Διαχείριση δικτύων

Νάνσυ Αλωνιστιώτη

nancy@di.uoa.gr

PROJECT-BASED

- ΕΙΣΑΓΩΓΗ
- ΠΕΡΙΓΡΑΦΗ PROJECT
- SUPPORT LECTURES -ΤΡΙΤΕΣ
- PROJECT-TEAMS Q&A – ΠΑΡΑΣΚΕΥΕΣ (ΚΑΤΟΠΙΝ ΗΛΕΚΤΡΟΝΙΚΗΣ ΥΠΟΒΟΛΗΣ ΕΡΩΤΗΣΕΩΝ)

The notion of network management….

- "autonomous" systems (aka "network"): 100s or 1000s of interacting hardware/software components
- other complex systems requiring monitoring, control:
  - jet airplane
  - nuclear power plant
  - others?

"Network management includes the deployment, integration and coordination of the hardware, software, and human elements to monitor, test, poll, configure, analyze, evaluate, and control the network and element resources to meet the real-time, operational performance, and Quality of Service requirements at a reasonable cost."

Network manager - Διαχειριστής Δικτύου: super-hero or super-engineer?

Απο την απόγνωση στην επίγνωση

Requirements in new generation networks - Οι απαιτήσεις στα σύγχρονα δίκτυα

Support up to 1000 times more traffic

Enable Gbps peak speeds

Improve energy efficiency

Deliver safe superior customer experience

Manage up to 10 times more users

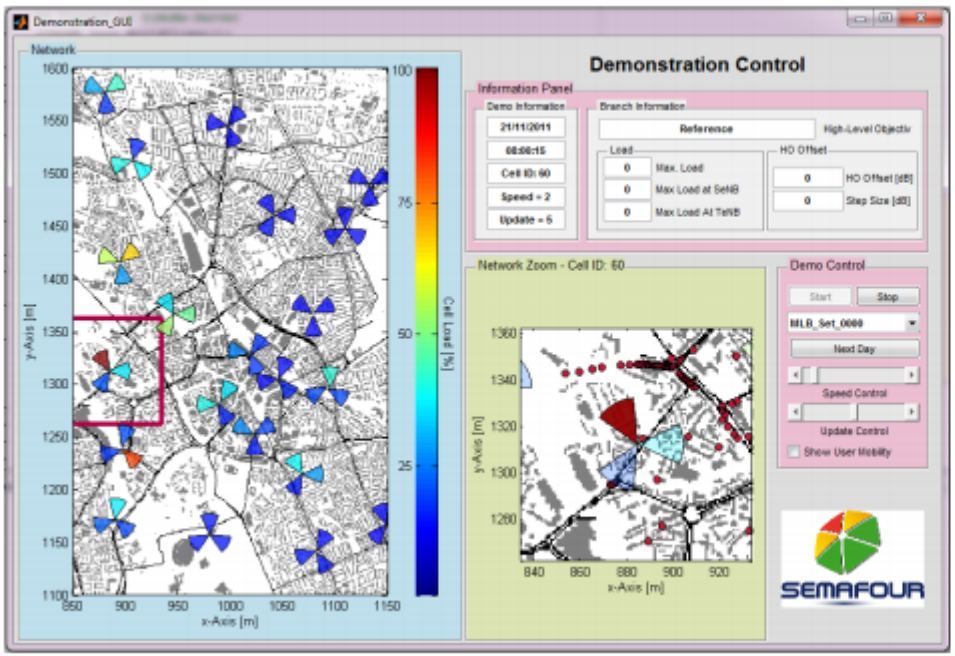
Reduce latency to milliseconds

Make networks self-aware, self-adaptable, and intelligent

Τι είναι (σε γενικές γραμμές) η διαχείριση δικτύου

- Ο απομακρυσμένος έλεγχος και (ανα)διαμόρφωση δικτυακών συσκευών.
- Η παρακολούθηση και βελτιστοποίηση της συμπεριφοράς του δικτύου μέσω των συσκευών που το συνθέτουν.
- Η σύνθεση βάσεων δεδομένων με το “ιστορικό” της δραστηριότητας του δικτύου.
- Η δυνατότητα αυτόματων ειδοποιήσεων, συναγερμών (alarms) στις δικτυακές συσκευές.

Η «εικόνα» ενός εργαλείου διαχείρισης δικτύου



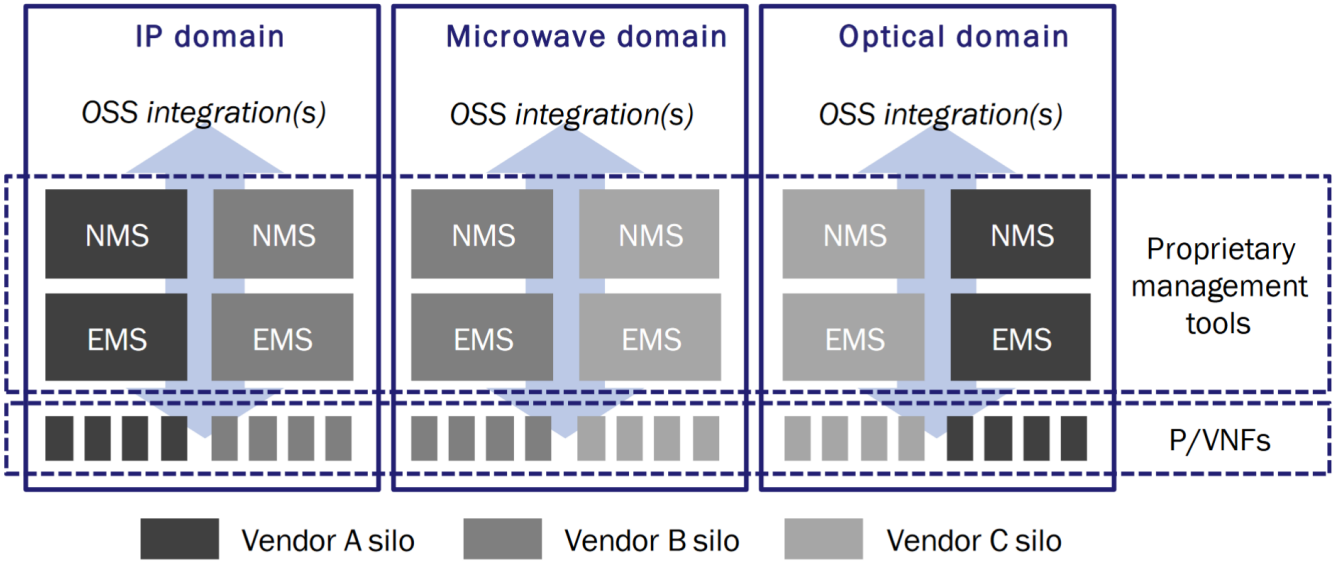
What are we talking about? FCAPS model

- Network Management Tasks
  - fault management
  - configuration management
  - accounting management
  - performance management
  - security management
  - inventory management

Legacy Networks

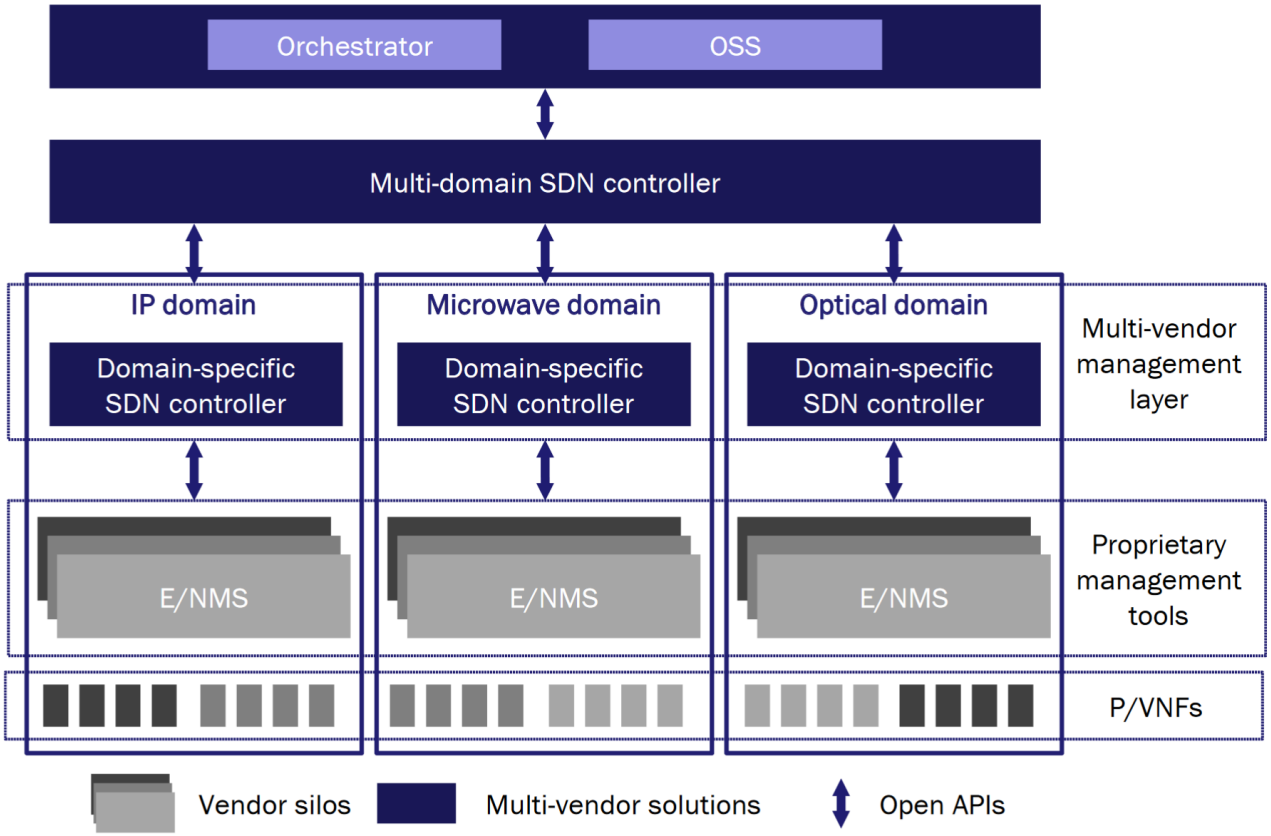
Figure 1 shows an example of a disaggregated network where the CSP has selected different vendors for different solutions and deployed best of breed physical and virtual network functions (P/VNFs) across network domains. Many vendors continue to build their applications without open APIs that require the vendor's own EMSs and NMSs, creating a layer of proprietary management tools made up of individual vendor silos. Each silo will then need to be integrated with the CSP's OSS and will typically result in domain-specific interfaces (for example, CORBA, SNMP, MTOSI, XML, FTP, REST, CLI).

Figure 1: Example of management silos across network domains and different vendors' solutions



Source: Analysys Mason, 2021

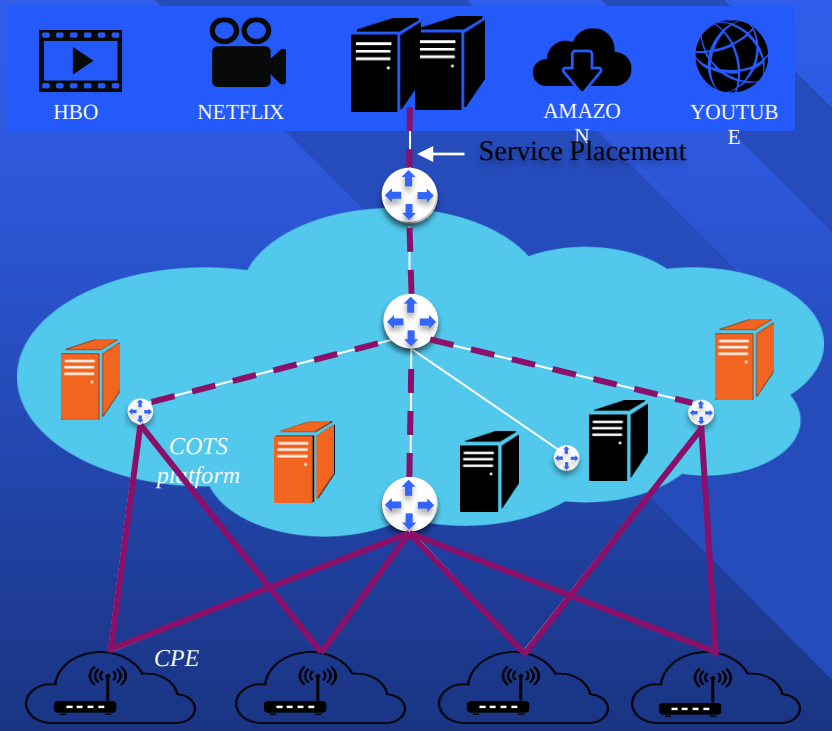
Hierarchical multi-vendor, SDN-based network management architecture for the transport network

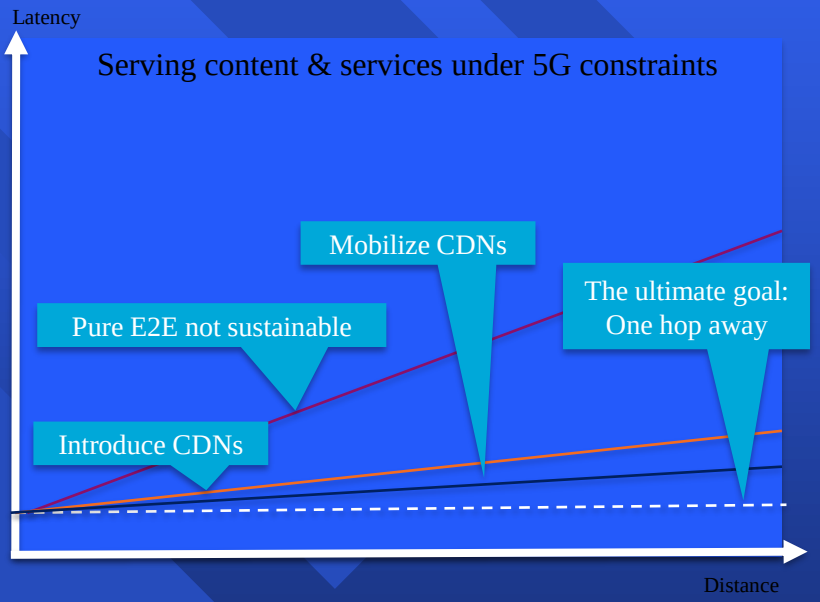


Trends

- **ZERO-TOUCH AUTOMATION**: Intensive Automation is providing rapid provisioning; what used to take weeks and months now takes seconds and minutes. More complex network implementations including reactive network changes, zero downtime upgrades, and automatic threat response.
- Frees up network engineers' time, projects that generate revenue vs. just keeping the lights on. The same person who used to manage 10 network devices can now manage 1,000 network devices or start working on a next-generation, self-tuning monitoring system. The career opportunities for managers and for network engineers become more exciting.

Meeting 5G KPIs
Your Service Just One Hop Away





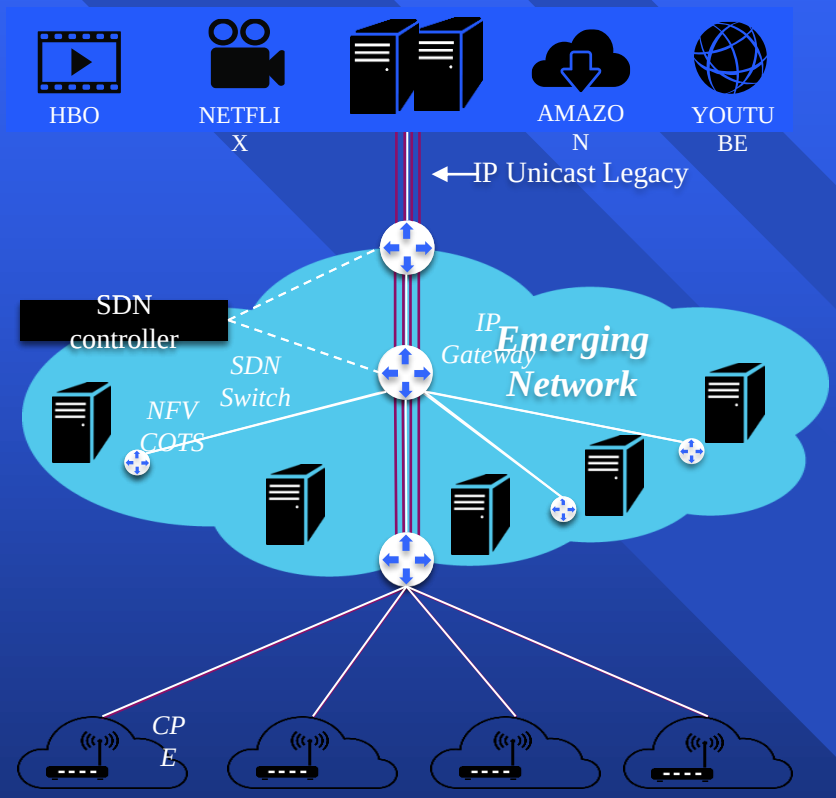
Project Elevator Pitch

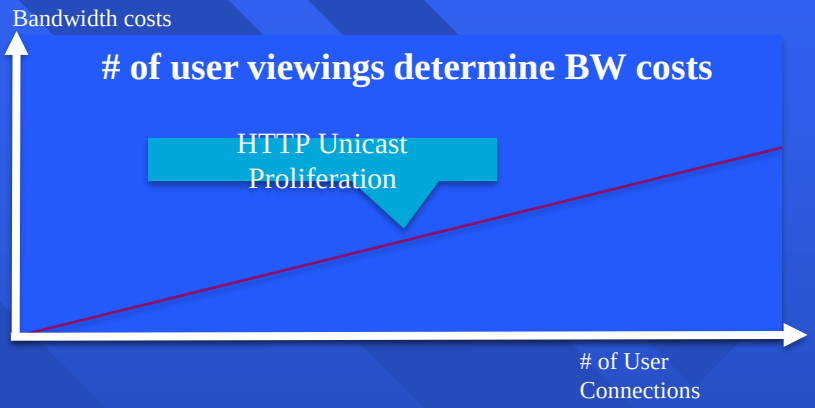
Reinventing the approach to IP based services through a backward compatible introduction of new methodologies supported by an SDN/NFV enabled network fabric & designed to meet challenging 5G KPIs

