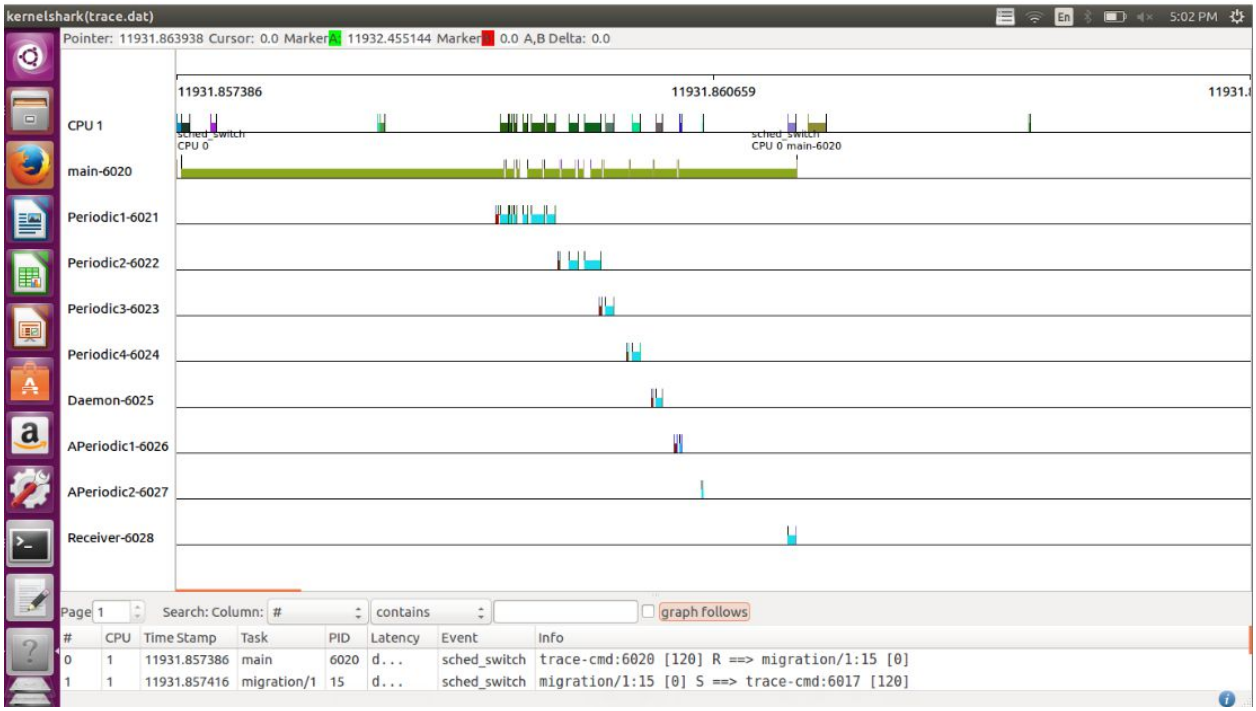
## To implement shared message queues in User Space

## Analysis of kernelshark trace for thread execution

### 1. FIFO Execution of threads

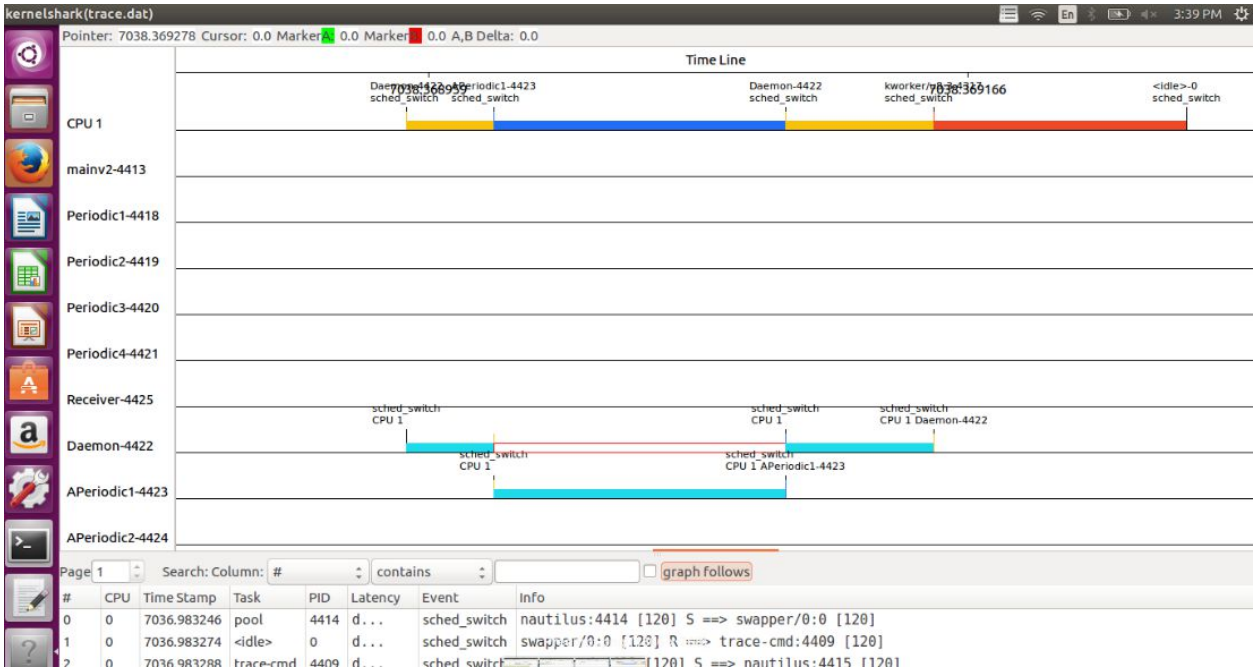



We can see from screenshot above, First-In First-Out execution of the Periodic and receiver threads.

Priority(Aperiodic threads) > Priority(Periodic threads) > Priority(Receiver thread) and hence the execution sequence is

Periodic1 -> Periodic2 -> Periodic3 -> Periodic4 ->Receiver

## 2. Pre-emption by High Priority thread




Above image explains the concept of Pre-emption.
Priority(Aperiodic thread) > Priority(Daemon thread)
When the Daemon thread posts the semaphore for click event, the Aperiodic thread gets ‘unblocked’ and pre-empts the Daemon thread to start ‘running’.
This puts Daemon thread on ‘block’ state until the Aperiodic thread finishes its execution and only then the Daemon thread continues with its execution.