Duke Systems

CPS 310 / ECE 353

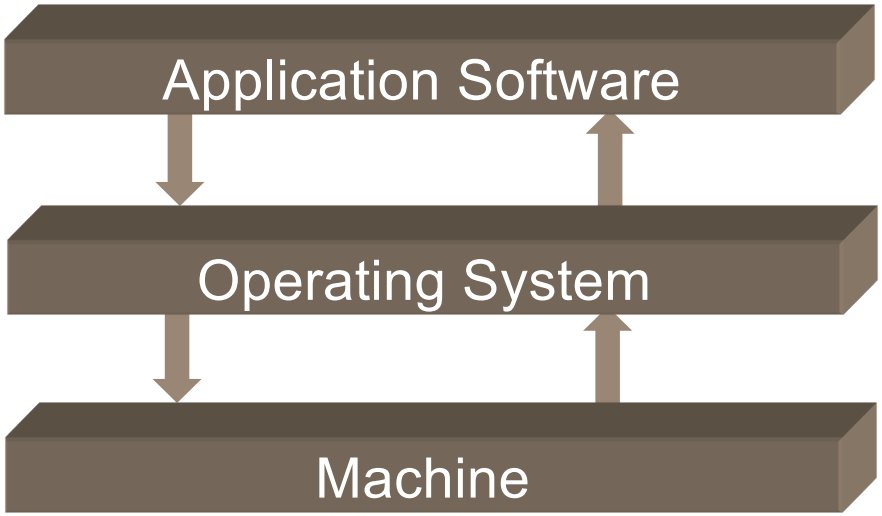
Introduction to Operating Systems

Jeff Chase

Duke University

What is this course about?

- Programs
- Platforms
- Sharing
- Concurrency
- Storage
- Protection and trust
- Resource management
- Virtualization
- Scale and performance
- Abstractions







A broader view of OS

- OS are **platforms** for running programs on machines.
  - **Program**: software packaged for the platform.
  - **Platform**: defines an environment for software and its data and interactions.
  - **Machine**: it runs software—may be virtual.
- Examples
  - Mobile device
  - Embedded/IoT
  - Supercomputer
  - Cloud services
  - Web browser

Platform abstractions

- Platforms provide “building blocks”…
- …and APIs to use them to construct software.
  - Instantiate/create/allocate
  - Manipulate/configure
  - Attach/detach
  - Combine in uniform ways
  - Release/destroy
- Abstractions are layered.
  - Expose the power through APIs
  - Hide the details behind APIs

The choice of abstractions reflects a philosophy of how to build and organize software systems.

Unix: A lasting achievement?

"Perhaps the most important achievement of Unix is to demonstrate that a powerful operating system for interactive use need not be expensive…it can run on hardware costing as little as $40,000."

