

DISTRIBUTED SYSTEMS

---

NETWORK VIRTUALIZATION

**Logical separation of the request for some service from the physical resources that actually provide that service.**

what is virtualization?

# VIRTUALIZATION

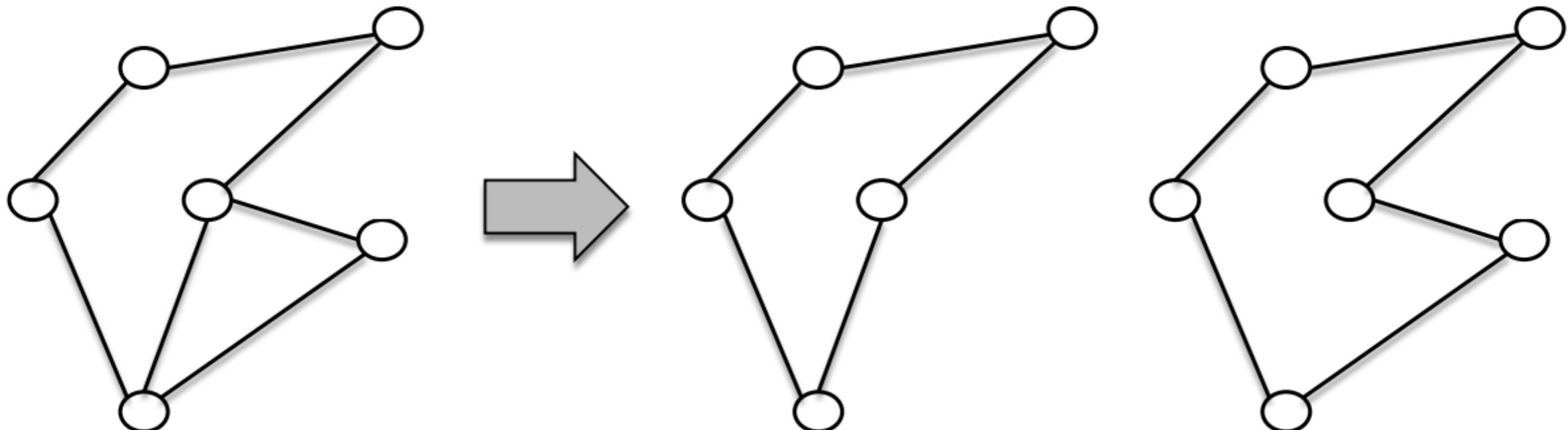
---

- ▶ Virtualization deals with “extending or replacing an existing interface so as to mimic the behavior of another system”
- ▶ Provides a level of logical abstraction that liberates applications system services, and even the OS that supports them from being tied to a specific piece of hardware
- ▶ Focusses on logical operating environments rather than physical ones
- ▶ Makes applications, services, and instances of an OS portable across different physical computer systems

# NETWORK VIRTUALIZATION FOR DUMMIES

---

- Making a physical network appear as multiple logical ones



# WHY VIRTUALIZATION?

---

## Hurdle 1 : Deployment Dilemma

- ▶ An unfortunate catch-22
  - ▶ Must deploy an idea to demonstrate feasibility
  - ▶ Can't get an undemonstrated idea deployed
- ▶ A corollary: the testbed dilemma
  - ▶ Production network: real users, but can't change
  - ▶ Research testbed: easy changes, but no users
- ▶ Bad for the research community
  - ▶ Good ideas sit on the shelf
  - ▶ Promising ideas do not grow up into good ones

# WHY VIRTUALIZATION

---

## Hurdle 2 : Too many design goals

- ▶ Many different system-engineering goals
  - ▶ Scalability, reliability, security, privacy, robustness, performance guarantees, ...
  - ▶ Perhaps we cannot satisfy all of them at once
- ▶ Applications have different priorities
  - ▶ Online banking: security
  - ▶ Web surfing: privacy, high throughput
  - ▶ Voice and gaming: low delay and loss
- ▶ Compromise solution isn't good for anyone

