

OS Structures

Microkernels and VMMs

OS Organization

- Monolithic
 - Layered
 - Modular
 - Microkernels
-
- Pros and Cons: consider size, maintainability and correctness, extensibility and flexibility, performance, portability, ability to support existing applications...

Microkernels

- Liedtke: “... to minimize the kernel and to implement whatever possible outside of the kernel...”.
- What is a minimal kernel?
 - Provide minimal set of services, & support a minimal set of abstractions
 - Everything else implemented as a service outside of the ukernel

e.g.,

Abstraction	Services
Address space	Map, grant, flush pages
Execution context, i.e, thread	Context switch
Communication channel between address spaces	IPC/RPC

- and unique identifiers
- many other examples exist...
- is good performance feasible?

Virtualization

- Goldberg: “... software which transforms the single machine interface into the illusion of many. Each of these interfaces (VMs) is an efficient replica of the original computer system, complete with all of the processor instructions...”
- bottom layer -> Virtual Machine Monitor or hypervisor
 - Correctness and trust critical:
 - Must provide strong resource isolation
 - Must enable sharing of hardware securely
 - “Cost” associated with it:
 - Must limit performance overheads
 - OS within VM should not know it’s not executing directly on the real hardware...
 - Some conflicts & challenges here...

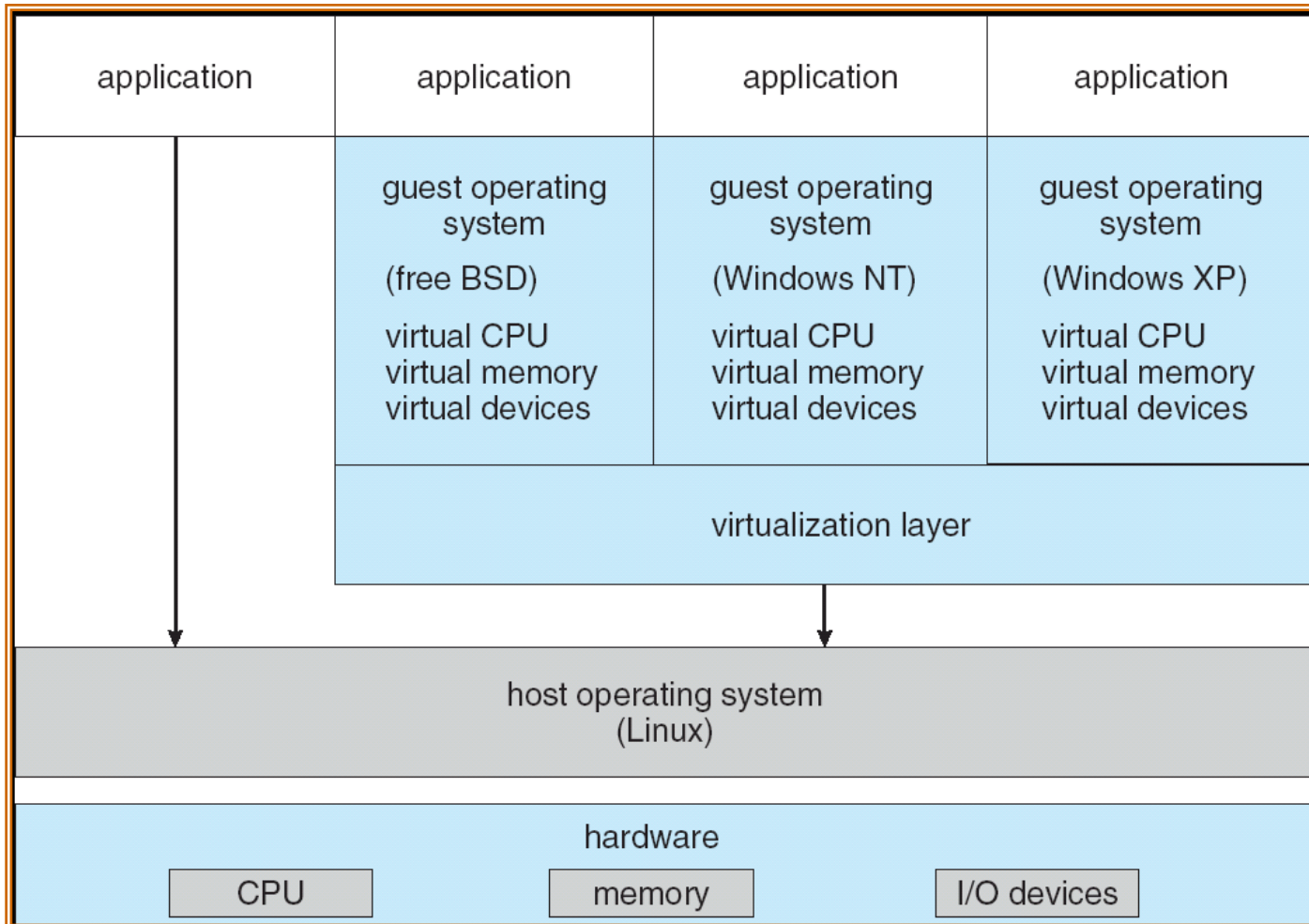
Benefits

- \$ - power, real estate, infrastructure...
- Manageability, Reliability, Availability
 - Server consolidation, staged upgrade deployment, failure confinement, ...
- Security
 - Encapsulate untrusted SW
 - Separate environment for trusted SW
- Support for legacy OSs, ISAs...
- Development