It looks like IP, it smells like IP, BUT with this technology inside networks will simply work better…

The target for this tech: Telcos & Switch Vendors

The Problem & Current Approaches



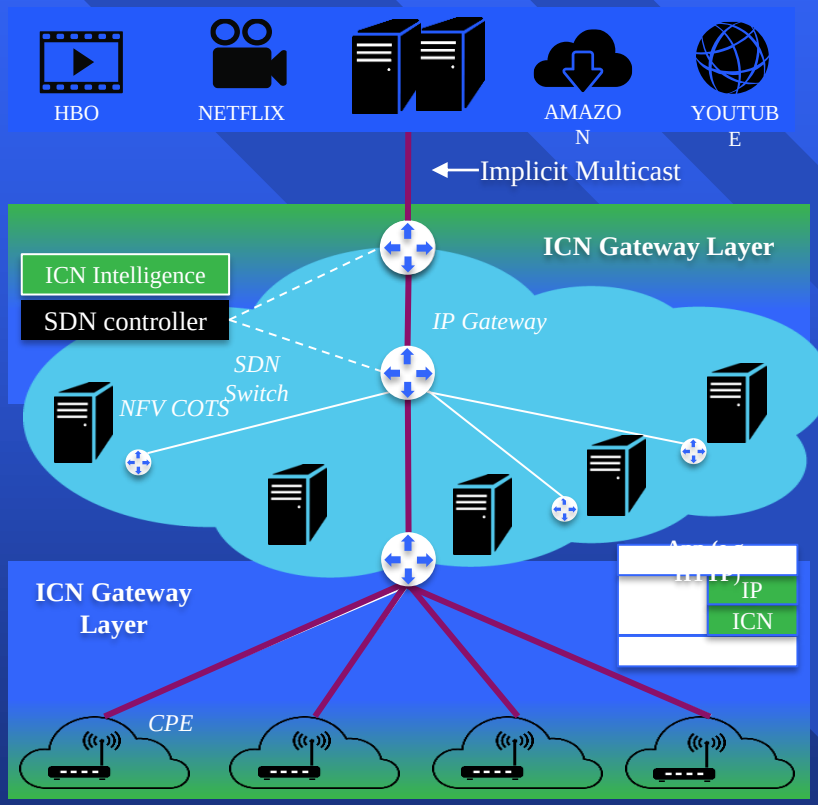


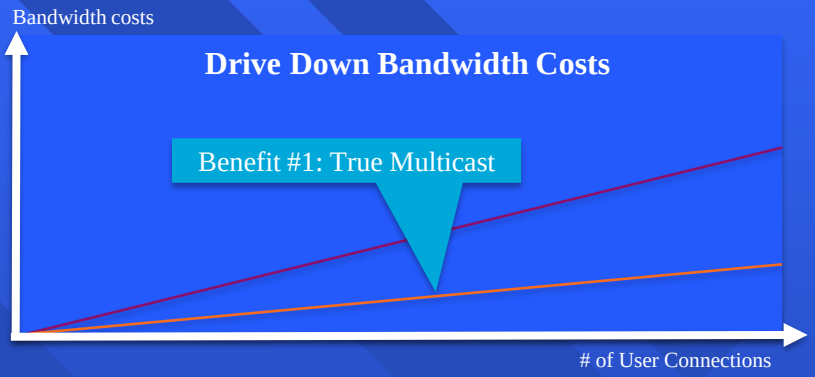
Two current approaches – Two shortcomings

- CDNs are currently used for popular content but this is overly complex and results in inefficiencies associated with indirections
- Overprovisioning of resources drives unsustainable spiraling costs

Both shortcomings are unsustainable for 5G

Re-Introducing Multicast

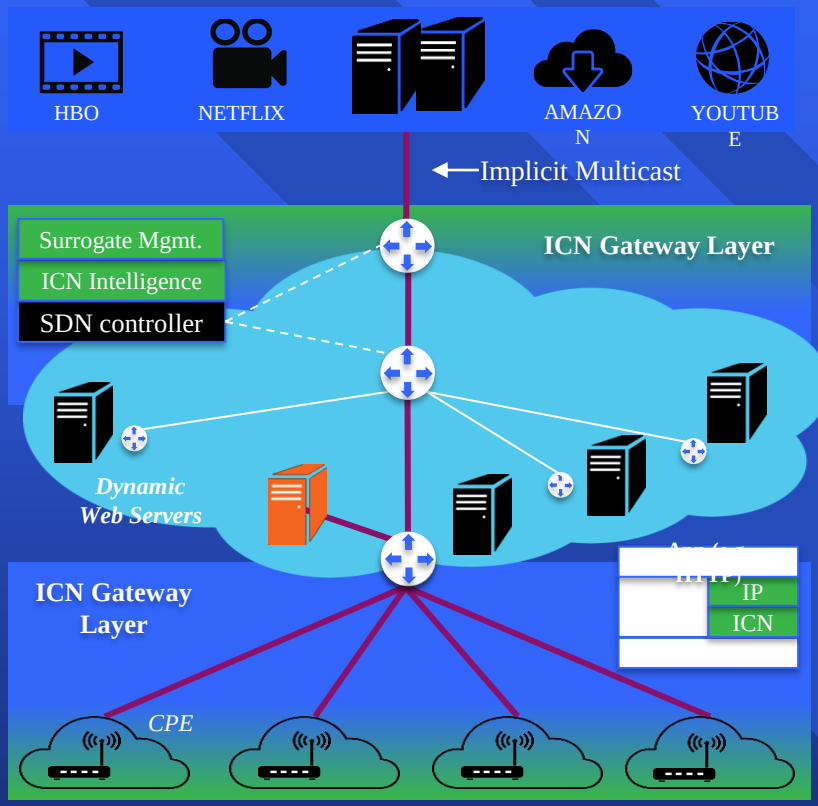


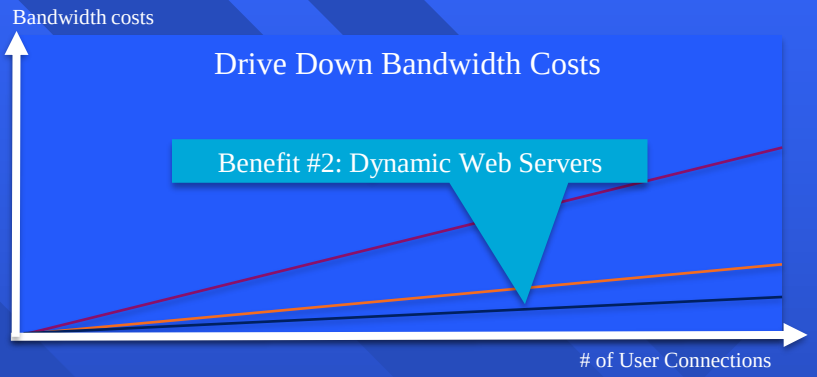


The Innovative ICN approach for competitive 5G (or before) operator networks

- Re-introduce multicast into world of predominantly personalized web experience
  -> **higher network utilization**
- Flexible routing at runtime through *cloudifiable* software elements
  -> **increased resilience, latency reduction**

Localize Communication





The next logical step: Dynamic Web Servers, spun up possibly just one hop away

- Creates new service possibilities for operators, utilizing in-network NFV-based computing capabilities
- Helps meeting challenging 5G KPIs, such as 5ms service-level latency & 1000x capacity increase

Trends

- **IoT and hyper connectivity will fundamentally disrupt traditional Network Management and security safeguards**
- **Network Management and security will ultimately be driven by machine learning and AI.**
- **User experience will be leveraged as a competitive differentiator.** Today, the value of a customer facing service is measured in high availability, security and performance. While these are important, what isn't emphasized is the user experience of that service, but this is because it is difficult to measure. Service providers will begin to quantify user sentiment, which is typically subjective, through the use of Natural Language Processing technology that can interpret human communication channels (e.g. Twitter, Facebook, message boards, etc.) and measure satisfaction.

Trends

- **Deployment of next generation networks**
  - The traditional network is hardware dependent, and runs on fragmented and sometimes inefficient technologies that can result in performance inequalities from location to location
  - Next generation networks, which will largely be software defined and have a management plane will provide IT with the ability to leverage the right network paths, assign appropriate priority to network traffic and ensure the health of network at all locations. These networks will also incorporate an integrated, end-to-end view of the user experience from the data center to the end devices at the edge so that anything that may jeopardize performance is identified and managed before the end user is impacted.
- **Emerging technologies such as augmented and virtual reality, as well as IoT, will drive the need for scaled and automated network management.**

Trends

- Networks becoming
  - More programmatic
  - Defined by owners and operators, not vendors
  - Faster changing, to meet operator needs
  - Lower opex, capex and power
- Abstractions
  - Will shield programmers from complexity
  - Make behavior more provable

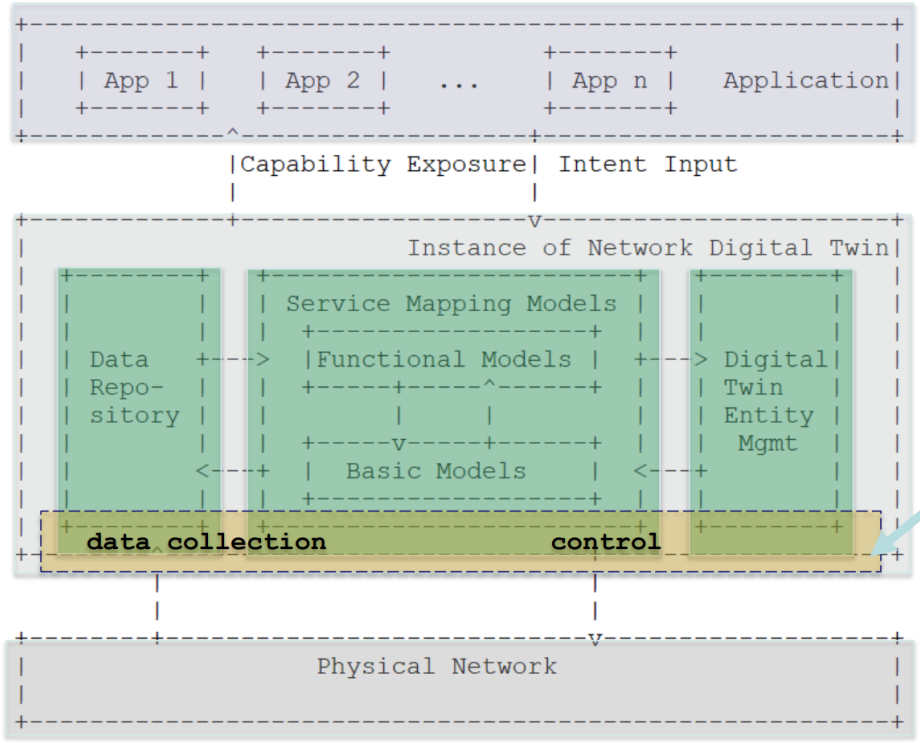
Digital twins

- Tower management/field service management: Various data including proximity, image, touch, temperature, motion and position, can be collected from telecom sites using sensor networks. This data can be fed into a digital twin of the tower to give operations and field service management key information before they go on site. When on site, experts can also assist field workers from the command center by observing the digital twin.

- Network planning and design: Maintaining an accurate inventory of network elements and keeping track of changing configuration has always been a challenge for operators. CSPs use a variety of tools in network modelling, planning, simulation, deployment and operation support activities. A digital twin could bring together all these tool capabilities to provide accurate network inventory and user data from live operations.

- Programmable network DevOps: Every new technology wave (e.g. SDN/NFV, 5G) requires testing the interworking of multiple vendor devices and solutions. A digital twin of the network and associated services together with all functionalities and behaviors could become the DevOps sandbox, where new services are simulated, tested and adjusted before being deployed on the real network.

Refine the Reference Management Architecture



Three-layer DTN reference architecture

- **The Lowest Layer:** Physical Network
- **Top Layer:** Network Application
- **The Intermediate Layer:** Network Digital Twin
  - Core layer of DTN system
  - 3 key subsystems: Data repository, Service mapping models, and Digital Twin entity mgmt
- **Optional sub-layer**

'Data collection' and 'change control' are regarded as southbound interfaces between virtual and physical networks. From an implementation perspective, they can **optionally** form a sub-layer or sub-system to provide common functionalities of data collection and change control, enabled by a specific infrastructure supporting bi-directional flows and facilitating data aggregation, action translation, pre-processing and ontologies.

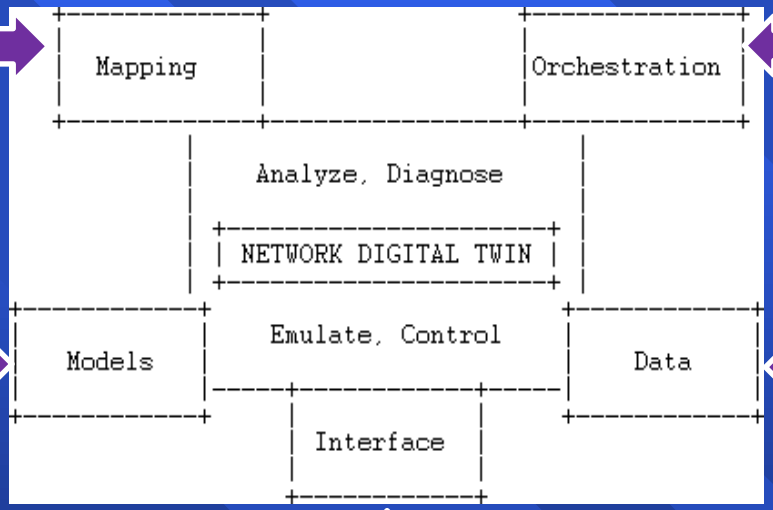
Digital twins

Key Enabling Technologies for Building DTNs

- **Data Collection**
  - Diverse existing tools (e.g., SNMP, NETCONF, Telemetry, INT, etc.) can be used to collect different type of network data
  - Innovative new tools (e.g., sketch-based measurement) can be explored
  - Semantic aggregation mechanisms for data integration and action translation
- **Data storage and services**
  - Unified data repository to effectively store large-scale and heterogeneous network data
  - To provide data services including fast search, batch-data handling, conflict avoidance, data access interfaces, etc.
- **Data Modeling**
  - For small scale network, network simulating tools (e.g., NS-2, GNS3) can be an option
  - For large scale network, low-cost solution is required to create network element and topology models
  - AI/ML can be used to build complex functional models in twin entity.
- **Visualization**
  - Display the network topology, operational status in multiple dimensions and fine granularity
  - The interactive visualizing the execution of models to help users better understand, deduce and explore the network Interfaces and protocols
- **Interfaces**
  - **Twin interfaces** between the physical network and its twin entity: existing interfaces (SNMP, NETCONF, etc.) or new interfaces
  - **Application-facing interfaces** between the network digital twin and applications, e.g., Intent, “what-if” planning app, ...
  - **Internal interfaces** within network digital twin: Interfaces of high-speed, high-efficiency, high-concurrency, etc.

Digital Twin Network Composition

Provides real-time interactive mapping between physical network and virtual twin network or mapping between two virtual twin networks



lifecycle management of components

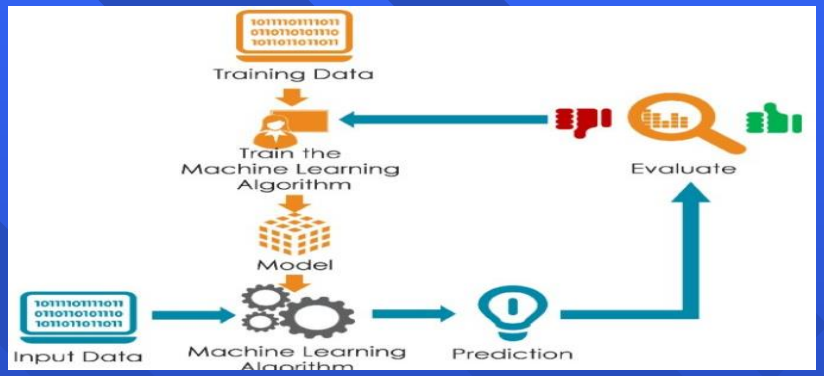
Comprise the models based on physical network simulation, statistical data, performance metrics, inventory information, log information, and Artificial Intelligence

Data about its real-world twin that are required by the models to represent and understand the states and behaviors of the real-world twin

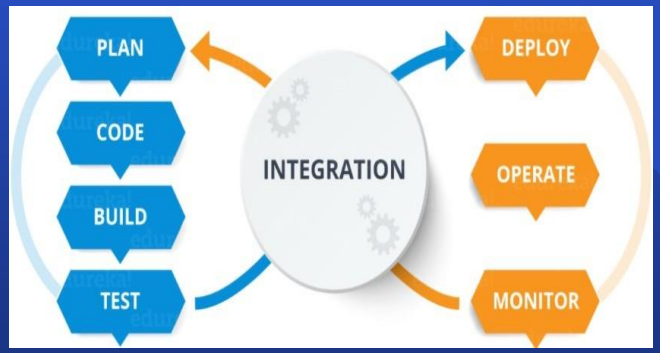
Service interfaces for network applications or other digital twins to access data and invoke capabilities. Telemetry interface for digital twin to populate and cache data

Sample Application Scenarios

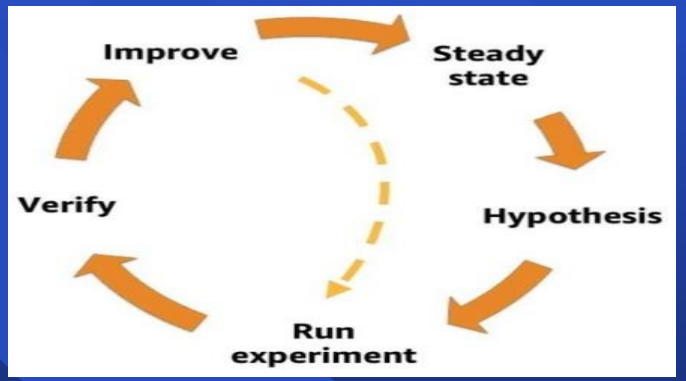
Network Maintenance Engineer Training



Machine Learning Training



DevOps oriented Certification



Network Fuzzing

Issue1: Why distinguish data from model?