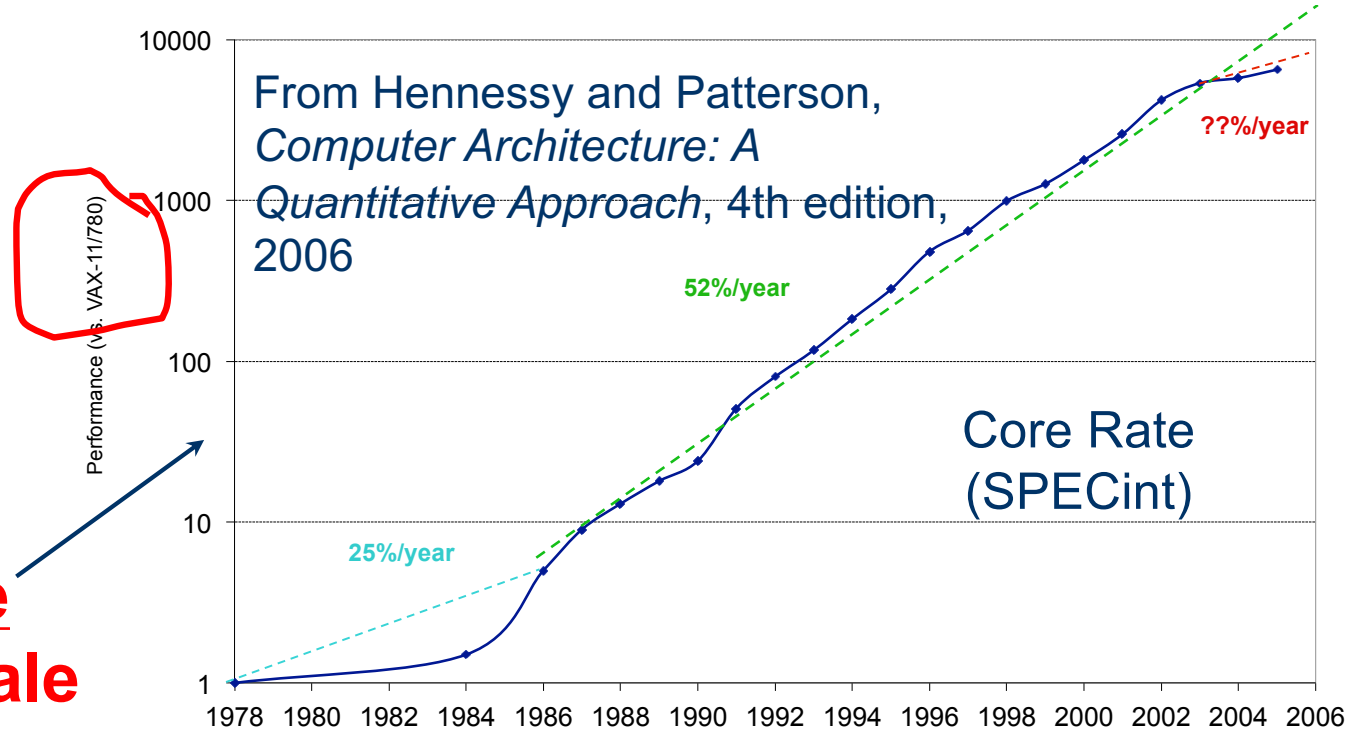
The UNIX Time-Sharing System*
D. M. Ritchie and K. Thompson
1974

DEC PDP-11/24

http://histoire.info.online.fr/pdp11.html

terminal (**tty**)
"teletypewriter"

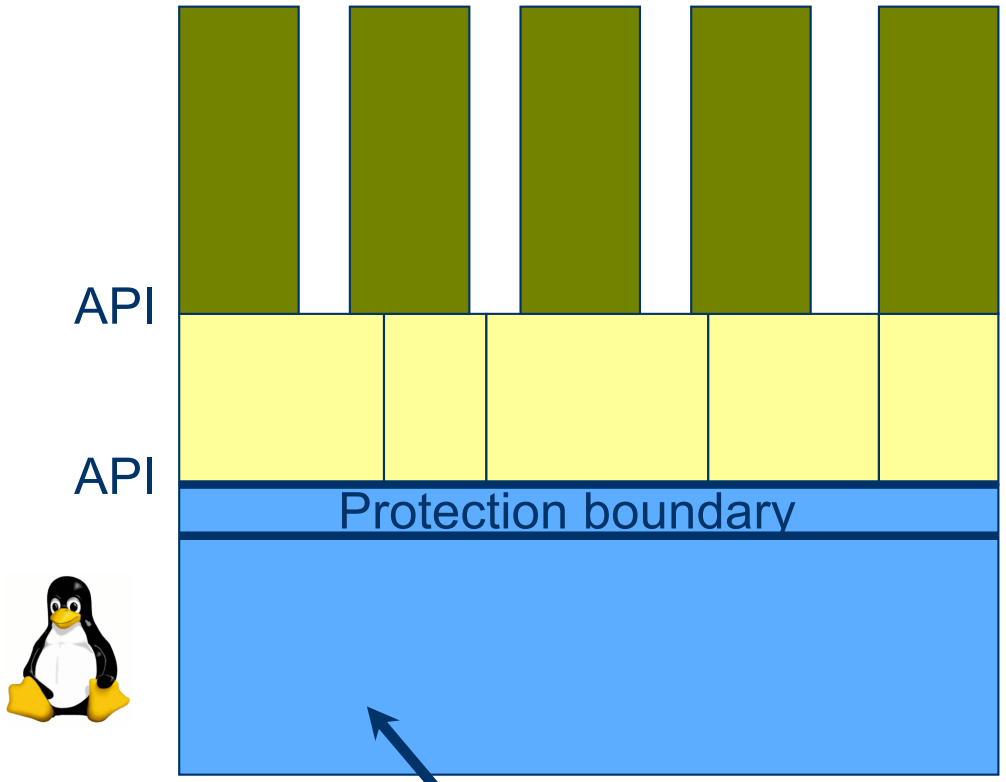
Let's pause a moment to reflect...



