Virtualization – Revision Notes

◇ Virtualization – Quick Revision Notes

Definition:
Virtualization is the technique of running multiple OS and applications on the same physical hardware by abstracting and sharing resources.
Focus Area:

- **Platform Virtualization** – Primary focus; runs multiple OS on a single machine.

Key Benefit:

- Simultaneously runs **multiple isolated OSes** (e.g., Linux + Windows) on the **same hardware**, unlike dual-boot which runs one at a time.

◇ Hypervisor (Virtual Machine Monitor - VMM)

Popek and Goldberg Criteria:
For virtualization to work efficiently:

- **Privileged Instructions:** Trap if run in user mode.
- **Sensitive Instructions:** Can either **change** or **observe** system state.

☑ **Efficient virtualization is only possible when**:
Sensitive ⊆ Privileged Instructions → So hypervisor can trap and emulate them.

◇ Types of Virtualization

1. Full Virtualization

- Uses **Dynamic Binary Translation**
- Entire hardware stack (CPU, memory, I/O) is emulated.
- Guest OS is **unaware** it is virtualized.

Examples: QEMU, Bochs
Pros:

- OS & VMs are isolated.
- Portability between different hardware (Dell ↔ HP).
- Can run different architectures.

Cons:

- **Performance hit** due to emulation.
- **Ring 0 conflict**: Guest OS expects Ring 0, but VMM controls it → needs trapping/emulation.

2. Para-Virtualization

- Guest OS is **modified** to communicate directly with the VMM using APIs.
- No need for instruction trapping/emulation.

Methods:

- **Kernel recompilation** (e.g., Novell Linux)
- **Para-virtualized drivers** (for partial virtualization)

Pros:

- Better performance than full virtualization.

Cons:

- Needs OS modification or specific builds.
- Not supported by all OS vendors (e.g., Microsoft).

3. Hardware-Assisted Virtualization

- Uses CPU extensions: **Intel VT**, **AMD-V**
- Guest OS runs at **Ring 0**, VMM runs at a special **Virtual Ring -1**
- No need to modify OS (supports **legacy OS**)

Pros:

- Runs unmodified OSes.
- Removes many traditional VMM complexities.

Cons:

- Unmodified OSes cannot use advanced virtualization features (solved via partial para-virtualization).

◇ Network Virtualization

Definition:
Allows multiple service providers to build, manage, and isolate **multiple virtual networks** dynamically over shared infrastructure.
Model Components:

- **Business Model**, **Architecture**, **Design Principles**, **Design Goals**

◇ Design Principles

1. **Concurrent Heterogeneous VNs**
2. **Recursion of VNs**
3. **Architectural Attribute Inheritance**
4. **Virtual Node Revisitation**

◇ Design Goals

- **Flexibility**

- **Manageability**
- **Scalability**
- **Security & Privacy**
- **Isolation**
- **Programmability**
- **Heterogeneity**
- **Experimentation**
- **Legacy Support**

◇ **Typical Network Virtualization Approach**
- Technologies: **IP**, **ATM**
- Layers: Layer-specific virtualization
- Domains: Network resource mgmt, spawning new VNs
- Virtualization Levels:
  - **Node Virtualization**
  - **Full Virtualization**

☑ **Exam MCQ Pointers Summary**

Topic	**Keyword/Definition to Remember**
Virtualization	Run multiple OS on same hardware
Hypervisor	Software managing multiple virtual machines
Popek & Goldberg Requirement	Sensitive ⊆ Privileged → efficient virtualization possible
Full Virtualization	No OS modification, total emulation (QEMU, Bochs)
Para-Virtualization	Modified OS, uses APIs, better performance
Hardware-assisted Virtual.	Intel VT / AMD-V, unmodified OS, special ring -1
Network Virtualization	Multiple VNs over shared infrastructure
Design Goals	Scalability, Security, Flexibility, etc.
Virtualization Levels	Node Virtualization, Full Virtualization