

Worker

The worker is lowest component of all the system. The task of a worker is to read a portion of the file very quickly. The directive are read from standard input:

- a string that indicates the path of the file
- a number that indicates the char where the worker must start his work
- a number that indicates the char where the worker must end his work

Start and end points are included in the read operation. The worker after the reading operation from the standard input casts the two numbers and returns an error in case one or more fail.

Before start working the worker checks if there is enough memory for allocation. If so he allocates all work amount. If the available memory is smaller than the work amount the worker tries to allocate the 50% of the available memory.

When the work is finished the worker try to read another directives from the standard input.

Considerations

The worker can read huge file (we test it with a single file of 15 Gb). This the main reason for the unsigned long long type. The read operation is split if the work amount assigned to the worker is greater than 1.5 Gb. We made this choice because the kernel doesn't allow big read operation in one time.

The atomicity of the write operation is around 4000 Bytes but in this case is not a problem because other components wait for it.

The worker performance are very impressive. The 15 Gb was read in about 30 seconds (intel i7 7th gen, 40 Gb ram and ssd), the main problem is the pipe for the communication. They have the limit size of about 65000 bytes.

Schema

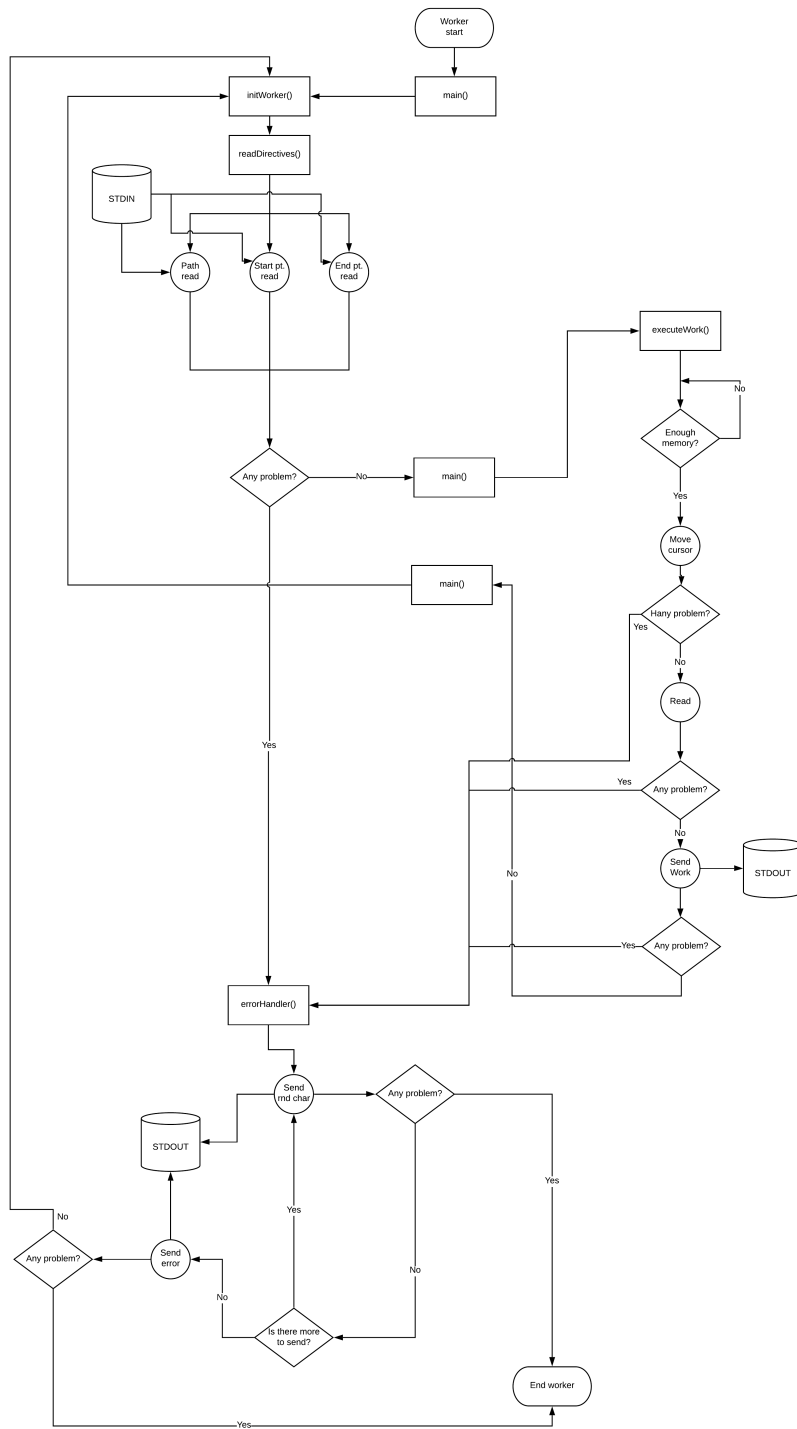


Figure 1: Worker schema