- Data and models are usually the two common and separated components/elements in other industrial digital twin entities (e.g., manufacturing, factory-floor).
  - Of course, data can be structured to follow a set of well-known data models.
- Data and Model in digital twin networks
  - *Data*: a digital twin should contain data about its real-world twin (i.e., physical network) that are required by the models to represent and understand the states and behaviors of the real-world twin.
  - *Models*: A digital twin should contain computational or analytic models that are required to describe, understand, and predict the twins' operational states and behaviors, and models that are used to prescribe actions based on service logic and objectives about the corresponding real-world object.
  - In brief, data is cornerstone for constructing a DTN system; and various models are the power and source to Analyze, Diagnose, **Emulate** and Control the physical network.

Issue 2: How Orchestration is different from other components?

Basically orchestration component aims to control and manage the twin entity, then helps to provide an integrated service to various applications.

- Two main orchestration features can be provided: -

1. Control the digital twin network environment and its components to derive the required/expected behavior
2. Deal with the dynamic lifecycle management of these components by providing

    - Repeatability: Replicate network conditions on demand
    - Reproducibility: Replay successions of events, possibly under controlled variations

Issue 3: How should the interfaces be defined?

- Three types of interfaces were identified:
  1) Twin interface: between the physical network and its twin entity
     - It can be implemented using a variety of existing tools (telemetry, SNMP, NETCONF, etc.) or new ones.
  2) Application-facing interface: between the network digital twin and applications that make use of the emulated network.
     - For example, Intent, “what-if” planning app, ...
  3) Internal interfaces between components within network digital twin
- We need to first define or choose the first two types of interfaces, then focus on the internal interfaces to build the twin image.
  - The first two interfaces should be open, standardized, real-time, secure, and reliable.
  - Internal interfaces should be with capability of high-speed, high-efficiency, high-concurrency etc.

Issue 4: Which component is responsible for checking for deviation of the underlay network vs. the image?

- Mapping component is responsible for such checking
- From traditional simulation to emulation, with real-time interactive mapping.
  - Digital twin network provides **real-time interactive mapping** between physical network and virtual twin network, that **emulates** the behavior of a network by calculating the deviation between the different network entities (routers, switches, nodes, access points, links, etc.) in the physical network and corresponding entities in the virtual twin network.

- Mapping can be:
  - One to one mapping (pairing, vertical): Synchronize between a physical network and its virtual twin network **with continuous flow**
  - One to many mapping (coupling, horizontal): Synchronize among virtual twin networks with **occasional data exchange**

Issue 5: Continuous Verification vs CI/CD

- Modern DevOps practices involve continuous development/testing/integration, /deployment/monitoring of software applications throughout its development life cycle:
  - ***Continuous Integration (CI)*** allow implement small changes and check in code to version control repositories frequently.
    - E.g., committing all your application code in a single repository
  - ***Continuous Delivery (CD)*** automates the delivery of applications to selected infrastructure environments. (e.g., network digital twin)
    - E.g., Travis CI allows automatically run CI tasks like unit tests and push your code to a hosting platform every time you push new changes to a branch.

- Continuous Verification (CV) is an extension of DevOps practices that are concerned with verifying the system as a whole.
  - The application of CI/CD models in network management operations increases the risk associated to deployment of non-validated updates
  - CV can be used in **DevOps-oriented certification application** to address it.

Τι προσφέρει η διαχείριση δικτύων

- Proactive:
  - remote configuration.
  - network profiling.
  - modeling changes to the network.
- Reactive:
  - Faults - ειδοποίηση για προβλήματα στο δίκτυο.
  - Diagnosis -διάγνωση προβλημάτων.
  - Reconfiguration -αυτόματη διαμόρφωση του δικτύου σε περίπτωση σφάλματος.
- Interactive:
  - interactive troubleshooting

Τι standards υπάρχουν

- ISO: CMIP και CMIS.
- IAB (Internet Architecture Board): SNMP, SNMPv2, RMON, CMOT.
- IEEE: CMOL.
- Το SNMP επικράτησε κυρίως λόγω της απλότητας στην υλοποίηση και την διαχείριση.
  - 1988: SNMPv1.
  - 1993: SNMPv2 (updated το 1996).
  - 1998: SNMPv3 (draft από IESG: RFC 2570-2575).

Network Management standards

OSI CMIP

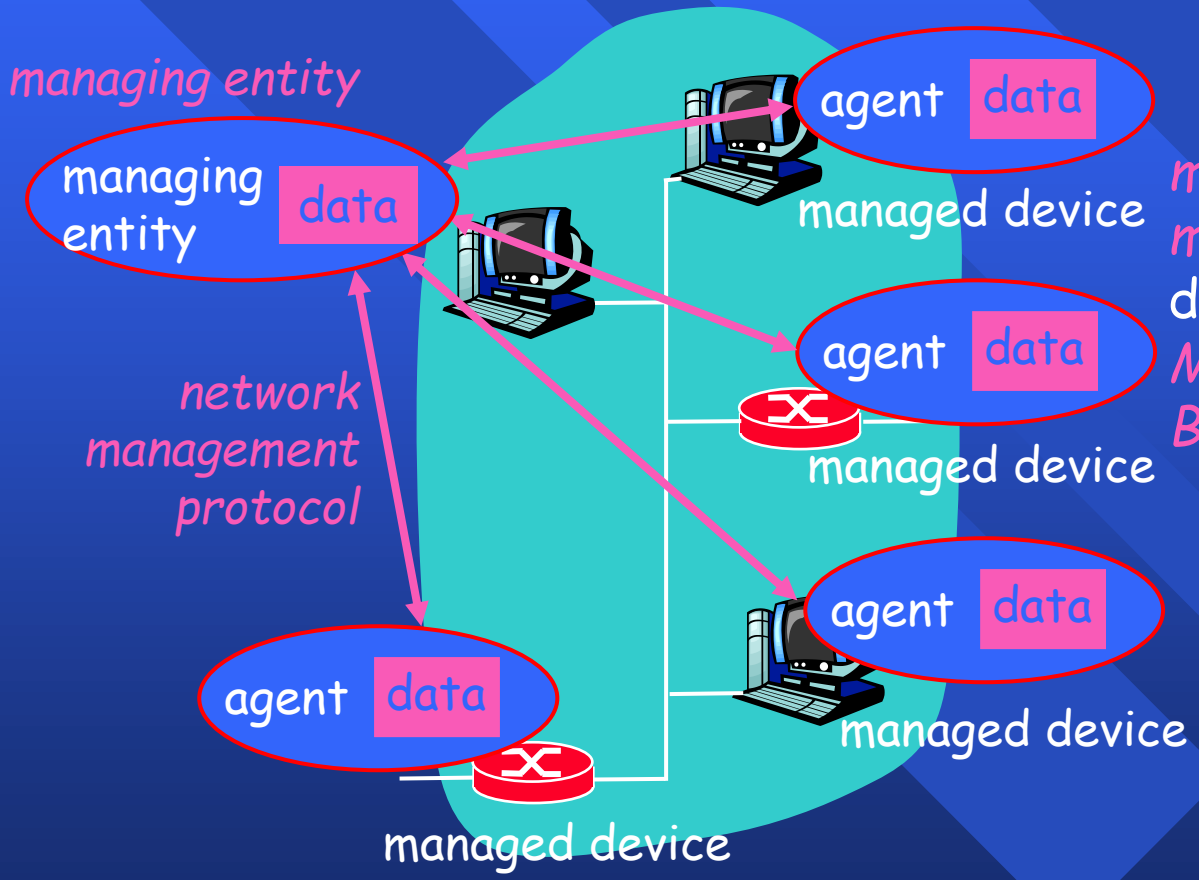
- Common Management Information Protocol
- designed 1980's: *the* unifying net management standard
- too slowly standardized

SNMP: Simple Network Management Protocol

- Internet roots (SGMP)
- started simple
- deployed, adopted rapidly
- growth: size, complexity
- currently: SNMP V3
- *de facto* network management standard

Infrastructure for network management

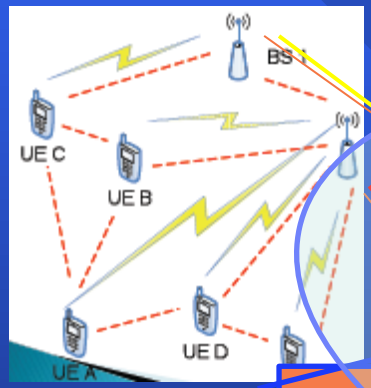
definitions:



managed devices contain *managed objects* whose data is gathered into a *Management Information Base (MIB)*

Heterogeneous Networks

Mobile/wireless



Sensors / actuators, Cooperating object networks

Personal space

Seamless service – content networks, new traffic requirements

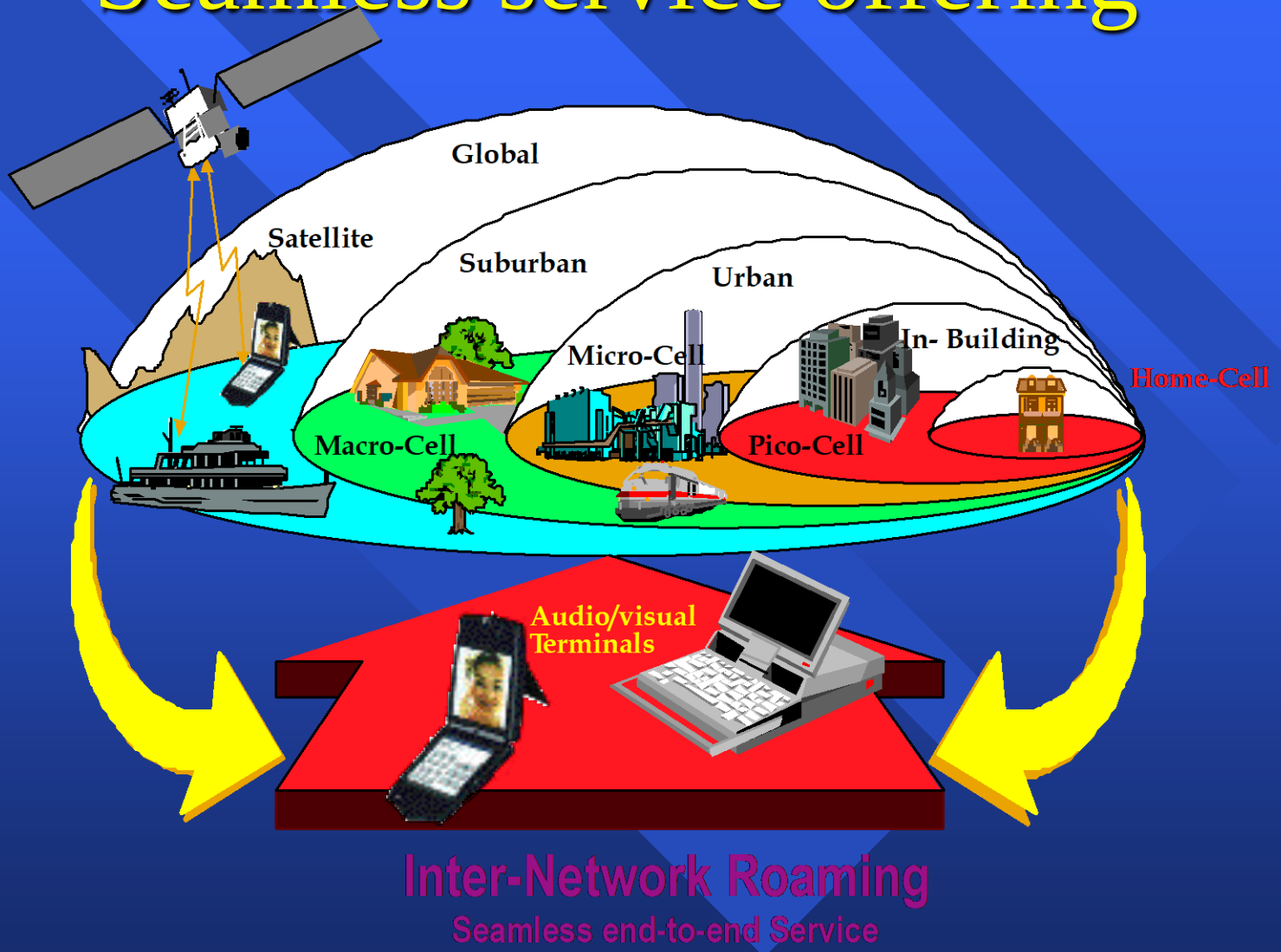
Virtualised Network, Stack A

Virtualised Network, Stack B

Virtualised Network, Stack C

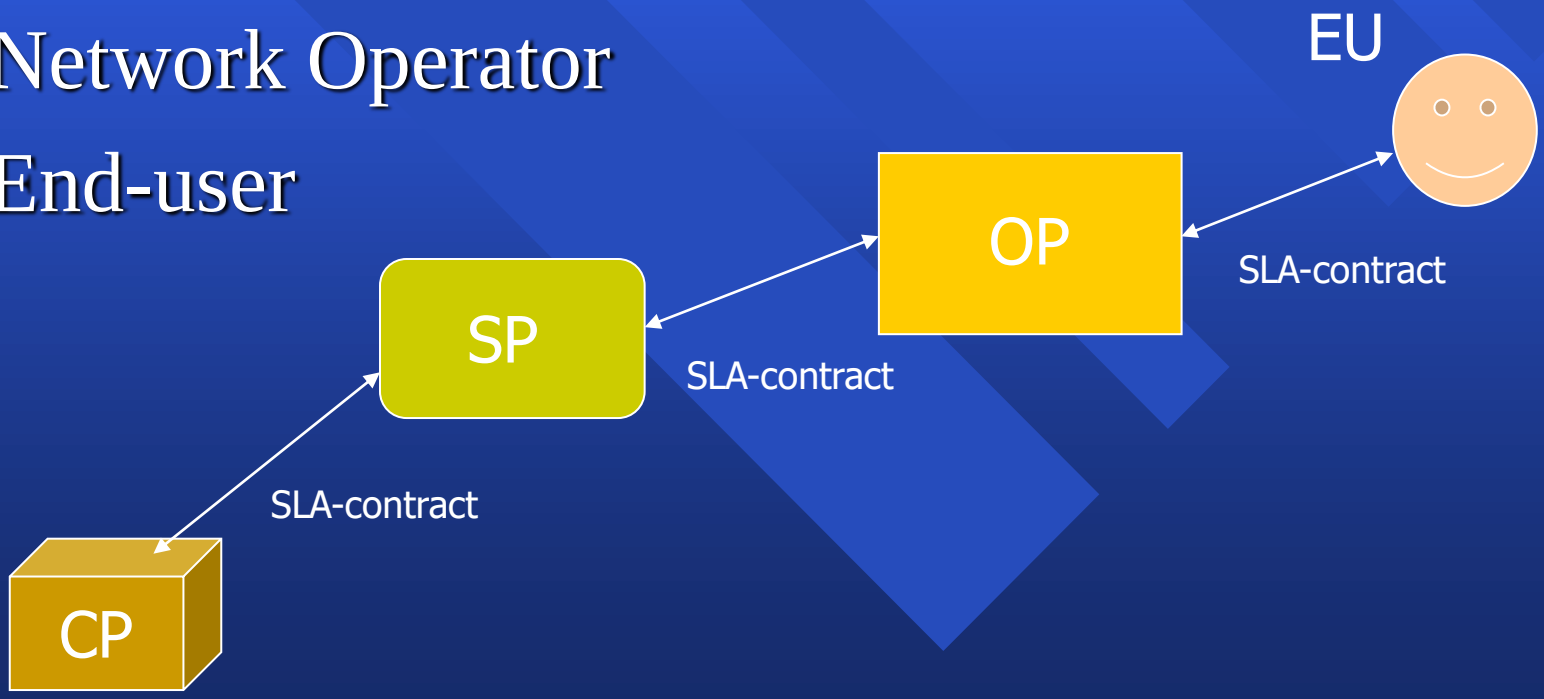
Mesh
Relay
ad-hoc

Seamless service offering



Value chain-Βασική αλυσίδα αξίας επιχειρηματικού μοντέλου για παροχή υπηρεσιών

- Content provider
- Service/Application developer/provider
- Network Operator
- End-user



Service Level Agreement

- Ενα service level agreement είναι ένα κείμενο που καθορίζει τις σχέσεις μεταξύ δύο μερών: του προμηθευτή και του καταναλωτή (the provider and the recipient).
- Προδιαγράφει το πλαίσιο συνεργασίας, αποτροπής προβλημάτων, εγγύησης των προδιαγεγραμμένων υπηρεσιών και της σχετικής ποιότητας, το πλαίσιο επίλυσης διαφορών που μπορεί να προκύπτουν κατά τη χρήση των υπηρεσιών κ.α.

Service Level Agreement

- SLA should embrace a wide range of issues. Amongst these are usually the following:
  - Services to be delivered
  - Performance, Tracking and Reporting
  - Problem Management
  - Legal Compliance and Resolution of Disputes
  - Customer Duties and Responsibilities
  - Security
  - IPR and Confidential Information
  - Fees and expenses
  - Termination

What is network management?

- **System & Service monitoring**
  - Reachability, availability
- **Resource measurement/monitoring**
  - Capacity planning, availability
- **Performance monitoring (RTT, throughput)**
- **Statistics & Accounting/Metering**
- **Fault Management (Intrusion Detection)**
  - Fault detection, troubleshooting, and tracking
  - Ticketing systems, help desk
- **Change management and configuration monitoring**

Getting started

Make sure that the network is up and running. Thus, we need to monitor it:

- Deliver projected SLAs (Service Level Agreements)
- Depends on policy
  - What does your management expect?
  - What do your users expect?
  - What do your customers expect?
  - What does the rest of the Internet expect?
- Is 24x7 good enough?
  - There's no such thing as 100% uptime (as we'll see) →

Getting started: “Uptime”

What does it take to deliver 99.9 % uptime?

30.5 x 24 = 762 hours a month

(762 – (762 x .999)) x 60 = 45 minutes
only 45 minutes of downtime a month!

Need to shutdown 1 hour / week?

(762 - 4) / 762 x 100 = 99.4 %

Remember to take planned maintenance into account in your calculations, and inform your users/customers if they are included/excluded in the SLA

How is availability measured?

