COMP 3430

Operating Systems

May 15th, 2019

Goals

By the end of today's lecture, you should be able to:

- List the states that a process can have.
- Describe the lifecycle of a process.
- Describe context and mode switching.

Pixabay License

States

Let's take a look at states in `struct task_struct`.

- We're going to draw a **state transition graph** (COMP 2080, COMP 3030)
- The **start state** is the *default* state – what's the default on Linux?
- Now identify what the *next* state is, until we find an **end state**.
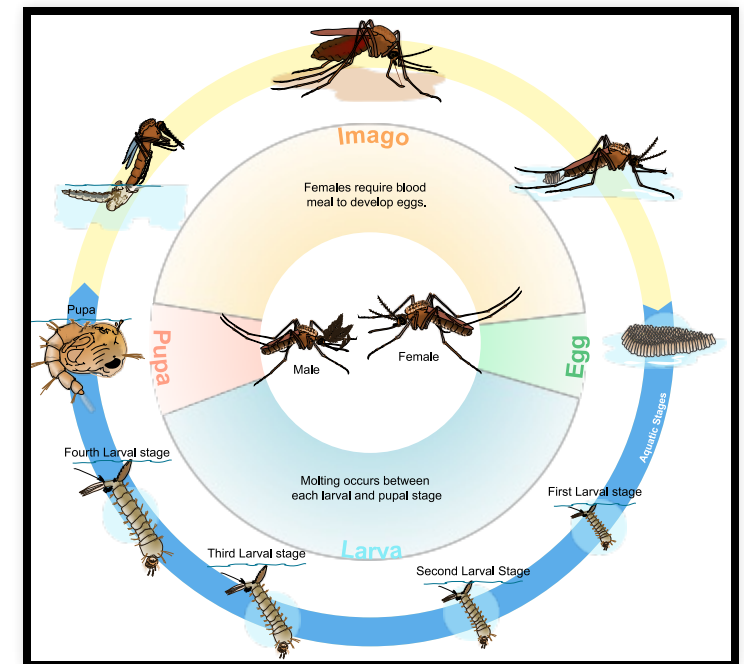
Public Domain

Process states

- This is **one** piece of metadata an OS keeps about a process.
- Helps inform:
  - Scheduling activities.
  - Maintenance activities.
- We can represent these states as a graph $G = (V, E)$ where:
  - V are the states
  - E are possible transitions *between* states

Lifecycle of a process

- We're going to draw ✏ a *timeline* of the life of a process. Draw along!
- Process **state** and **lifecycle** are *deeply interrelated*
    - State is a *description* of process lifecycle.
- The process lifecycle starts *before* the process exists.

Public Domain

Trap tables

- Remember: The OS is *tightly coupled* with the hardware.
  - The OS *cooperates* with the hardware to accomplish its job.
- Let's peruse the system calls for `x86_64` on Linux
- How does the kernel register system calls with hardware???
  - A *bit* outside the scope of this course.
  - …you can read about it on LWN (1 and 2).

© English Lock (CC BY-SA 3.0)

A process starts

We've typed in

`./myprocess`

and hit Enter on our keyboard.

Let's draw ✎

Pixabay License

Process ends

- The process calls `exit()` (… or crashes) (… `Segmentation Fault`)
- What responsibilities might the OS have here?
    - What kinds of things does it need to **clean up**?

Pixabay License

Context switching

Main question: Is context switching fundamentally different from mode switching?

How to tell a centipede that he's been a bad centipede

Process Lifecycle

- (… again) Processes have a state.
    - State is a representation of stage in lifecycle.
- The OS *cannot* do everything by itself, it needs help from hardware.
    - OS interacts with hardware by registering system calls.
    - Hardware enables *mode switching*.
- Context switching and mode switching are related to one another.

Next week

Let's take a look at the schedule.
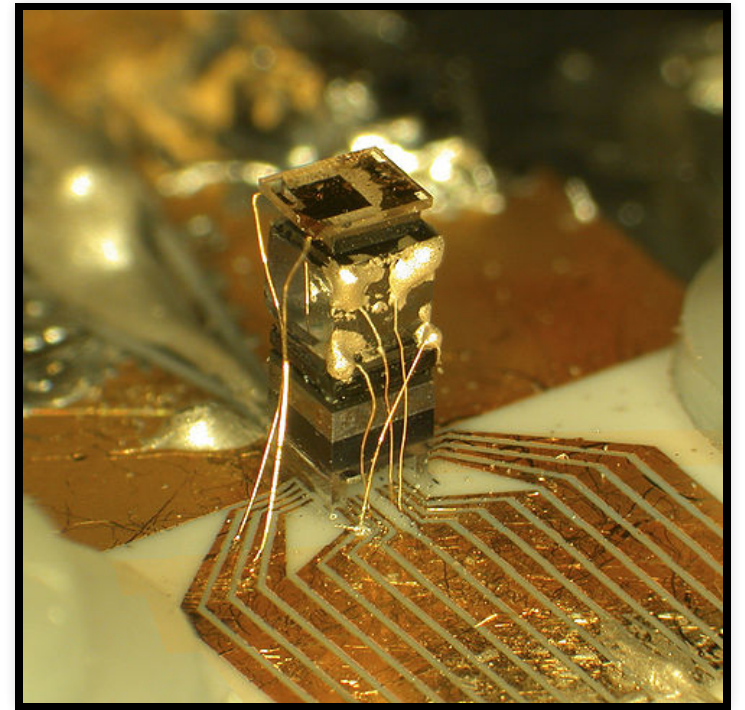
Chip-scale atomic clock (Public Domain)