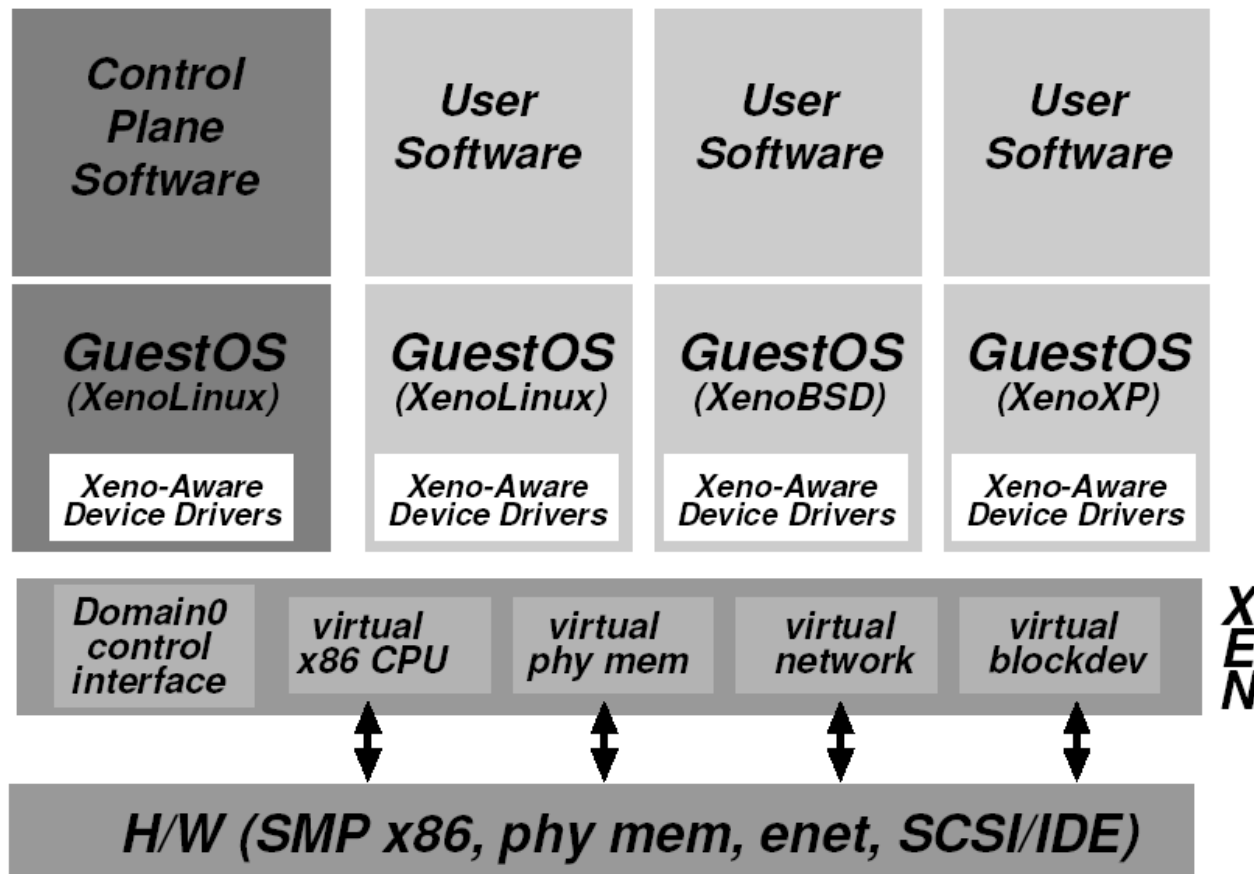
Virtualization Options

- Hosted or Hypervisor-based
 - UML – hosted
 - Xen - hypervisor
- Para- or Full-virtualization
 - modified (Xen) or unmodified (VMWare) guest OSs
 - OS needs to be modified because: certain instructions don't trap on some hardware (e.g., x86), certain instructions rely on ring in which they are executed (and in virtualized system the hypervisor or host OS is in ring 0 and guest OS is not), ...
 - Alternative to modifying OS in those cases is to do binary rewriting (e.g., original VMWare approach)
- Software or Hardware supported
 - Hardware supported – Intel VT-enabled, AMD Pacifica...

Example: Hosted, Full



Xen: hypervisor-based, paravirtualization



VM – Device Interaction

- E.g., Xen example
- Challenges:
 - Single device, shared among VMs
 - Device drivers buggy & many
 - => special Service/Control domains
- Dom0 in Xen
- Possibilities for improved efficiency...