In the core? End-to-end? From the Internet?

Getting started: Baselining

What is normal for your network?

If you've never measured or monitored your network you need to know things like:

- Load on links
- Jitter between endpoints
- Percent usage of resources
- Amount of "noise":
  - » Network scans
  - » Dropped data
  - » Reported errors or failures

Why network management?

Know when to upgrade

- Is your bandwidth usage too high?
- Where is your traffic going?
- Do you need to get a faster line, or more providers?
- Is the equipment too old?

Keep an audit trace of changes

- Record all changes
- Makes it easier to find cause of problems due to upgrades and configuration changes

Keep a history of your network operations

- Using a ticket system let you keep a history of events.
- Allows you to defend yourself and verify what happened

Why network management?

Accounting

- Track usage of resources
- Bill customers according to usage

Know when you have problems

- Stay ahead of your users! Makes you look good.
- Monitoring software can generate tickets and automatically notify staff of issues.

Trends

- All of this information can be used to view trends across your network.
- This is part of baselining, capacity planning and attack detection.

Attack Detection

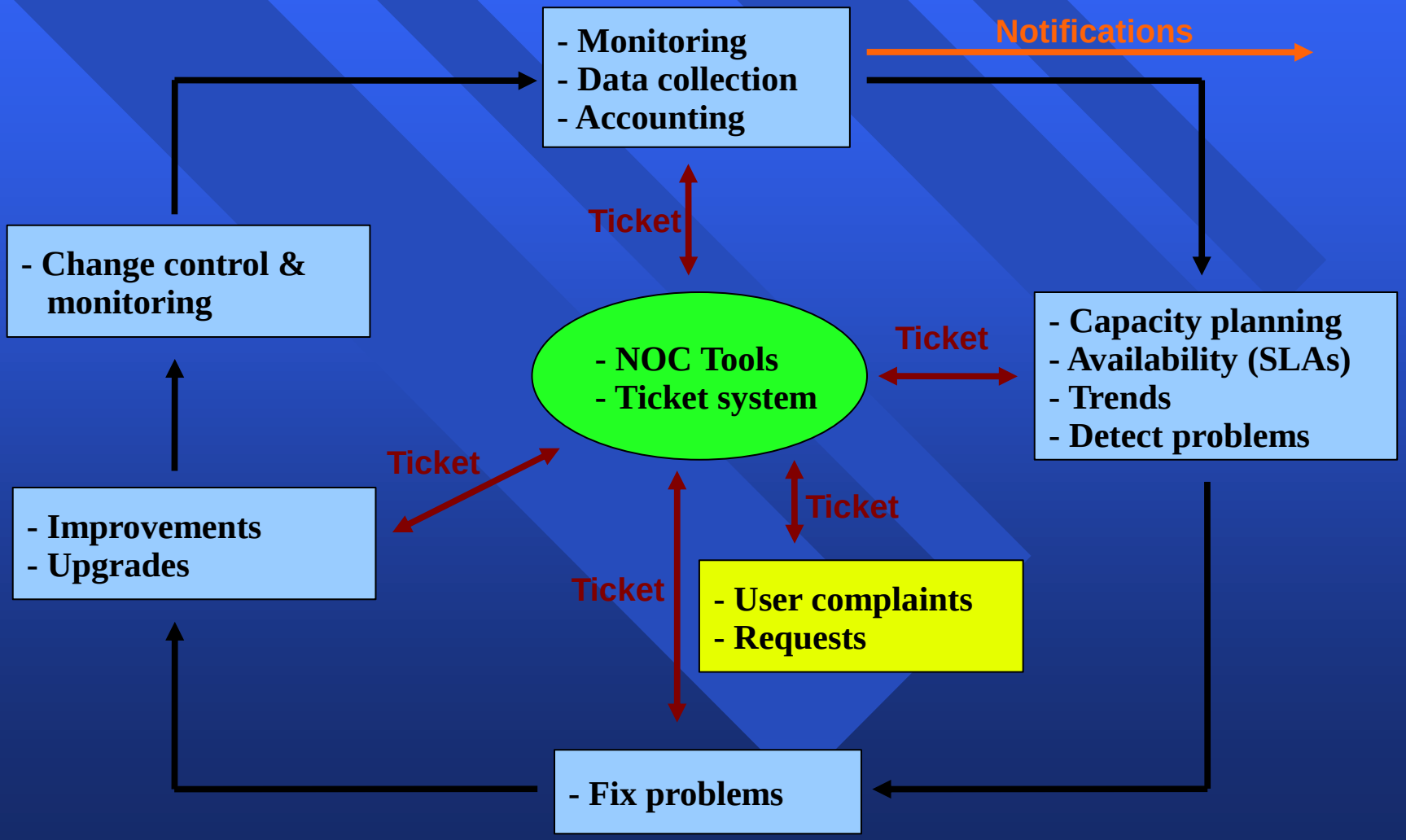
- Trends and automation allow you to know when you are under attack.
- The tools in use can help you to mitigate attacks:
  - Flows across network interfaces
  - Load on specific servers and/or services
  - Multiple service failures

Consolidating the data

The Network Operations Center (NOC) "Where it all happens"

- Coordination of tasks
- Status of network and services
- Fielding of network-related incidents and complaints
- Where the tools reside ("NOC server")
- Documentation including:
  - Network diagrams
  - database/flat file of each port on each switch
  - Network description
  - Much more as you'll see a bit later.

The big picture



The notion of Future Internet management

- The design of Future Internet network elements with the aim of mastering the increasing complexity of communication networks
- The network should be capable of real-time, secure and cost-effective delivery of data. It is of utmost importance to increase the user's perceived quality of life anywhere and anytime.
  - human-to-human
  - human-to-machine
  - machine-to-machine

Network Management
Basics

- Network management requirements
- OSI Management Functional Areas
  - Network monitoring: performance, fault, accounting
  - Network control: configuration, security
- Standardization in network management

- Practical issue: introduction to SNMP

Network Management
Requirements

Example of approach

- Controlling strategic assets
- Controlling complexity
- Improving service
- Balancing various needs: performance, availability, security, cost
- Reducing downtime
- Controlling costs

What are we talking about? FCAPS model

- Network Management Tasks
  - fault management
  - configuration management
  - accounting management
  - performance management
  - security management
  - inventory management

Network Management
OSI functional areas

- Fault management
  - Detect the fault
  - Determine exactly where the fault is
  - Isolate the rest of the network from the failure so that it can continue to function
  - Reconfigure or modify the network in such a way as to minimize the impact
  - Repair or replace the failed components

  - Tests: connectivity, data integrity, response-time, ….

Fault Management

- detection
- exception alarm generation
- investigation and analysis
- statistics for steady state behaviour characterisation

Fault Management Sub-categories

Prioritization	• Prioritize faults in the order in which they should be addressed • Use in-band management packets to learn about important faults • Identify which fault events should cause messages to be sent to the manager • Identify which devices should be polled and at what intervals • Identify which device parameter values should be collected and how often • Prioritize which messages should be stored in the manager's database
Timeliness Required	• Management Station is passive and only receives event notifications • Management Station is active and polls for device variable values at required intervals • Application periodically requests a service from a service provider
Physical Connectivity Testing	• Using a cable tester to check that links are not broken
Software Connectivity Testing	• Using an application that makes a request of another device that requires a response. ❑ The most often application for this is Ping.Exe. It calls the Internet Control Message Protocol (ICMP) which sends periodic Echo Request messages to a selected device on a TCP/IP network ❑ Application on one device makes a request of an application on another device
Device Configuration	• Devices are configured conservatively to minimize chances of dropped packets.
SNMP Polls	• Devices are periodically polled to collect network statistics
Fault Reports Generated	• Thresholds configured and alarms generated • Text media used for report • Audio media used for report • A color graphical display used to show down devices • Human manager is notified
Traffic Monitored	• Remote Monitors used • Protocol analyzers used • Traps sent to Network Management Station • Device statistics monitored
Trends	• Graphical trends generated to identify potential faults

Fault Management notification cycle

Understanding the need for Fault Management

Fault management is usually mentioned as the first concern in network management.

Its main role is to ensure high availability of a network

Hence, involving a procedure to anticipate and avoid network failures

In the case where a failure cannot be avoided, the necessary steps are required to contain the damage and resolve the effects on the network

Defining fault management

The goal of Fault Management

The goal of *fault management*

- detect,
- log,
- notify users of, and (to the extent possible)
- automatically fix

network problems to keep the network running effectively.

Because faults can cause downtime or unacceptable network degradation, fault management is perhaps the most widely implemented of the ISO network management elements.

Fault Management

- Fault management involves first.

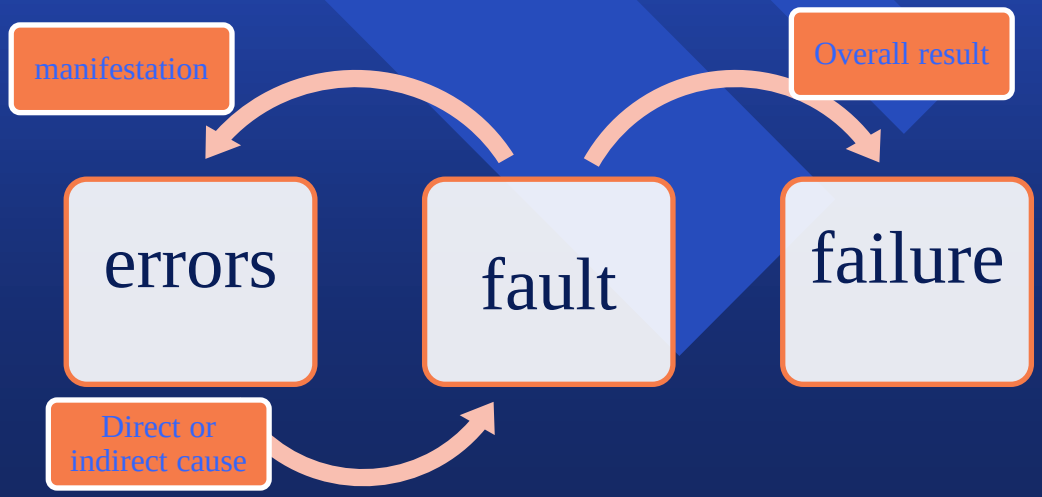
First,
- determining symptoms and isolating the problem

Second,
- Then the problem is fixed and the solution is tested on all-important subsystems

Finally,
- The detection and resolution of the problem is recorded.

FAULT MANAGEMENT – Defining the terms

- A *fault* is a software or hardware defect in a system that disrupts communication or degrades performance
- An *error* is the incorrect output of a system component. If a component presents an error, we say the component fails → This is a *component failure*



Fault symptoms can be associated to four types of error

These symptoms may take one of the following forms:

These symptoms may take one of the following forms:

timing error	*An output with an expected value comes either too early or too late*
timely error	*An output with an unexpected value within the specified time interval*
commission error	*An output with an unexpected value outside the specified time interval → response produced*
omission error	*An output with an unexpected value outside the specified time interval → no response is produced*

network faults can be generally divided according to their duration into three groups:
permanent, intermittent and transient:

1st: A permanent fault v st in the network until repair action has beer his results in permane maximum degradati rvice

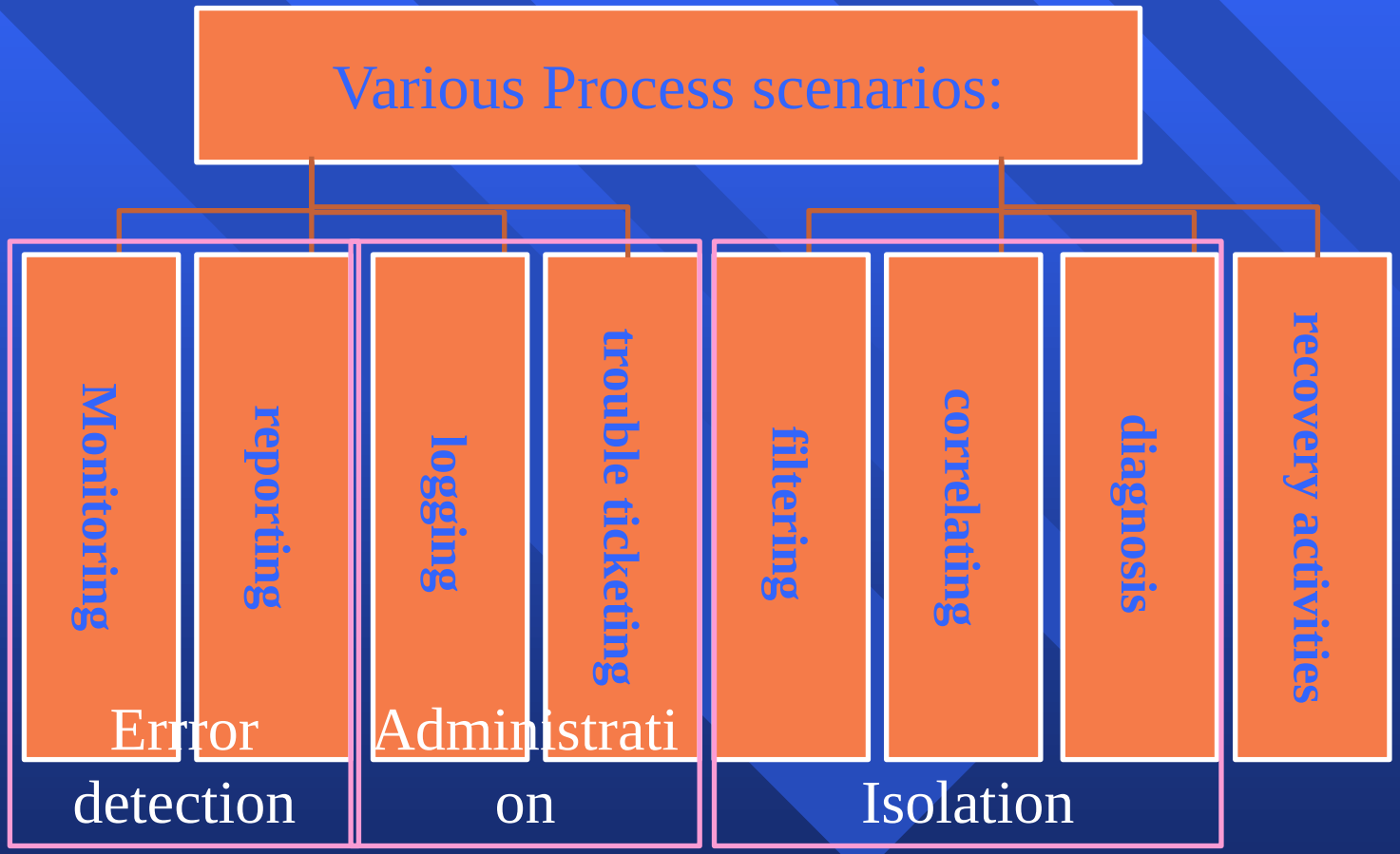
2nd :An intermittent fault occurs in a discont d a periodic manner. Its outcome will be failur current processes. This implies maximum de of the service level for a short period of time.

3rd :A transient fault will momentarily cause a minor degradation of the service.

FAULT:

- For faults of the first type, it will cause an event report to be sent out and changes made in the network configuration to prohibit further utilisation of this resource.
- For a fault of the second type, the severity of the fault may transfer from being intermittent to being permanent if an excessive occurrence of this kind of fault becomes significant.
- For a fault of the third type, it will usually be masked by the error recovery procedures of network protocols and therefore may not be observed by the users.

Fault Management Process



Typical fault

- **Observable fault**
- **Unobservable fault**
  - For example, the existence of a deadlock between co-operating distributed processes may not be observable locally.
  - Other faults may not be observable because the vendor equipment is not instrumented to record the occurrence of a fault
- **Too many related observation**:
  - A single failure can affect many active communication paths. The failure of a WAN back-bone will affect all active communication between the token-ring stations and stations on the Ethernet LANs, as well as voice communication between the PBXs.
- **Propagation of failures**
  - a failure in one layer of the communications architecture can cause degradation or failures in all the dependent higher layers.

Network Management
OSI functional classification

- Performance management:
  - What is the level of capacity (χωρητικότητα) utilization?
  - Is there excessive traffic?
  - Has throughput been reduced to unacceptable levels?
  - Are there bottlenecks?
  - Is response time increasing?

  - Indicators: availability, response time, accuracy ← service
    throughput, utilization ← efficiency

Performance Management

- Availability and Reliability metrics
- Quality metrics
- real-time measurement
- historical analysis

Performance Management Sub-Categories

Collecting Baseline Utilization Data	• Measuring link utilization using a probe • Counting packets received/transmitted by a specific device • Measuring device processor usage • Monitoring device queue lengths • Monitoring device memory utilization • Measuring total response times
Collecting a History of Utilization Data	• Measuring utilization and response times at different times of the day • Measuring utilization and response times on different days over an extended period
Capacity Planning	• Manually graphing or using a network management tool to graph utilization as a function of time to detect trends • Preparing trend reports to document projected need for and the cost of network expansion.
Setting Notification Thresholds	• Having a network management tool poll devices for values of critical parameters and graphing these values as a function of time • Setting polling intervals • Setting alarms/alerts on those parameters when the threshold is reached or a percentage of it is reached • Initiating an action when the threshold is reached such a sending a message to the network manager.
Building Databases	• Having the network management tool create a database of records containing device name, parameter, threshold and time for off-line analysis. • Using the database to extract time dependence of utilization • Using the time dependence of parameters to decide when network upgrades will be necessary to maintain performance
Running Network Simulations	• Using a simulation tool to develop a model of the network • Using the model's parameters and utilization data to optimize network performance
Latency	• Query/Response time interval

Network Management
OSI functional classification

- Configuration and Name Management:
  - Installation of new hardware/software
  - Tracking changes in control configuration
  - Who, what and why? - network topology
  - Revert/undo changes
  - Change management
  - Configuration audit
  - Does it do what was intended

Configuration Management

- installation of new hardware/software
- tracking changes in control configuration
  - who, what and why!
- revert/undo changes
- change management
- configuration audit
  - does it do what was intended?