Note log scale

Today Unix runs embedded in devices costing < $100.

OS Platform: a model



Applications/services. May interact and serve one another.

Libraries/frameworks: packaged code used by multiple applications

OS platform: same for all applications on a system E.g., classical OS **kernel**

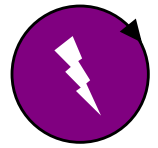
OS mediates access to shared resources.
That requires trust, protection and isolation.

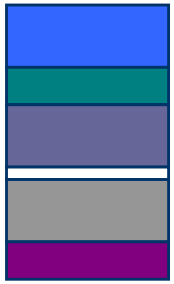
[RAD Lab]

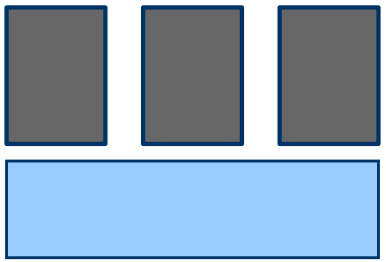
Isolated environments for programs

Processes and virtualization

- In a (classical) OS programs run as processes.
- A **process** is a running program instance.
- A process has a **thread** (at least one) to run code.
- OS multiplexes the computer among processes.
- Processes are isolated from one another.
  - A process has one **virtual address space**.
  - VAS defines its view of machine's memory.
  - It sees only what the kernel lets it see.
- The OS kernel controls **everything**.
- OS kernel creates/destroys processes.

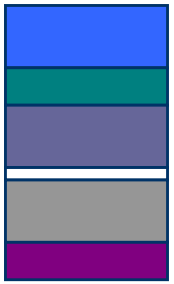






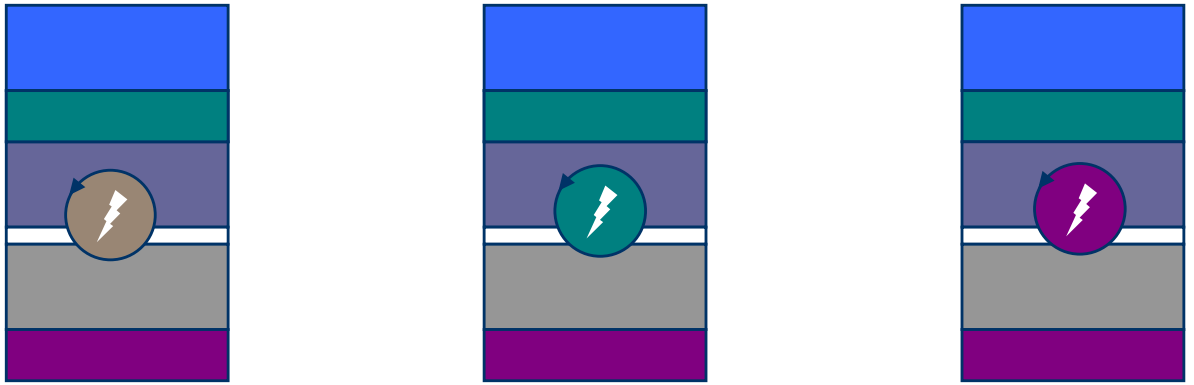
Virtual address spaces

- A virtual address space (VAS) is a window on memory.
  - Defines what portions of memory a process can access.
  - Defines what names (addresses) to use.
- The machine and kernel work together to support VAS.
- Metaphors:
  - **Sandbox**: secure to play in; can't get out.
  - **Lockbox**: nobody else can get in.