# WHY VIRTUALIZATION

---

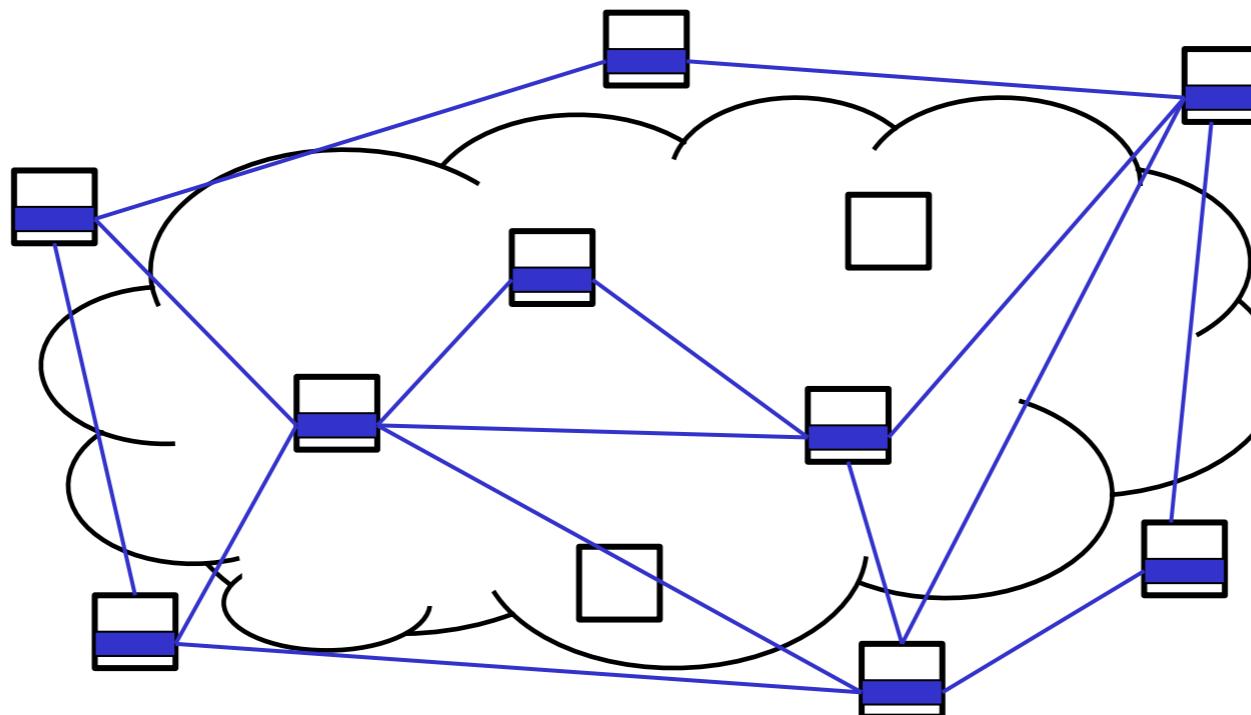
## Hurdle 3 : Coordination constraint

- ▶ Difficult to deploy end-to-end services
  - ▶ Benefits only when most networks deploy
  - ▶ No single network wants to deploy first
- ▶ Many deployment failures
  - ▶ QoS, IP multicast, secure routing, IPv6,....
  - ▶ Despite solving real, pressing problems
- ▶ Increasing commoditization of ISPs

# VIRTUALIZATION TO THE RESCUE!

---

- ▶ Multiple customized architectures in parallel
  - ▶ Multiple logical routers on a single platform
  - ▶ Isolation of resources, like CPU and bandwidth
  - ▶ Programmability for customizing each “slice”



# OVERCOMING THE HURDLES

---

- ▶ **Deployment Dilemma**

- ▶ Run multiple experimental networks in parallel
- ▶ Some are mature, offering services to users
- ▶ Isolated from others that are works in progress

- ▶ **Too Many Design Goals**

- ▶ Run multiple operational networks in parallel
- ▶ Customized to certain applications and users

- ▶ **Coordination Constraint**

- ▶ Run multiple end-to-end services in parallel
- ▶ Over equipment owned by different parties