Configuration Management Sub-categories

Configuration (Local)	• Choice of medium access protocol • Choice of correct cabling and connectors • Choice of cabling layout • Determining the number of physical interfaces on devices • Setting device interface parameter values ❑ Interrupts ❑ I/O Addresses ❑ DMA numbers ❑ Network layer addresses (e.g. IP, NetWare, etc) • Configuration of multiport devices (e.g. hubs, switches and routers) • Use of the Windows Registry • Comparing current versus stored configurations • Checking software environments • SNMP service
Configuration (Remote)	• From the network management station • Disabling device ports • Redirecting port forwarding • Disabling devices • Comparing current versus stored configurations • Configuring routing tables • Configuring security parameters such as community strings and user names • Configuring addresses of management stations to which traps should be sent • Verifying integrity of changes

Configuration Management Sub-categories

Configuration (Automated)	• Using the Dynamic Host Configuration Protocol (DHCP) to configure IP addresses • Using Plug and Play enabled NICs for automatic selection of interrupts and I/O addresses • Domain Name Services (DNS) addresses • Trap messages from agents
Inventory (Manual)	• Maintaining records of cable runs and the types of cables used • Maintaining device configuration records • Creating network database containing for each device: • Device types ❑ Software environment for each device ❑ operating systems ❑ utilities • drivers • applications ❑ versions ❑ configuration files (.ncf, .ini, .sys) • vendor contact information • IP address • Subnet address
Inventory (Automated)	• Auto-discovery of devices on the network using an NMS • Auto-determination of device configurations using an NMS • Creation of a network database • Auto-mapping of current devices to produce a network topological map • Accessing device statistics using an NMS and the Desktop Management Protocol

Network Management
OSI functional classification

- Security management
  - Security services: generating, distributing, storing of encryption keys for services
  - Exception alarm generation, detection of problems
  - Uniform access control to resources
  - Backups, data security
  - Security logging

Security Management

- exception alarm generation
- detection
- uniform access controls to resources
- backup

Security Management Sub-categories

Applying Basic Techniques	• Identifying hosts that store sensitive information • Management of passwords • Assigning user rights and permissions • Recording failed logins • Setting remote access barrier codes • Employing virus scanning • Limiting views of the Enterprise network • Tracking time and origin of remote accesses to servers
Identifying Access Methods Used	• Electronic Mail • File Transfer • Web Browsing • Directory Service • Remote Login • Remote Procedure Call • Remote Execution • Network Monitors • Network Management System
Using Access Control Methods	• Encryption • Packet filtering at routers • Packet filtering at firewalls • Source host authentication • Source user authentication
Maintenance	• Audits of the activity at secure access points • Executing security attack programs (Network Intrusion Detection) • Detecting and documenting breaches
Accessing Public Data Networks	• No restrictions - hosts are responsible for securing all access points • Limited access - only some hosts can interface with the Public Data Network using a proxy server
Using an Automated Security Manager	• Queries the configuration database to identify all access points for each device. • Reads event logs and notes security-related events. • Security Manager shows a security event on the network map. • Reports of invalid access point attempts are generated daily for analysis

Network Management - FCAPS
OSI functional classification

- Accounting management
  - Identifying consumers and suppliers of network resources - users and groups
  - Mapping network resources consumption to customer identity
  - Billing

Accounting Management

- identifying consumers and suppliers
  - of network resources
- mapping network resources to customer identity
- charge back
  - volumetric data
  - time data
  - date time of day

Accounting Management Sub-categories

Gather Network Device Utilization Data	• Measure usage of resources by cost center • Set quotas to enable fair use of resources • Site metering to track adherence to software licensing
Bill Users of Network Resources	• Set charges based on usage. • Measure one of the following ❑ Number of transactions ❑ Number of packets • Number of bytes • Set charges on direction of information flow
Use and Accounting Management Tools	• Query usage database to measure statistics versus quotas • Define network billing domains • Implement automatic billing based on usage by users in the domain • Enable billing predictions • Enable user selection of billing domains on the network map
Reporting	• Create historical billings trends • Automatic distribution of billing to Cost Centers • Project future billings by cost center

IP Route Management

- routing integrity (ακεραιότητα δρομολόγησης)
- consistency with customer requirements
- consistency with external peers
- conformance with imposed policy constraints

Problem Tracking

- reporting procedures (διαδικασίες αναφορών)
- fault management (διαχείριση σφαλμάτων)
- escalation and referral (κλιμάκωση και προσφυγή)
- historical data for component reliability analysis

Inventory Control (έλεγχος απογραφής δικτυακών στοιχείων)

- hardware
  - components
  - identity
  - location
- software
  - version control

Knowledge Based Management

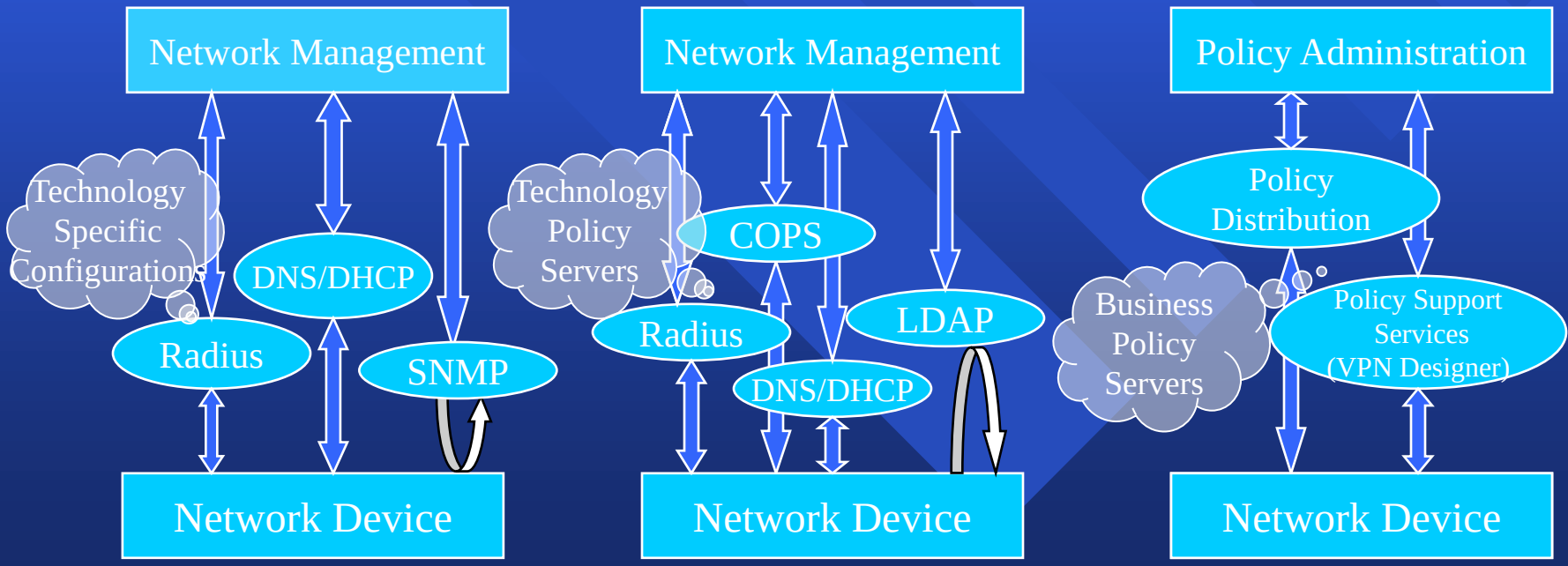
- "expert" systems
- Modelling
  - simulation
  - routing
  - configuration changes

Evolving to Next Generation Network Management

LEGACY
- Independent device and independent services management
- Table-driven device functions
- Client(NM)-Server(Device) architecture
- SNMP

MAIN APPROACH
- Directories drive data unification
- Central policy management on service basis
- Dynamic device functions
- Policy agents added

POLICY-BASED
- Distributed policy management
- Integrated services through policies
- Reactive agents added
- Complex & reactive policy capabilities



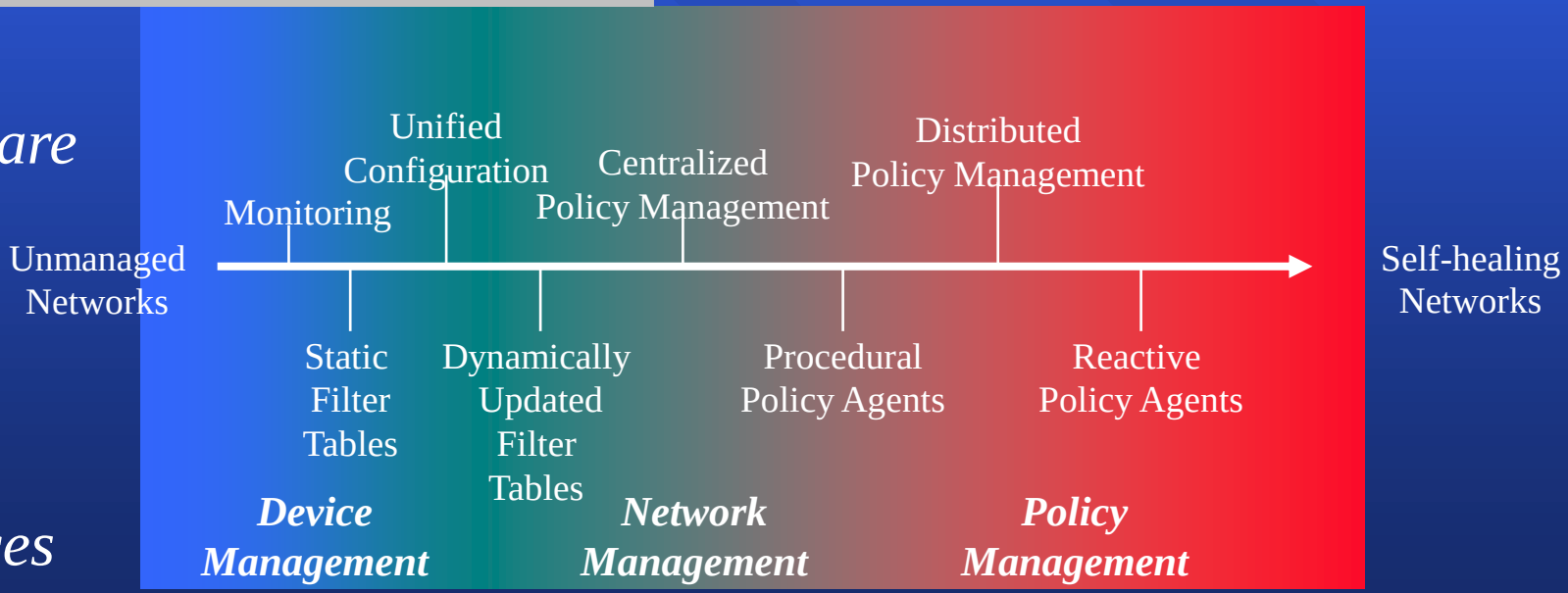
Complex Networks and New Dynamic Services Drive Changes to Policy Management and Infrastructure

Issues

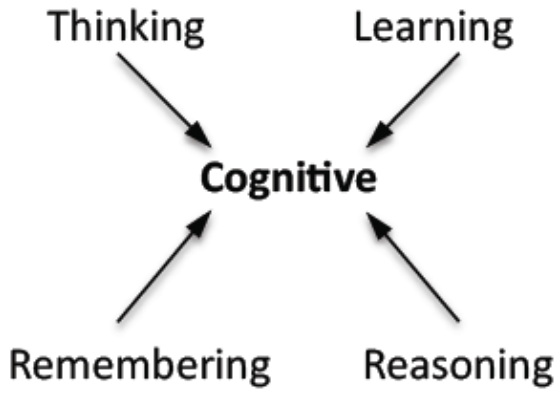
- Management is device configuration; needs to be offer & service related
- Associated data is per device per vendor and largely in tables; needs to be integrated and for the offer or service
- Data inconsistency and synchronization problems since data repeated for devices
- Management rules need to respond to changes in network conditions

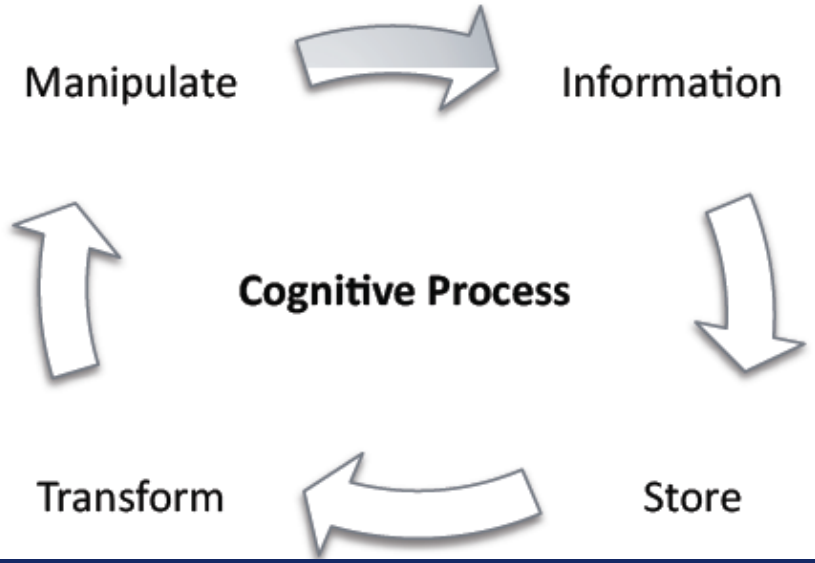
Solutions

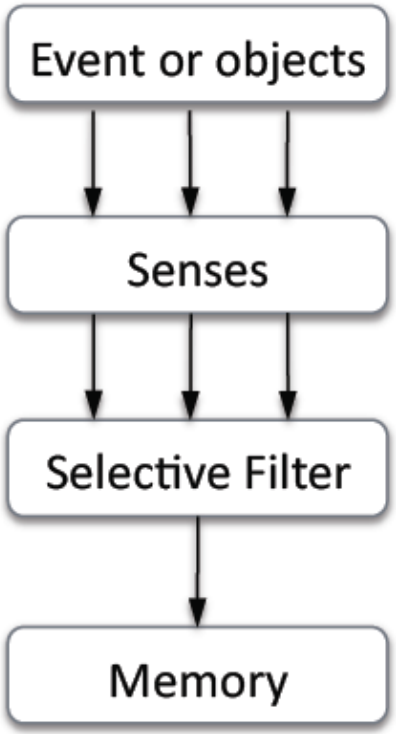
- Technology Policy ⇒ Service Policy
- Protocol Based Management Tables ⇒ Common Information Model
- Configuration ⇒ Policy Management
- Provisioned ⇒ Dynamic ⇒ Reactive Policy



Intro to cognitive (self-) management







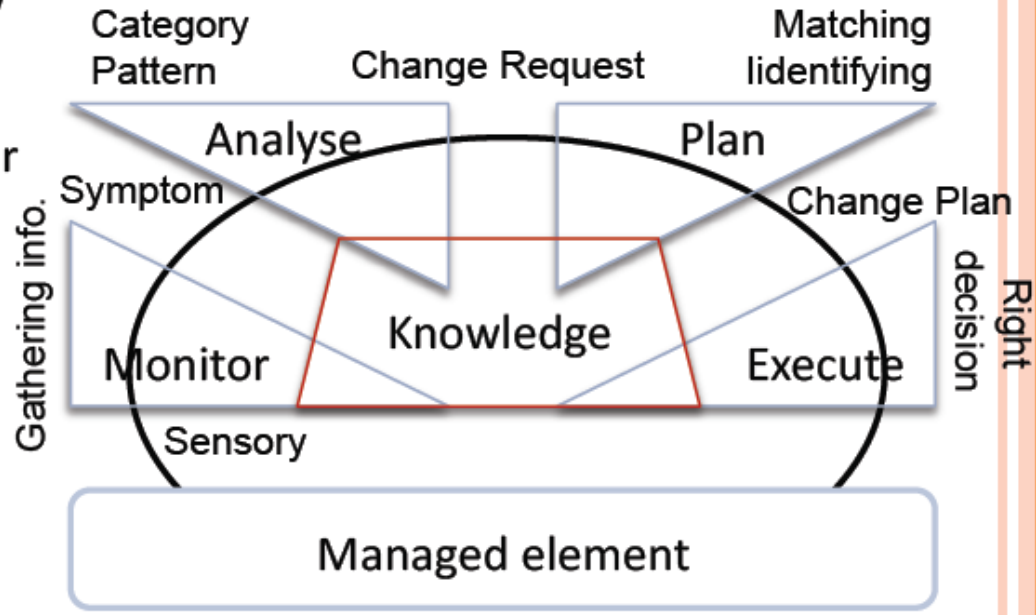
In 1958, British psychologist Donald Broadbent

