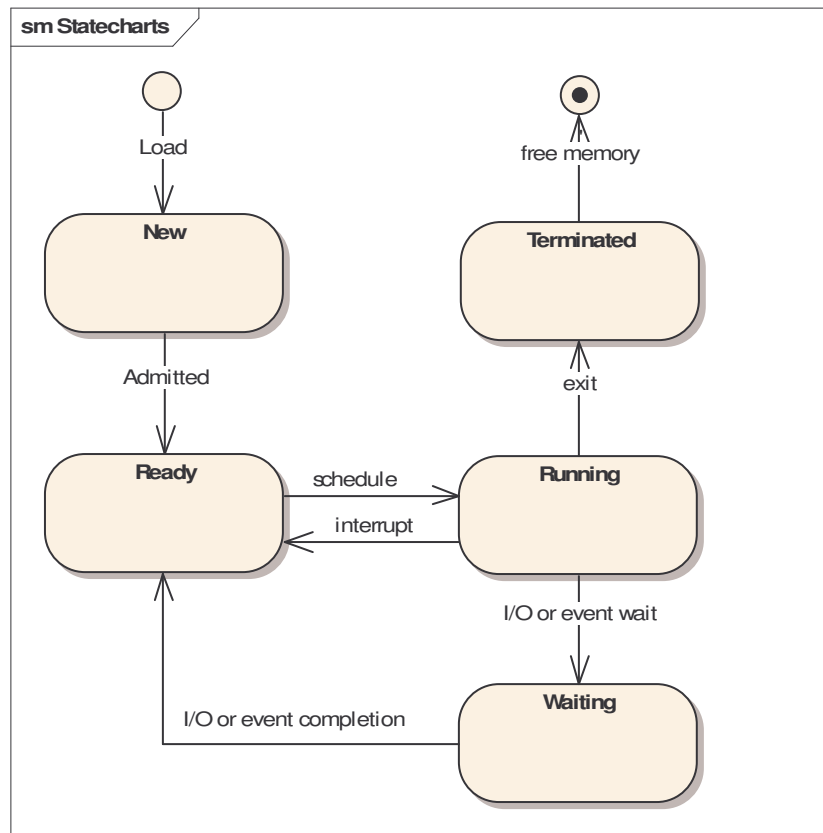


When a process is running, it “has” the CPU.

Processes in the waiting state have been context switched off the CPU and are waiting for their event to complete.



Most operating systems add a sixth state – “sleeping.”

The OS keeps track of processes using a **process table** or **process list**.

These abstract data structures are kept in OS memory space

Each process table entry is called a **process control block** (PCB) or task control block.

Process Control Block
Procnum
Procstate
Program counter
CPU registers
Memory info
Accounting info
I/O info

Program counter – what should be the next instruction to execute when the process is context switched back into memory.

Accounting info

How much CPU time has this process had so far? How many quanta has it had?

Real life time - Clock time at construction

I/O Info includes devices and files that are currently open

OS Scheduler

A kernel component or a separate process spawned by the OS

Selects the next process from the ready list / queue.

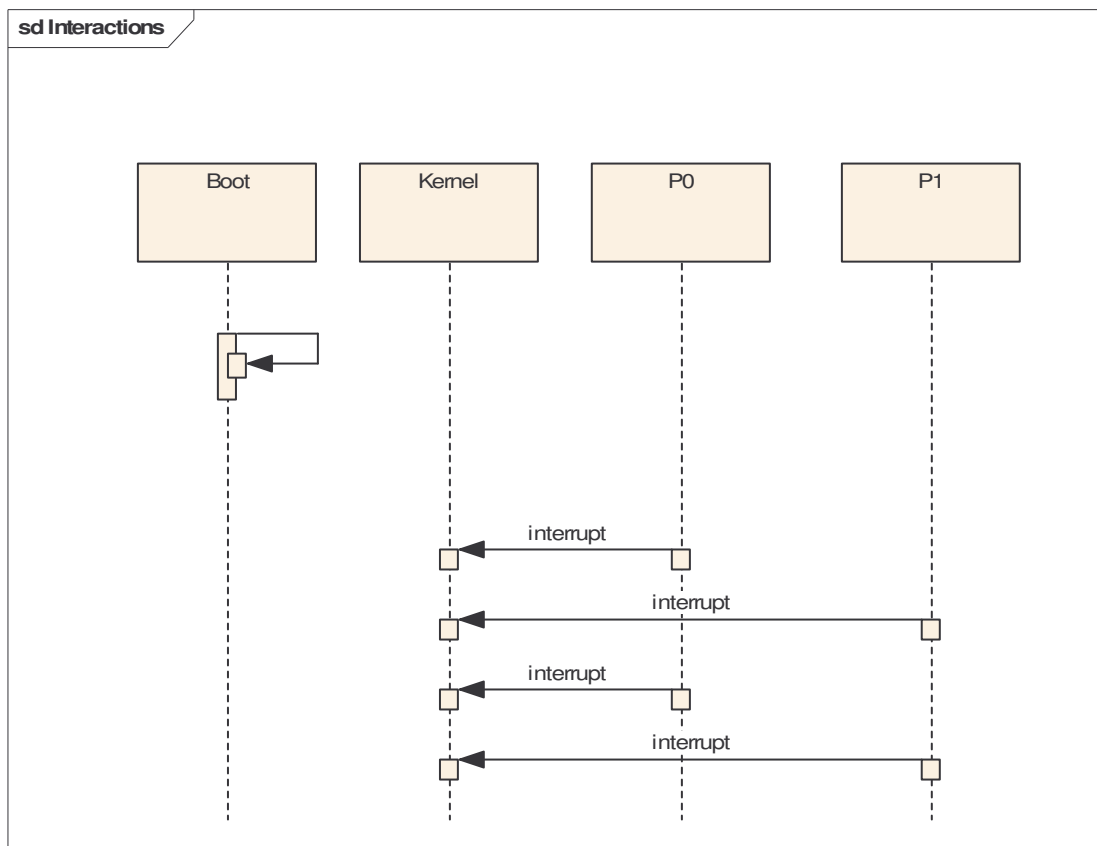
Choose the next process off the list, reorganize the queue and start the process going.

Different schedule algorithms exist.

Round Robin Scheduling

The simplest form of scheduling, RRS schedules processes in increasing (or decreasing) procnum.

$$P_0 \rightarrow P_1 \rightarrow P_2 \rightarrow \dots \rightarrow P_N \rightarrow P_0 \rightarrow P_1 \dots$$

$$P_N \rightarrow P_{N-1} \rightarrow P_{n-2} \rightarrow \dots \rightarrow P_2 \rightarrow P_1 \rightarrow P_0 \rightarrow P_N \rightarrow P_{N-1} \dots$$


System headers (on Unix / Linux) are generally stored in /usr/include/sys