OS: classical view

Programs run as independent processes.

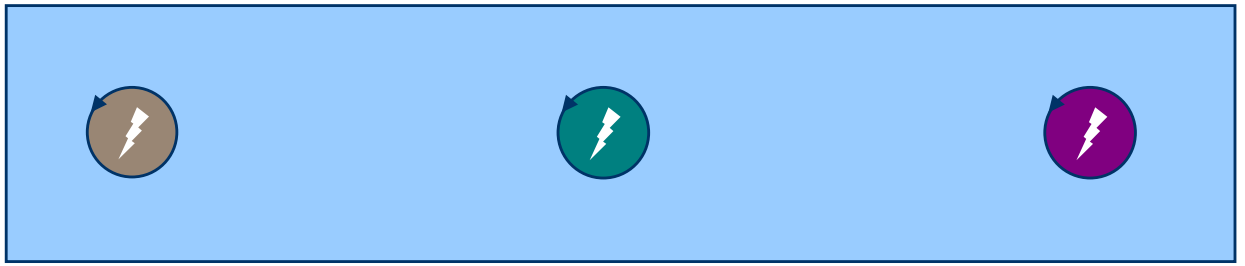


Each process has a private virtual address space and one or more threads.

Protected system calls

...and upcalls (e.g., signals)

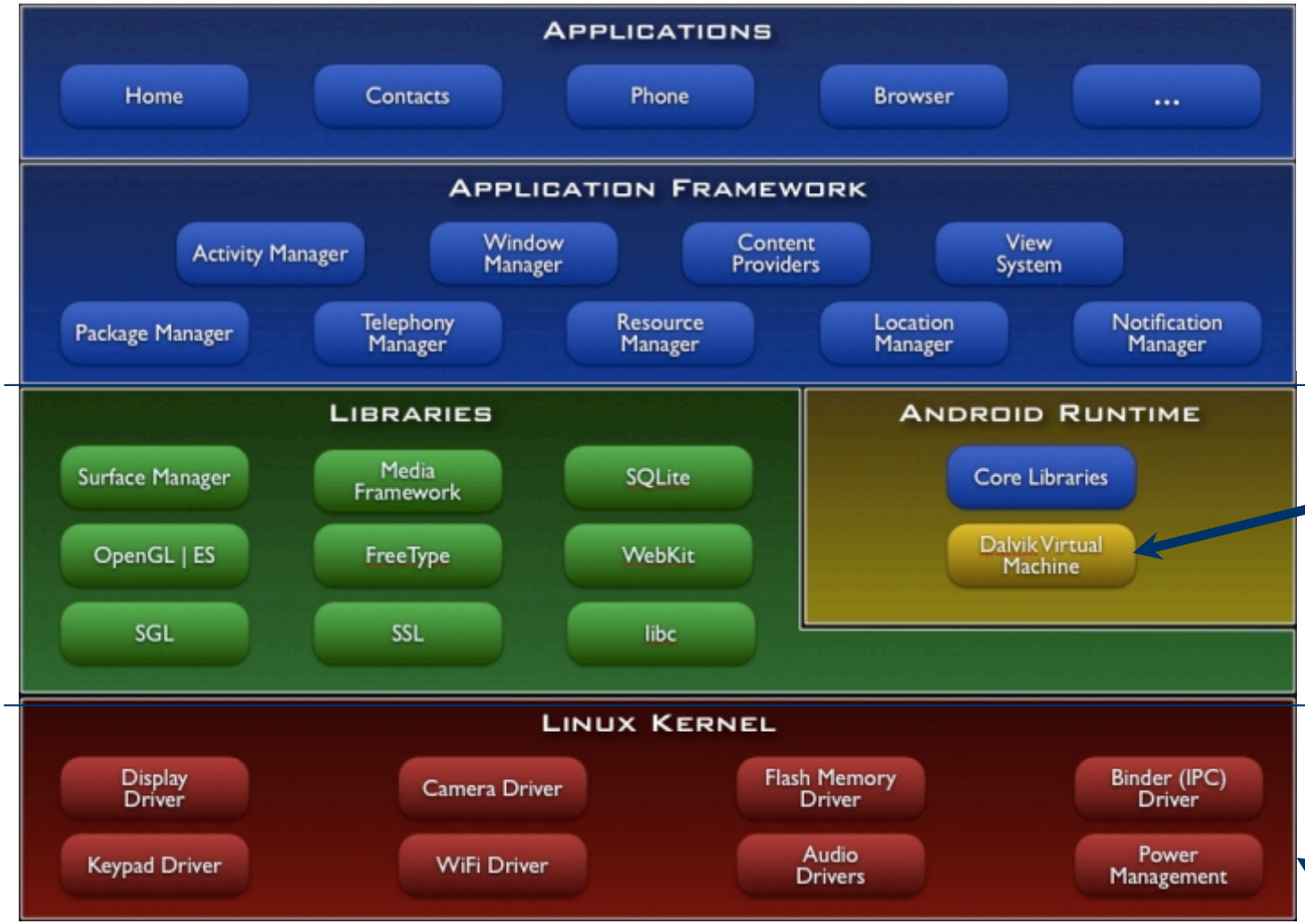
Protected OS kernel mediates access to shared resources.