Autonomic concepts

Self-Management

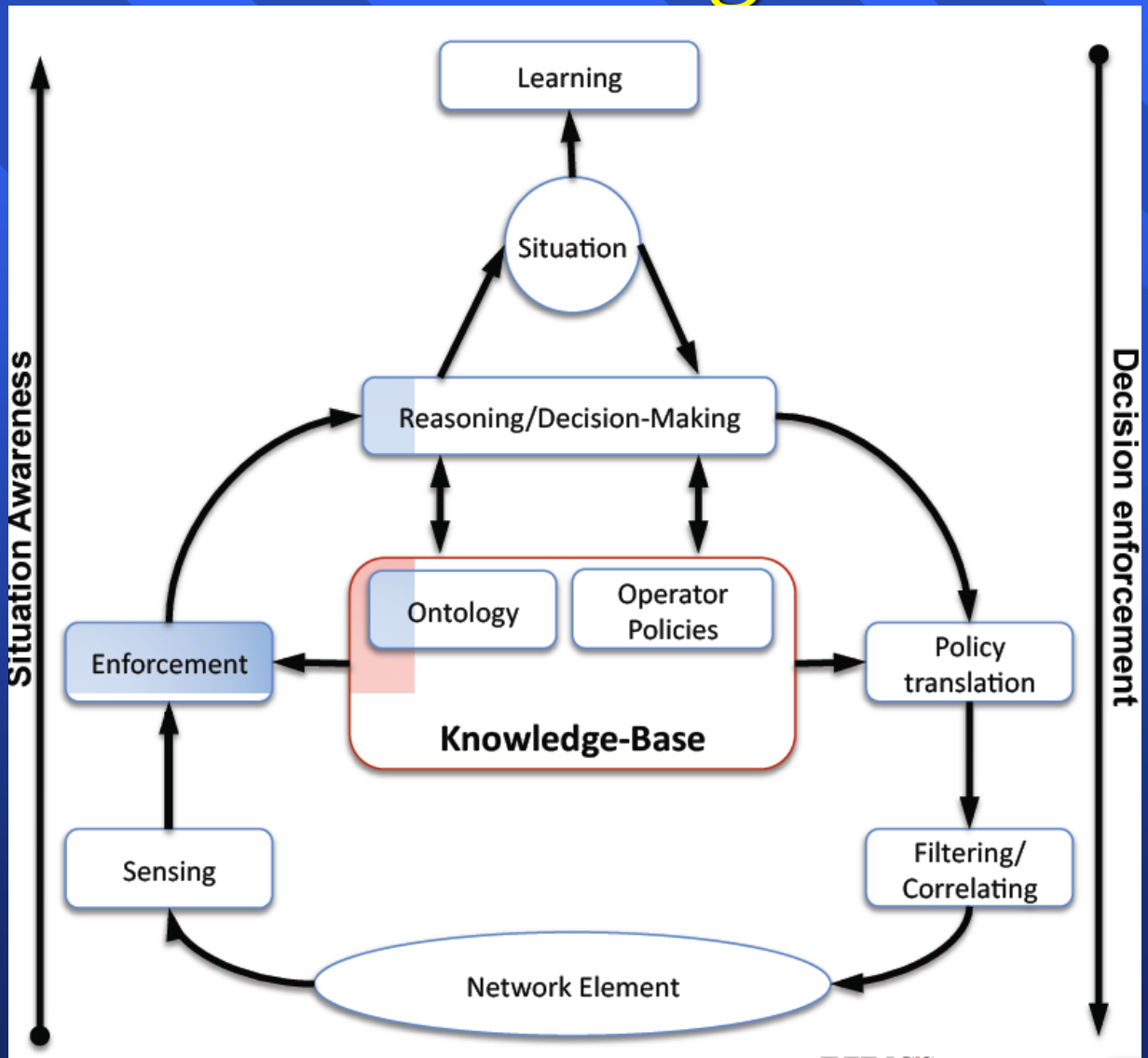
- Self-configuration
  - Adjusts automatically and seamlessly
- Self-optimisation
  - Seek opportunities to improve their own performance and efficiency
- Self-healing
  - Detects, diagnoses, and repairs localized software and hardware problems
- Self- protection
  - Defends against malicious attacks or cascading failures

Autonomic computing life cycle



In 2003, Autonomic Computing [IBM]

Situation awareness and decision making



Knowledge fusion

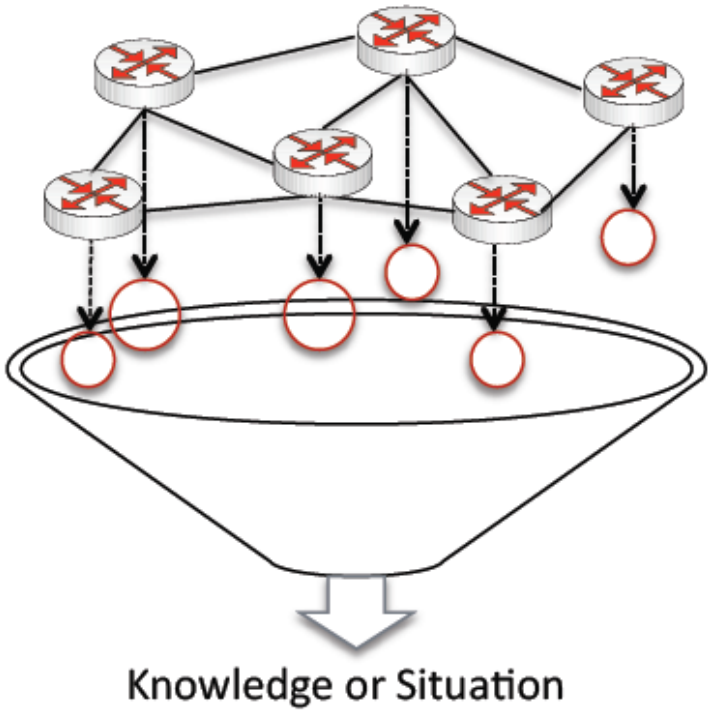
- Combine -> Transform -> Manipulate

Network monitoring tools

- NETCONF: Network Configuration Protocol
- Simple Network Management Protocol (SNMP)

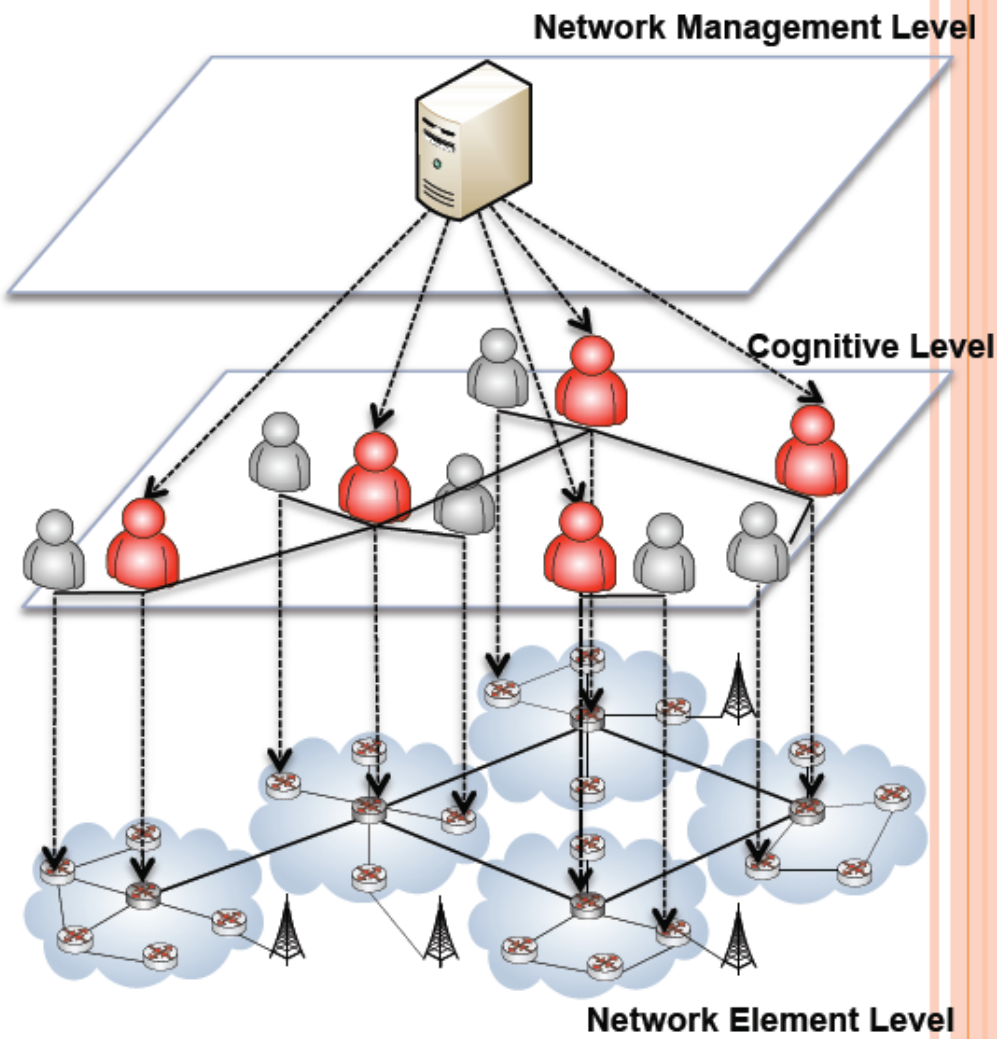
Data Fusion techniques

- Mathematical Methods
  - Dempster–Shafer's Evidential Theory
  - etc...
- Logic Programming
  - Inductive logic programming
  - Constraint logic programming
  - Abductive logic programming
  - etc...



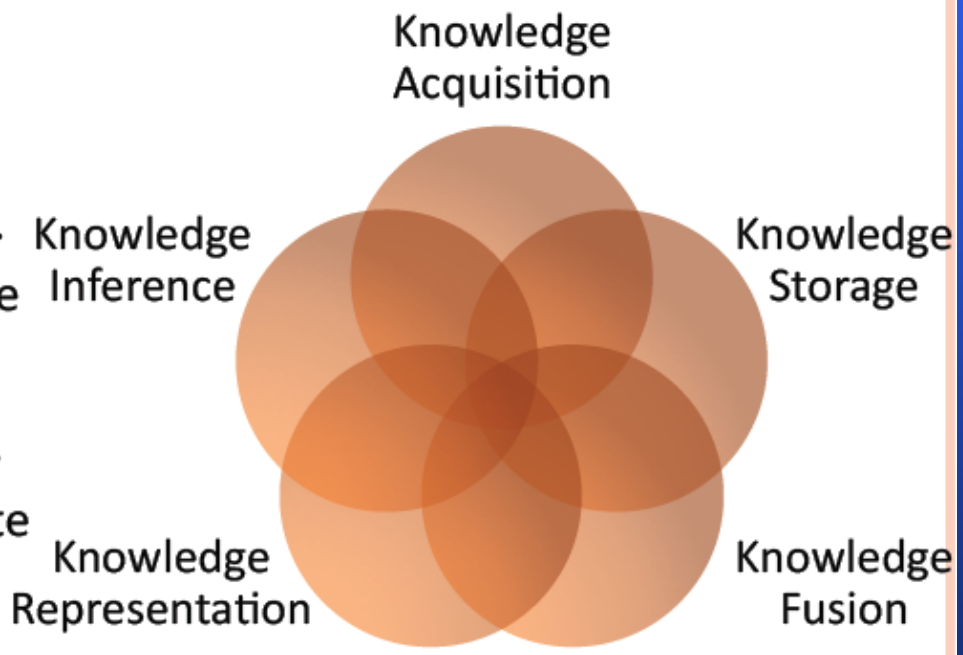
Cognitive network management

- Governing all intelligences in the network
- Tackling all the complexity tasks, performing tasks, monitoring events and making decisions on behalf of the network administrator
- Monitoring on all individual intelligence capabilities, interconnect all individual intelligence functionalities and manage all individual operation invocations in the system etc.



Cognitive Network Knowledge Tools

- Knowledge Acquisition
  - Collecting the right information, at the right time and the right location
- Knowledge Storage
  - Remembering data -> information -> object -> event then retrieve or reuse
- Knowledge Fusion
  - Collecting information -> category -> combine and transform -> manipulate
- Knowledge Representation
  - Organising information
- Knowledge Inference
  - Deriving a logical consequence conclusion



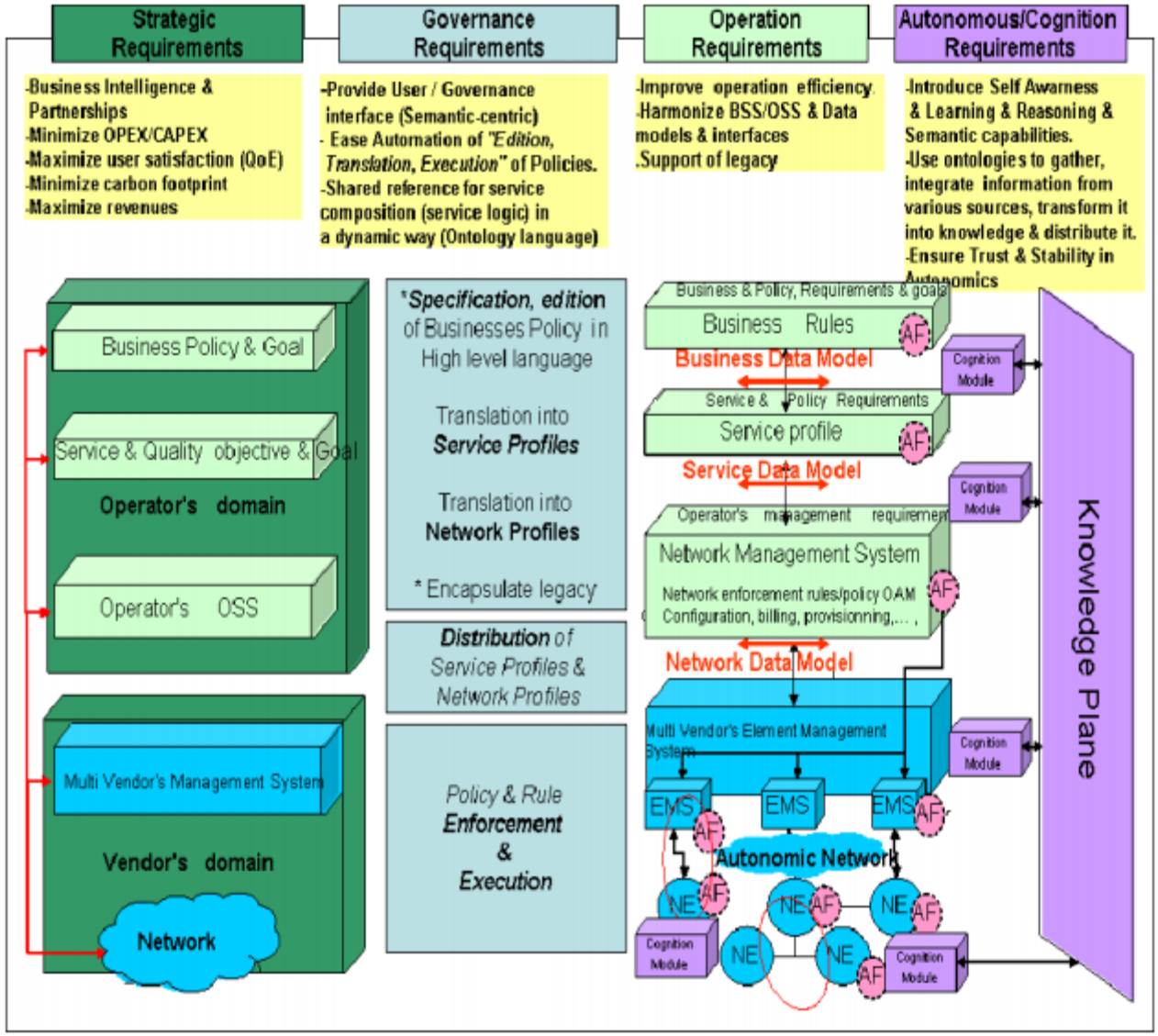


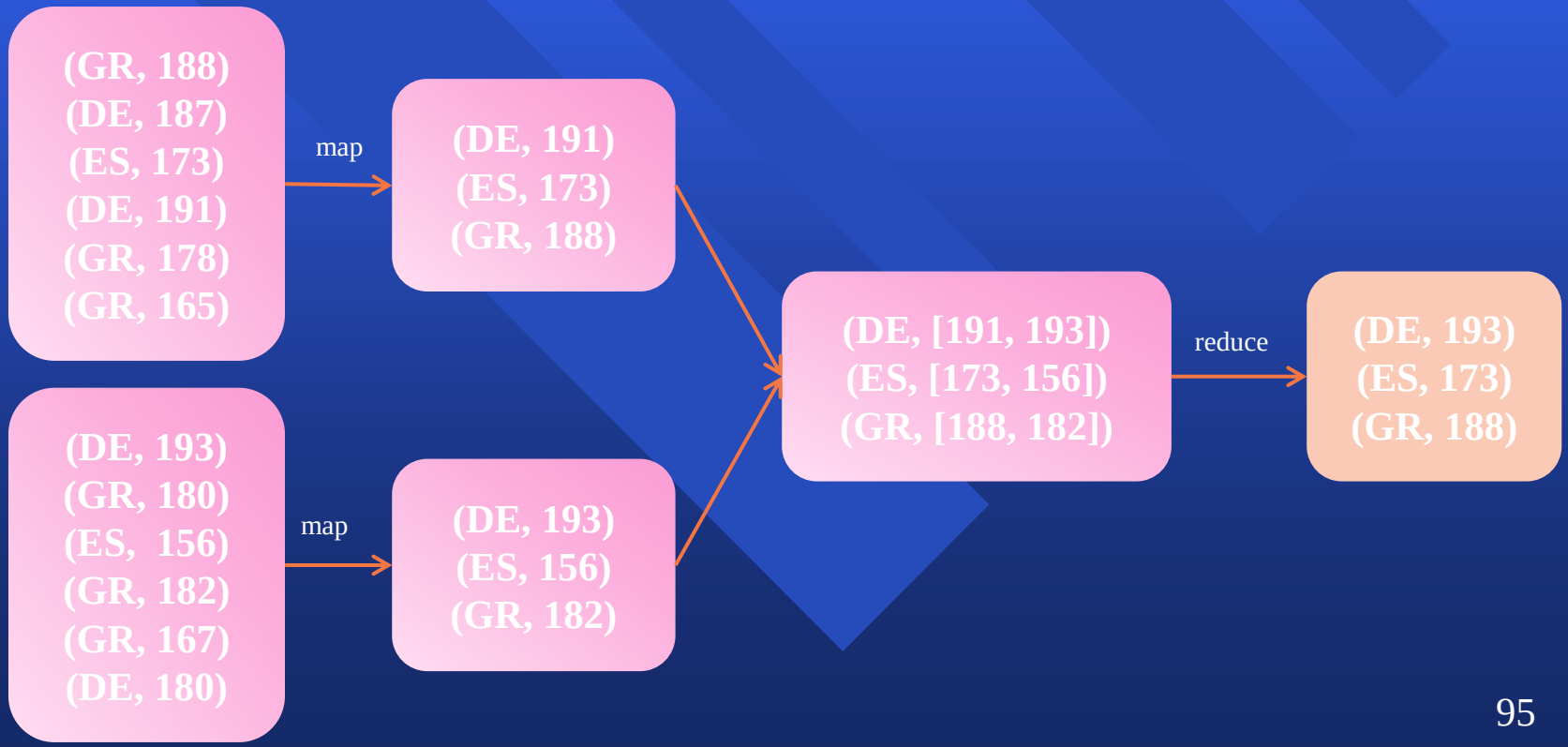
Figure 2: Requirement framework for a Policy- based management of an Autonomics Network

Big Data : Why Distributed Dimensionality Reduction?

