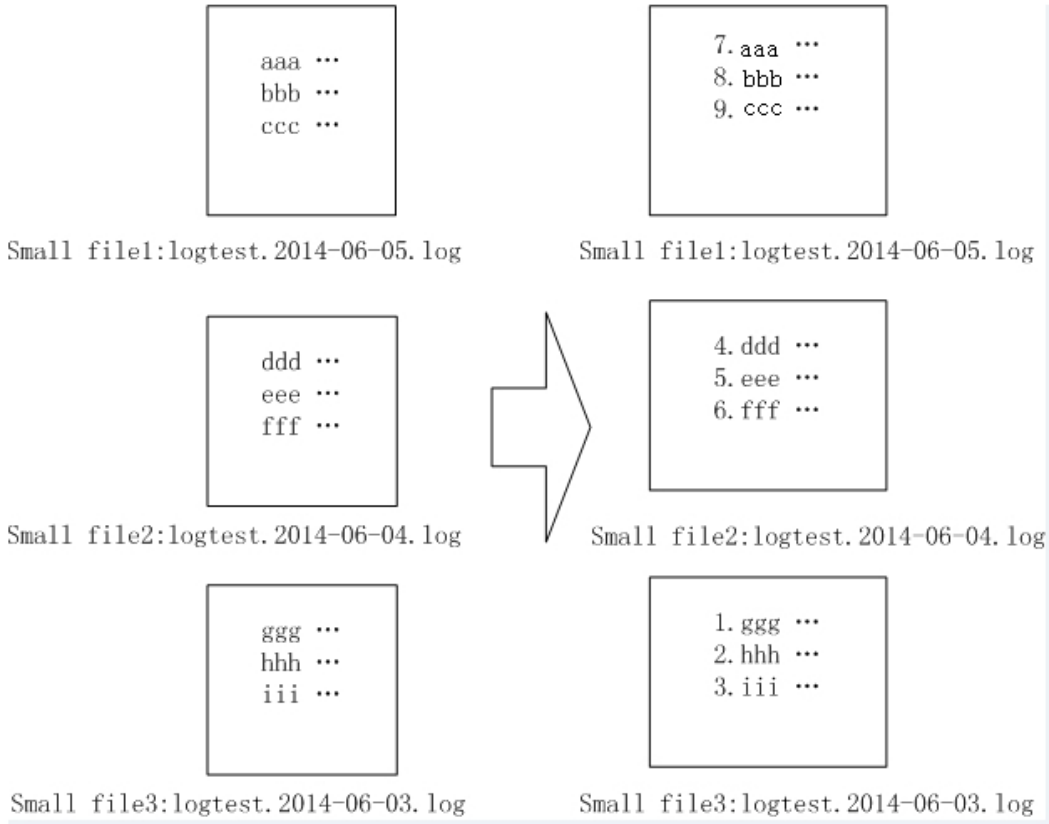
Created by Paul Li, last modified by Xiaobing Li just a moment ago

Subject

1. Please implement a program to solve the following problem.
Input: Many small log files (about 1000000 small files and assumes the total size is several TBs) and no line number for each line of the small log file.
Output: The same number of small log files and add a line number to each line of the log files according to the timestamp.
Constraints and Tips:

- The small files are named by timestamp, just like 'logtest.2014-07-11.log' and the file names are all unique.
- Assumes the storage sub-system is SSD and can handle multiple IO requests simultaneously.
- Assumes the CPU is powerful enough.
- Assumes the total memory you can use is 100MB.
- A simple solution is to create a single thread program that reads a block, processes it, reads next block, processes it, and so on. This design has significant disadvantages in performance because the processing is much faster than the disk read, so the CPU wastes lots of time to wait for the data being loaded from disk.
- You can test your program with just 20 small files, but your program can deal with 1000000 small files.
- User can configure the thread number when to launch the program.

Example:



2. Please apply the best practice of continuous integration.
Constraints and Tips:

- Use tool to control the source code version.
- Write at least one UT case for the program.
- UT tests are run automatically during building. And test results can be displayed or sent out through email after building.
- The continuous integration can work smoothly, including compiling, packaging, testing and deployment, and so on.

How do we evaluate your performance?

Can you finish the project in timely manner.
Can you clearly describe your design and implementation (you can create word documents, slides, or whatever format you like).
The document should clearly and accurately describe the system you implemented. Some applicants describe a very advanced system on paper. But their implementation is very rudimentary. Please don't do that. Just describe what you implemented.