# NETWORK VIRTUALIZATION MODEL

# NETWORK VIRTUALIZATION MODEL

---

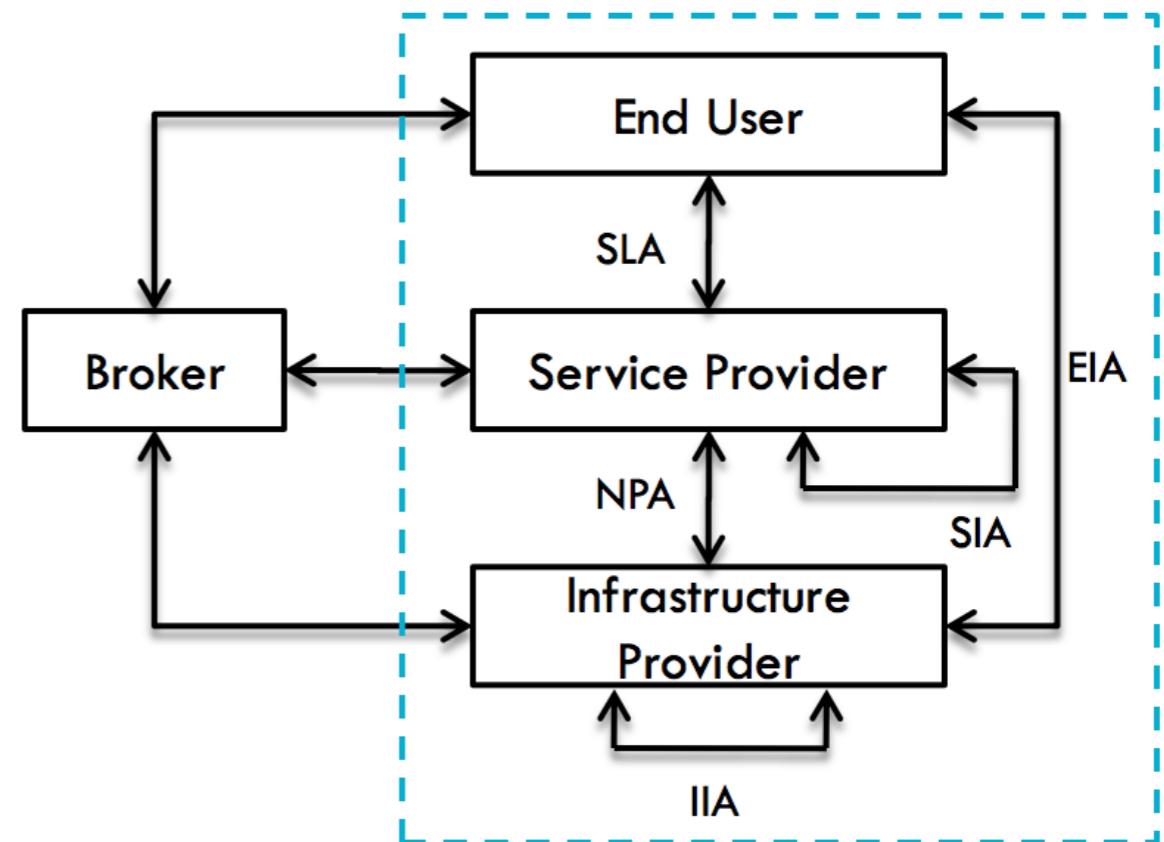
- ▶ Business Model
- ▶ Architecture
- ▶ Design Principles
- ▶ Design Goals