Threads enter the kernel for OS services.

The kernel code and data are protected from untrusted processes.

Platforms are layered/nested



Virtual Machine (JVM)

System libraries (native C/C++)

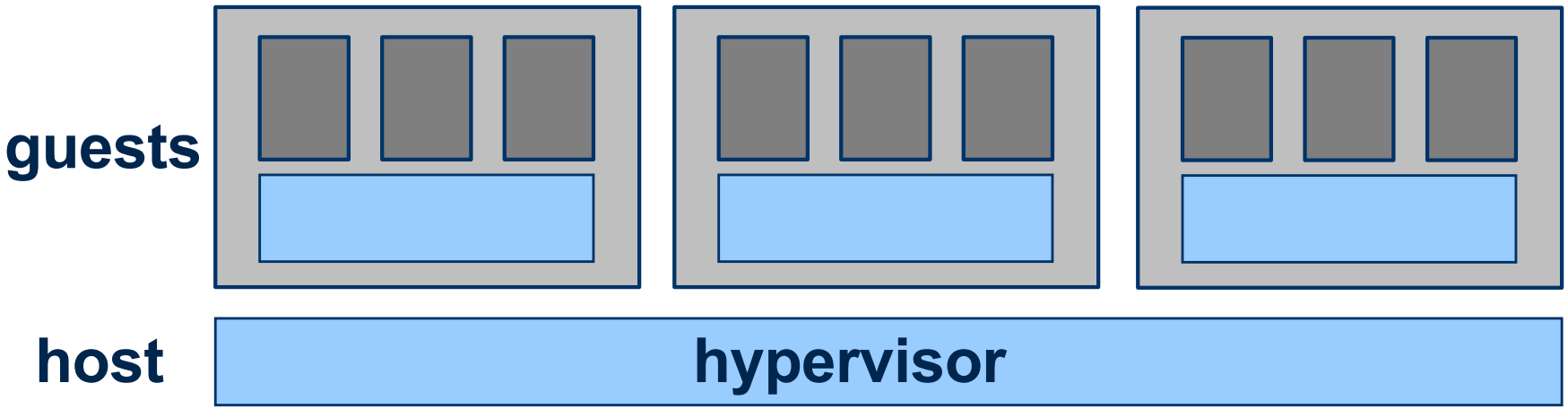
"Classical OS kernel etc.

Reloaded with a few new kernel extensions (drivers).