- Can handle very large datasets – processing billions of records cannot take place on a single device.
- If data are dispersed in a number of devices, it is resource consuming to transmit all information to one single node.

MapReduce Programming Paradigm

- Distributed data processing model
- Two phases: map & reduce
- Both phases have key – value pairs as input and output



Apache Hadoop

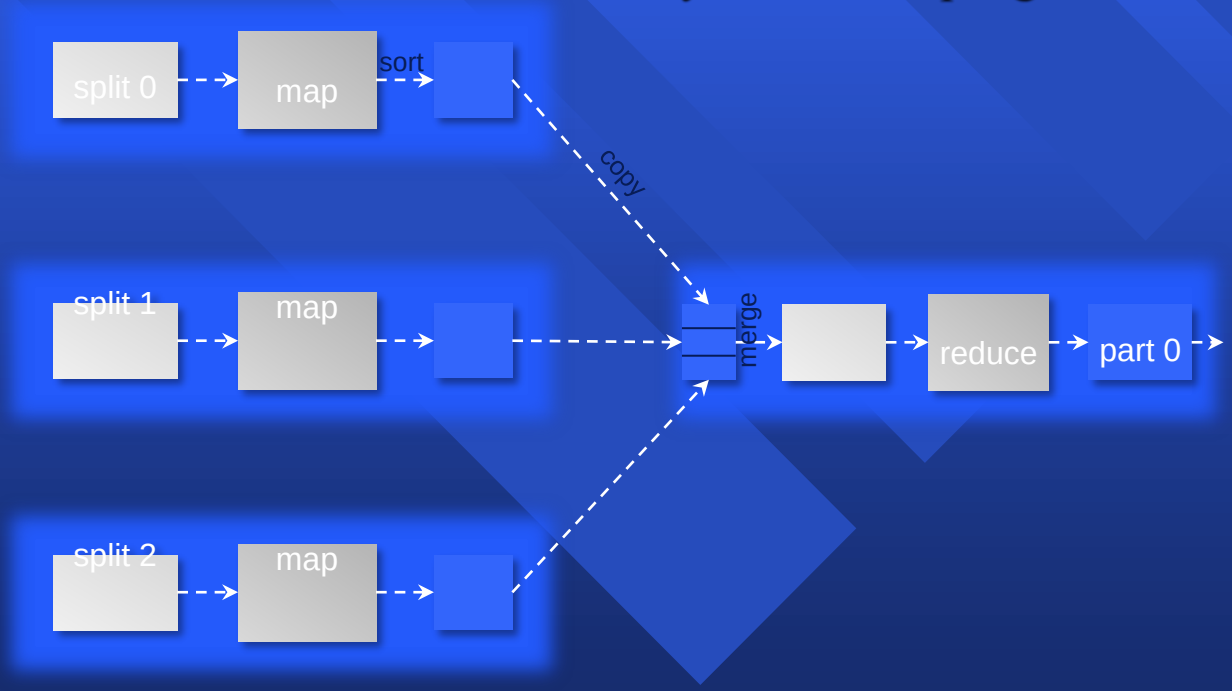
- Open source framework from Apache
- Designed for distributed processing on large scale datasets
- Written in Java, but supports other languages
- Two main subprojects:
  - HDFS → Hadoop Distributed File System
  - MapReduce framework → the most popular implementation of the MapReduce programming paradigm
- Other subprojects:
  - Pig → Distributed environment for data processing
  - HBase → Distributed database (column oriented)
  - ZooKeeper → Contains utilities for distributed processes

Apache Hadoop

- Runs on commodity hardware
- Designed to handle very large files (Gigabytes, Terabytes)
- Block size 64 Mb (default)
- Optimized for fast access to the whole dataset, not the first row
- Not a good choice for many small files
- Does not support simultaneous writers in a file, nor modifications in a random spot of a file

Apache Hadoop – MapReduce

- As many map tasks as the number of blocks of a file (input splits)
- After map phase, the mappers output is sorted and grouped by key
- The number of reducers can be defined by the user – programmer



The Managed Object

- A network may be managed by representing network resources as managed objects. Each MO is a data variable representing one aspect of the managed resource e.g. on/off status, number of packets sent, etc.
- A collection of MOs is called the MIB (Management Information Base), that is, a collection of access points at the agent(s) for the network management system (NMS).

The Managed Object

- Monitoring equates to retrieving values from MIB objects in agents.
- Controlling equates to setting values within the MIB objects in agents. MOs are standardized across systems.
- The Structure of Management Information (SMI) defines syntax (format) and semantics (meaning) of management information stored in the Management Information Base (MIB). Abstract Syntax One (ASN.1) is a formal language standardized by ITU-T (X.208 and X.680) and ISO 8824 that clarifies how data are arranged , what meaning they have and the expected data type.

Mininet: An instant Virtual Network on your Laptop

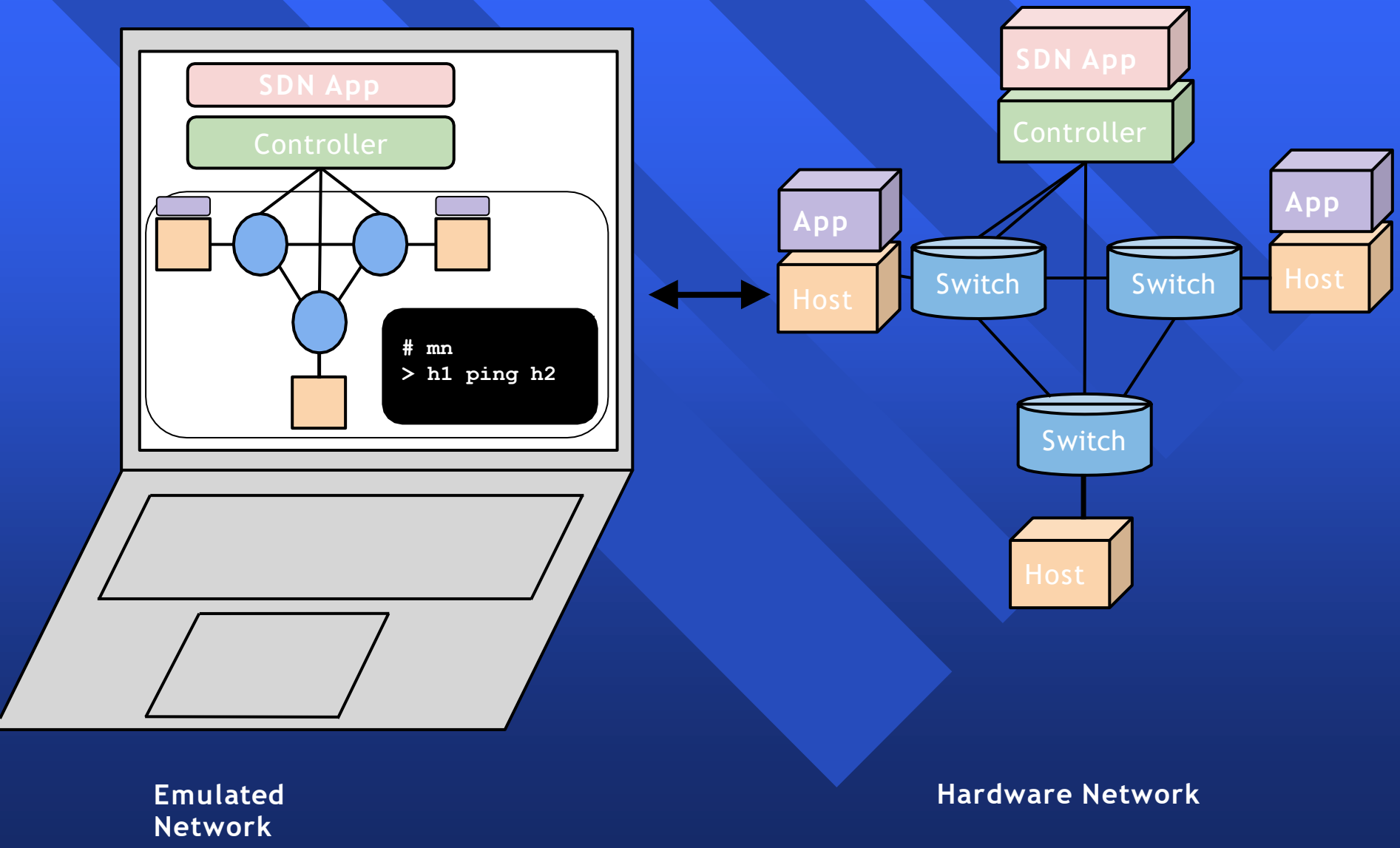
SDN

What is Network Emulation?

In this talk, ***emulation*** (running on an **emulator**) means running *unmodified* code *interactively* on *virtual hardware* on a *regular PC*, providing convenience and realism at low cost – with some limitations (e.g. speed, detail.)

This is in contrast to running on a **hardware testbed** (fast, accurate, expensive/shared) or a **simulator** (cheap, detailed, but perhaps slow and requiring code modifications.)

Apps move seamlessly to/from hardware



Emulated Network

Hardware Network

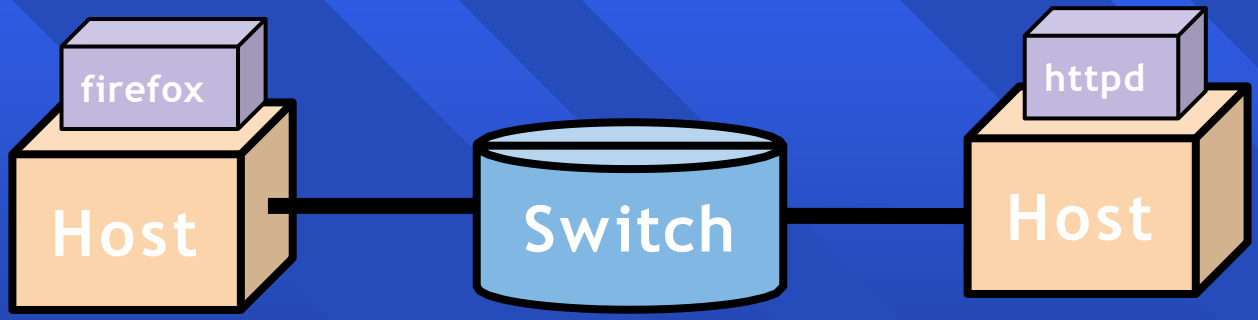
Creating a Network Emulator

Need to model **hosts**, **switches**, **links**, and (possibly) network **controllers** (for SDN/OpenFlow.)

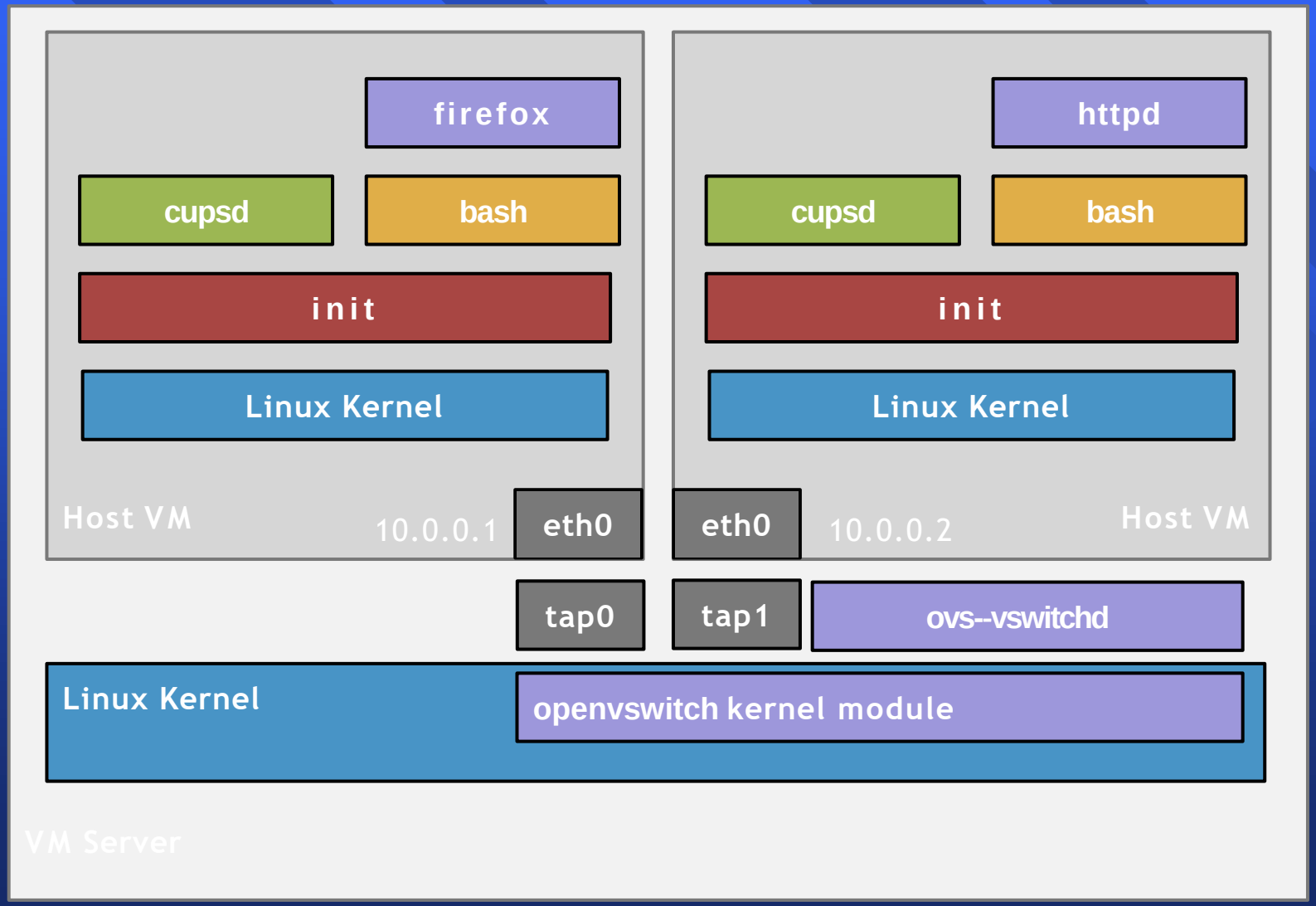
Scalability challenge: we would like to model networks of interesting size with practical performance.

How to do it? Virtualization!