# BUSINESS MODEL

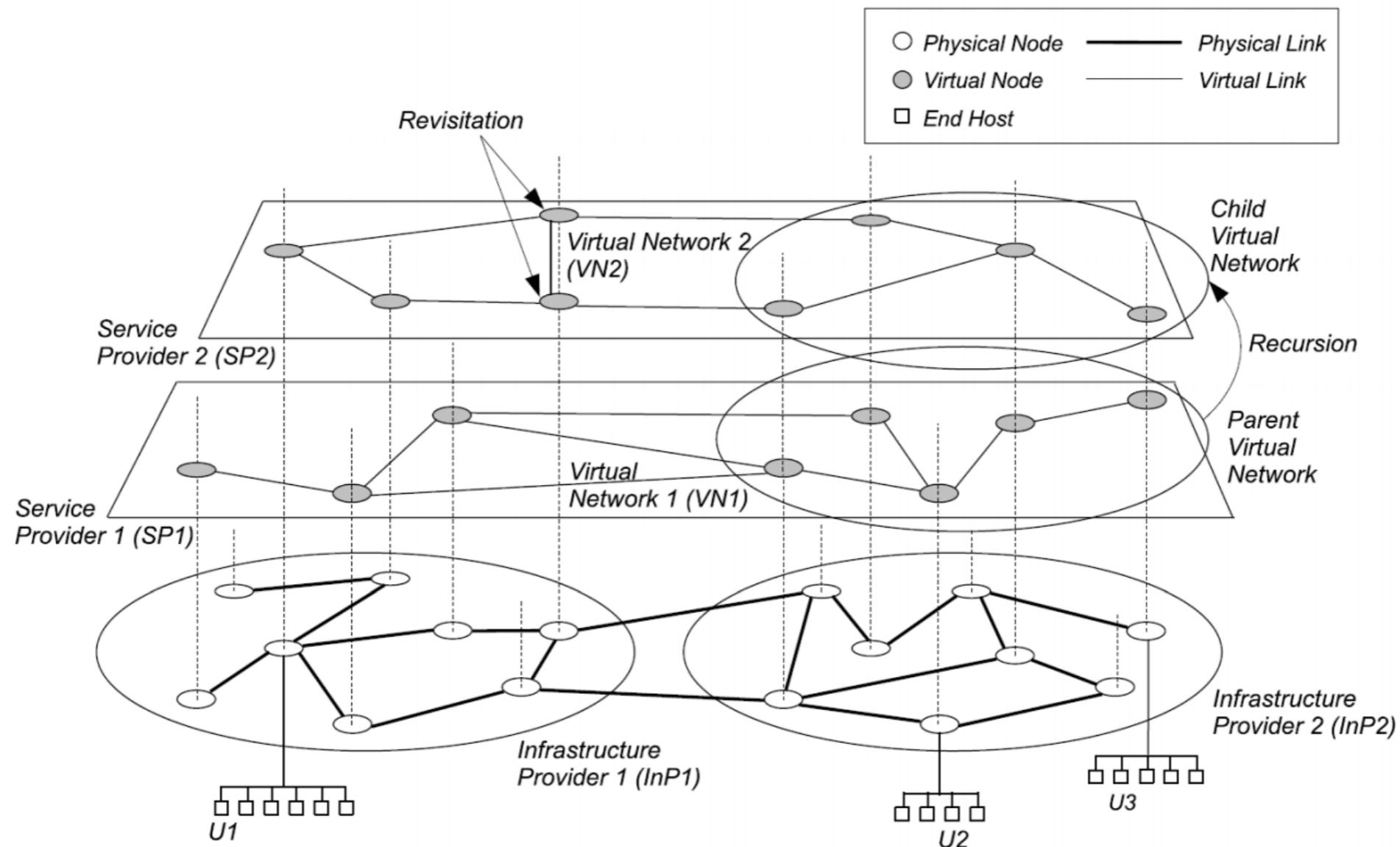
## Players

- Infrastructure Providers (*InPs*)
  - Manage underlying physical networks
- Service Providers (*SPs*)
  - Create and manage virtual networks
  - Deploy customized end-to-end services
- End Users
  - Buy and use services from different service providers
- Brokers
  - Mediators/Arbiters