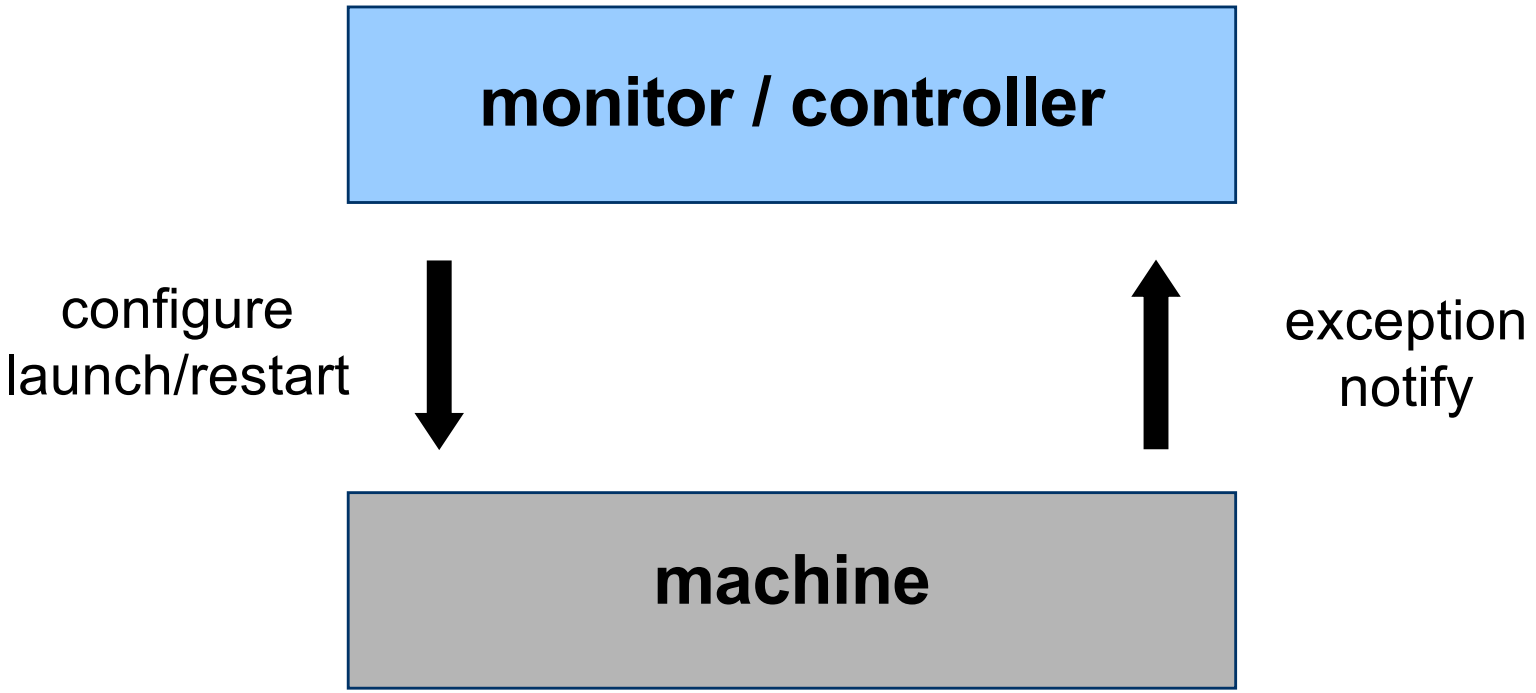
[http://www.android.com]

Virtual machines (VMs)

- Run a **hypervisor** on the “bare metal” physical machine.
  - New trusted layer!
- Kernel and processes run in a **virtual machine (VM)**.
  - Hypervisor manages multiple “instances” of the machine.
  - The VM “looks the same” to the OS as a physical machine.
- Can run multiple OS on a shared computer.



The p-p-paradigm

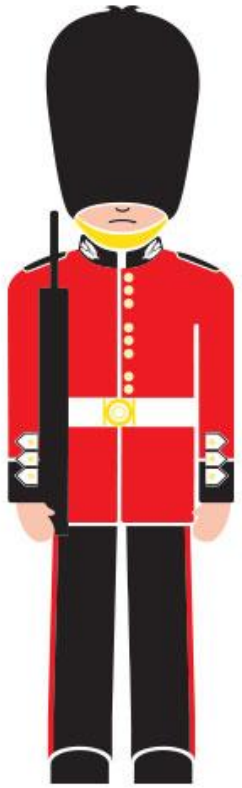


Machine runs according to its configuration. If it encounters a condition that requires controller to intervene, it suspends processing and generates an exception for the controller.

Summary: faces of your OS

Serving your requests

Directing traffic



Guarding your private property