## Relationships



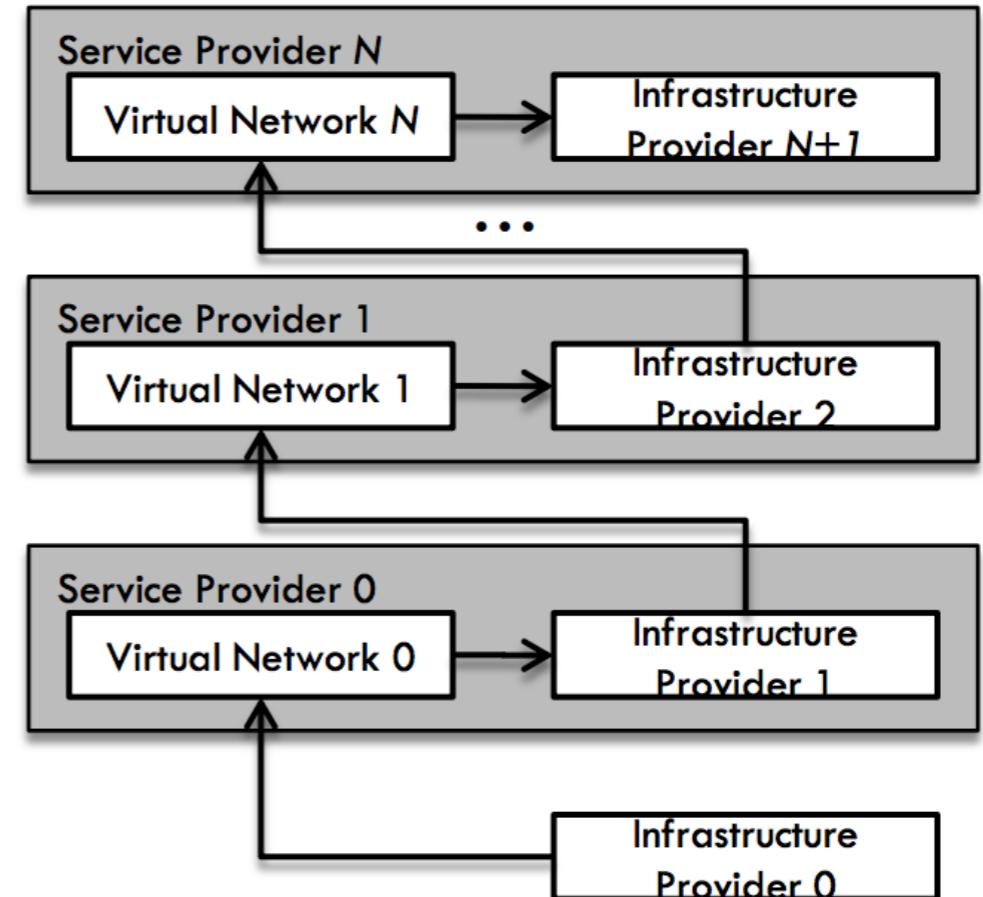
# ARCHITECTURE



# DESIGN PRINCIPLES

- Concurrency of multiple heterogeneous virtual networks
  - Introduces diversity
- Recursion of virtual networks
  - Opens the door for network virtualization economics
- Inheritance of architectural attributes
  - Promotes **value-addition**
- Revisitation of virtual nodes
  - Simplifies network operation and management

## Hierarchy of Roles



# DESIGN GOALS

---

- **Flexibility**
  - Service providers can choose
    - arbitrary network topology,
    - routing and forwarding functionalities,
    - customized control and data planes
  - No need for co-ordination with others
    - IPv6 fiasco should never happen again
- **Manageability**
  - Clear separation of policy from mechanism
  - Defined accountability of infrastructure and service providers
  - Modular management

# DESIGN GOALS

---

- **Scalability**
  - Maximize the number of co-existing virtual networks
  - Increase resource utilization and amortize CAPEX and OPEX
- **Security, Privacy, and Isolation**
  - Complete isolation between virtual networks
    - *Logical* and *resource*
  - Isolate faults, bugs, and misconfigurations
    - Secured and private

# DESIGN GOALS

---

- Programmability
  - Of network elements e.g. routers
  - Answer “*How much*” and “*how*”
  - Easy and effective without being vulnerable to threats
  
- Heterogeneity
  - Networking technologies
    - Optical, sensor, wireless etc.
  - Virtual networks

# TYPES OF VIRTUALIZATION

---

- ▶ Full virtualization
- ▶ Para-virtualization
- ▶ Hardware assisted virtualization

# ADVANTAGES

---

- ▶ Better Use of Existing Hardware
- ▶ 2. Reduction in New Hardware Costs.
- ▶ 3. Reduction in IT Infrastructure Costs.
- ▶ 4. Simplified System Administration.
- ▶ 5. Increased Uptime and Faster Failure Recovery.
- ▶ 6. Simplified Capacity Expansion.
- ▶ 7. Simpler Support for Legacy Systems and Applications.
- ▶ 8. Simplified System - Level Development.
- ▶ 9. Simplified System Installation and Deployment.
- ▶ 10. Simplified System and Application Testing.

# DISADVANTAGES

---

- ▶ It is not appropriate for all scenarios,
- ▶ It introduces real costs and concerns all its own:
  1. Single Point of Failure Problems
  2. Server Sharing and Performance Issues
  3. Per - Server Network Congestion
  4. Increase in Networking Complexity and Debugging Time
  5. Increased Administrative Complexity

# RELATED CONCEPTS

---

- Virtual Private Networks (VPN)
  - Virtual network connecting distributed sites
  - Not customizable enough
- Active and Programmable Networks
  - Customized network functionalities
  - Programmable interfaces and active codes
- Overlay Networks
  - Application layer virtual networks
  - Not flexible enough

*Fin*