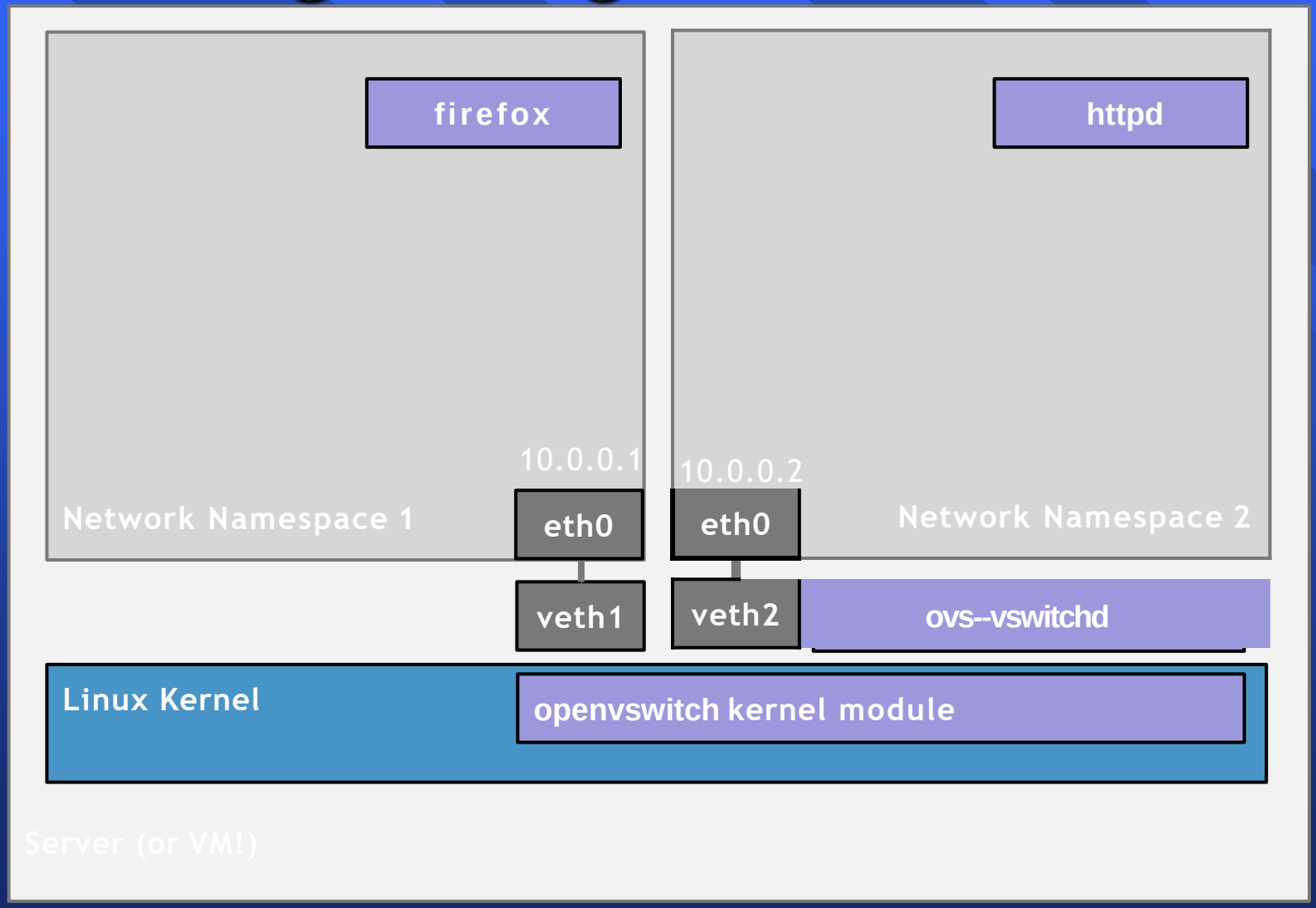
To start with, a Very Simple Network



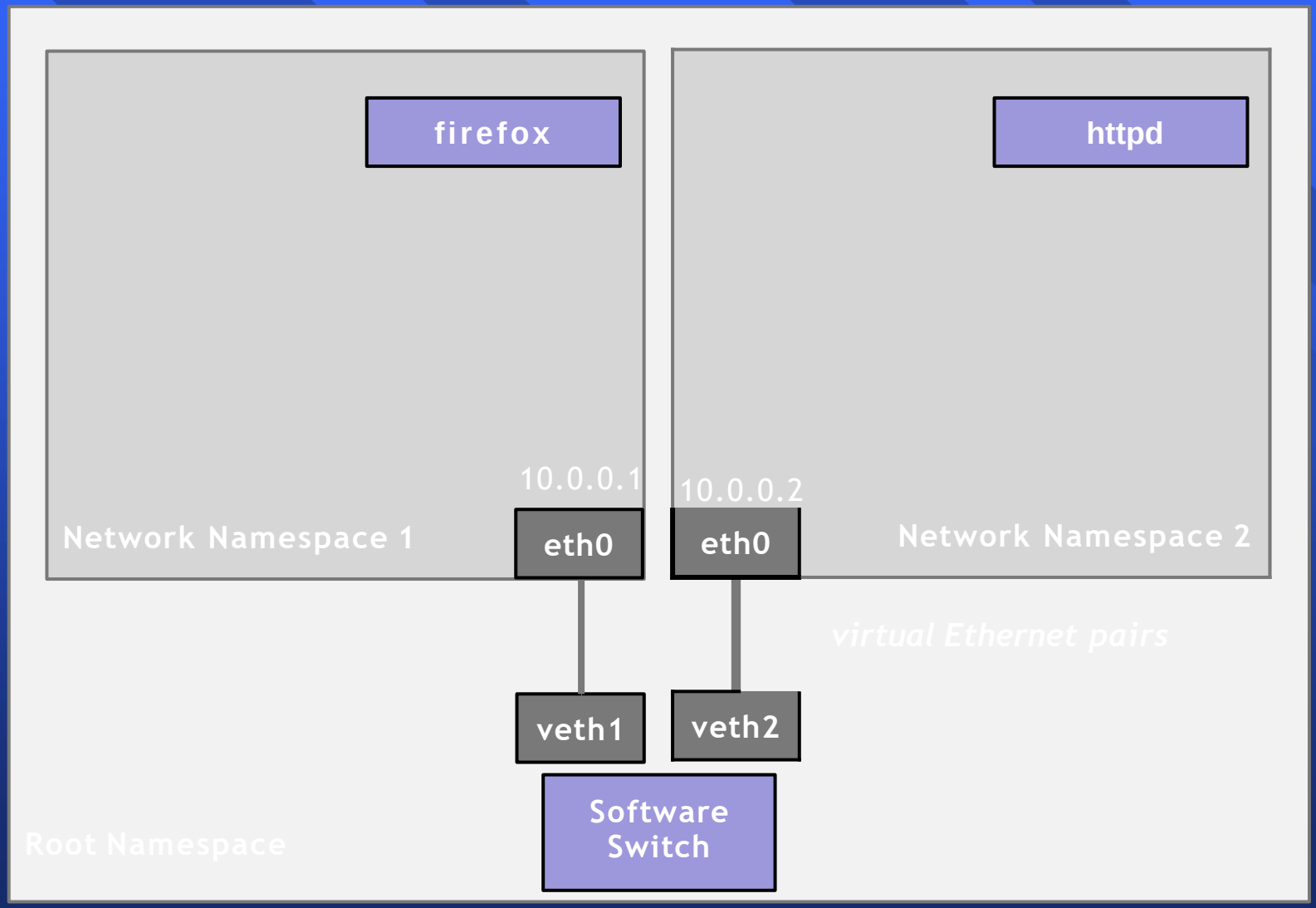
Very Simple Network using Full System Virtualization



Very Simple Network using Lightweight Virtualization



Network Namespaces and Virtual Ethernet Pairs

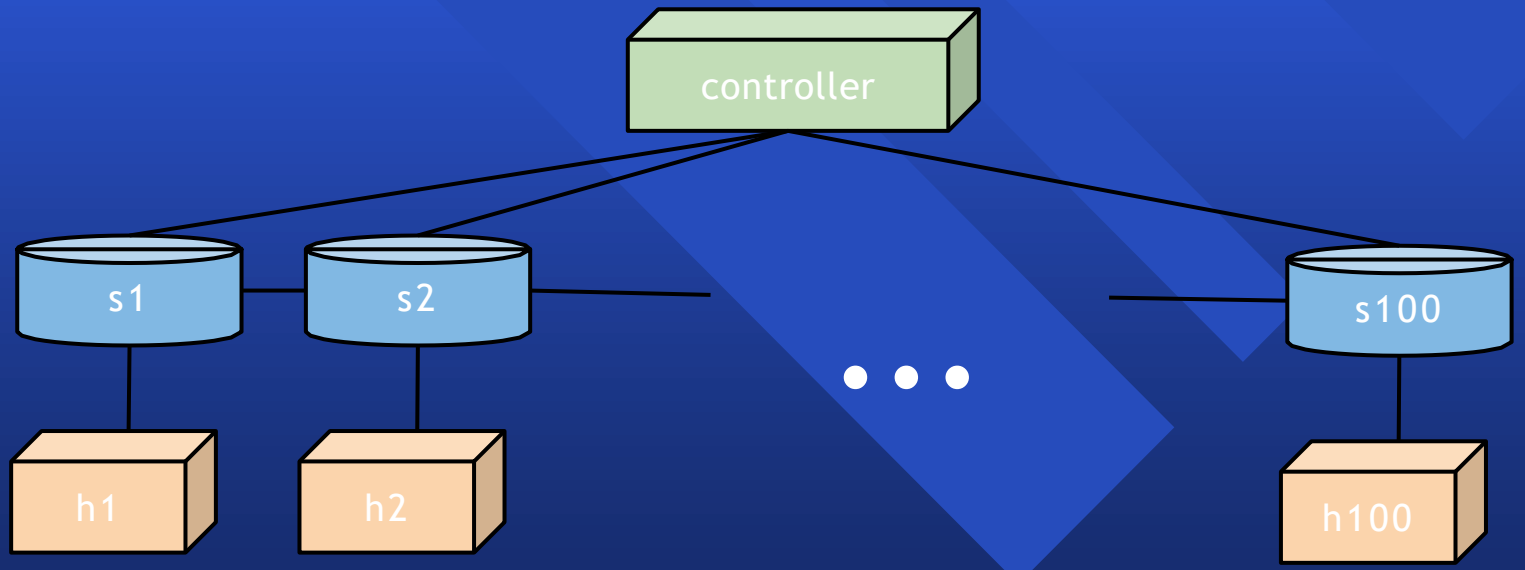


Lightweight Virtualization in Linux

Network component or property	Modeling mechanism	Configuration command(s)
Hosts	Processes in network namespaces	ip netns
Links	Virtual Ethernet pairs	ip link
Switches	Software switches (OVS)	ovs--vsctl
Controllers	Processes	controller

Demo

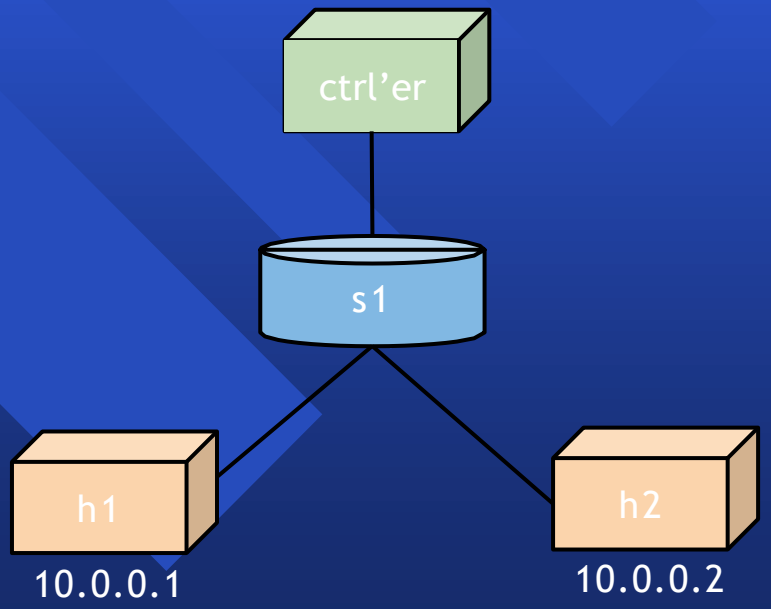
```
# mn --topo linear,100 --switch user
  ----controller ref
```



Demo: basic network setup in Linux

```
sudo bash
# Create host namespaces
ip netns add h1
ip netns add h2
# Create switch
ovs--vsctl add–br s1#
Create links
ip link add h1–eth0type veth peer name s1–eth1
ip link add h2–eth0type veth peer name s1–eth2 ip link
show
# Move host ports into namespaces
ip link set h1–eth0netns h1
ip link set h2–eth0 netns h2
ip netns exec h1 ip link show
ip netns exec h2 ip link show #
Connect switch ports to OVS ovs-
-vsctl add–port s1 s1-eth1 ovs--vsctl
add–port s1 s1-eth2 ovs--vsctl show
# Set up OpenFlow controller
ovs--vsctl set--controller s1 tcp:127.0.0.1 ovs-
-controller ptcp: &
ovs--vsctl show
```

```
# Configure network
ip netns exec h1 ifconfig h1–eth010.1 ip
netns exec h1 ifconfig lo up
ip netns exec h2 ifconfig h2–eth010.2 ip
netns exec h1 ifconfig lo up ifconfig s1-
-eth1 up
ifconfig s1–eth2 up#
Test network
ip netns exec h1 ping -c110.2
```



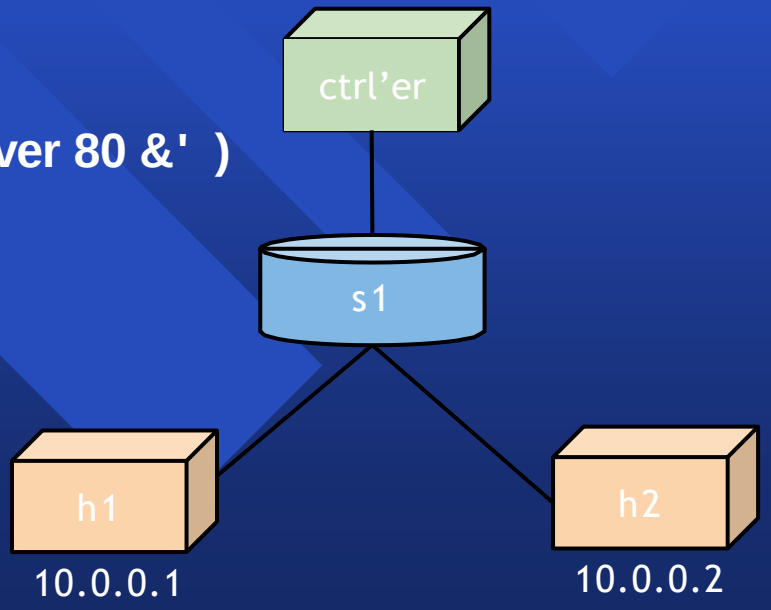
Wouldn't it be great if...

- We had a simple command-line tool and/or API that did this for us automatically?
- It allowed us to easily create topologies of varying size, up to hundreds of nodes, and run tests on them?
- It was already included in Ubuntu?

Basic network setup in Mininet (API)

```
net = Mininet()
h1 = net.addHost( 'h1' )
h2 = net.addHost( 'h2' )   s1
= net.addSwitch( 's1' )
c0 = net.addController( 'c0' )
net.addLink( h1, s1 )
net.addLink( h2, s1 )
net.start()
h2.cmd( 'python -m SimpleHTTPServer 80 &' )
sleep( 2 )
print h1.cmd('curl', h2.IP() )
CLI( net )
h2.cmd('kill %python')
net.stop()
```

```
# net is a Mininet() object
# h1 is a Host() object
# h2 is a Host()
# s1 is a Switch() object #
c0 is a Controller()
# creates a Link() object
```



mn command and Mininet CLI demo

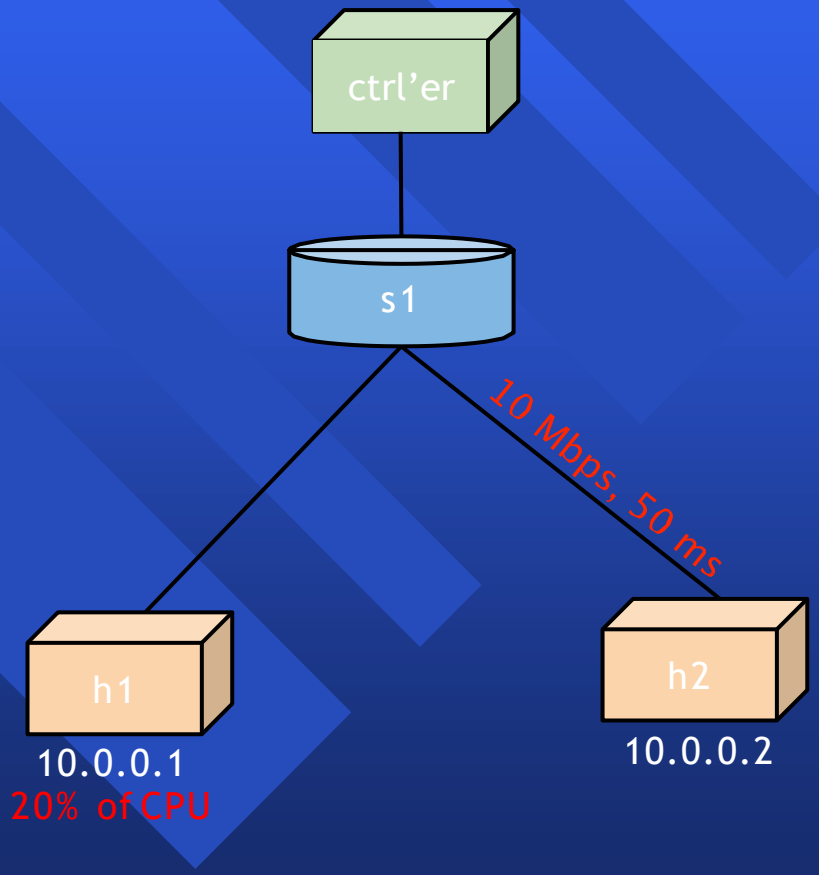
```
# mn --test pingall

# mn --topo tree,depth=3,fanout=3 --
 link=tc,bw=10                    -

mininet> xterm h1 h2
h1# wireshark &
h2# python -m SimpleHTTPServer 80
h1# firefox&
```

Performance modeling with Linux

Network component or property	Modeling mechanism	Configuration command(s)
Hosts	Processes in network namespaces	ip netns
Links	Virtual Ethernet pairs	ip link
Switches	Software switches (OVS)	ovs--vsctl
Controllers	Processes	controller
Link performance	Traffic Control (and netem subsystem)	tc
CPU performance	CPU Control Groups (CFS bandwidth limits)	cg{create,set,delete, classify}

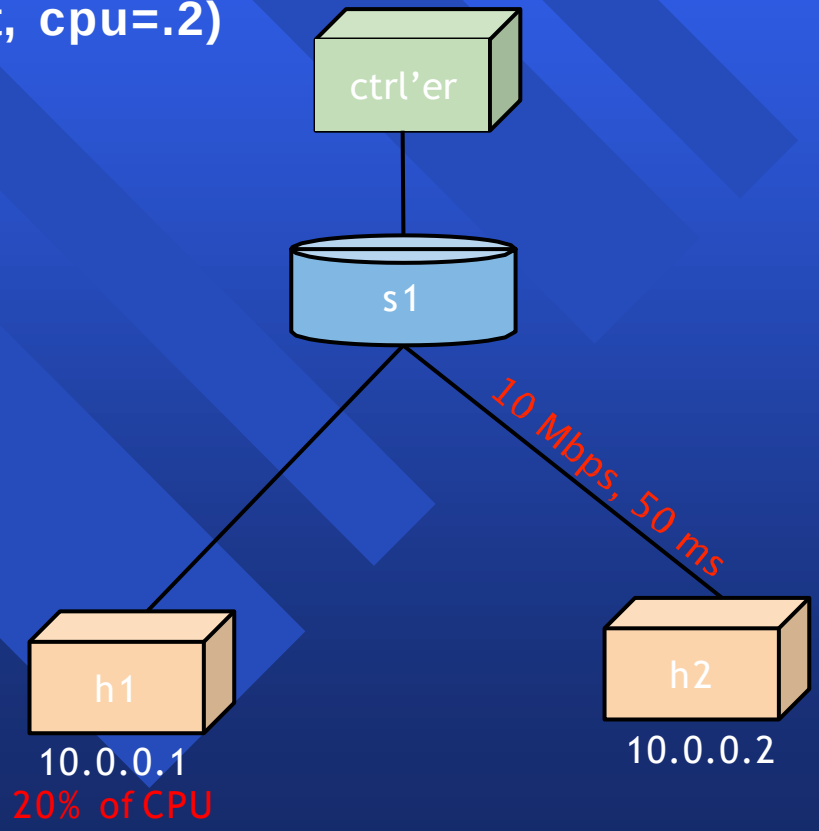
Demo: performance setup in Linux

```
# Limit link bandwidth and add delay
tc qdisc add dev s1-eth2 root handle 5: tbf rate
   10Mbit burst 5k latency 12ms
tc qdisc add dev s1-eth2 parent 5:1 handle 10: netem
   delay 50ms
ip netns exec h1 ping -c4 10.2
ip netns exec h2 iperf -s >&/dev/null &ip
netns exec h1 iperf -t5 -c10.2
# Limit CPU bandwidth
cgcreate -g cpu:/h1
cgset -r cpu.cfs_period_us=100000 /h1
cgset -r cpu.cfs_quota_us=20000 /h1
ip netns exec h1 bash -c "while true; do a=1;done" &
cgclassify -g cpu:/h1 $!
```



Performance setup in Mininet

```
# Limit link bandwidth and add delay
net.addLink(h2, s1, cls=TCLink,
                bw=10, delay='50ms')
# Limit CPU bandwidth
net.addHost('h1', cls=CPULimitedHost, cpu=.2)
```



Accuracy = Matching hardware

Experiments on emulator should match results on hardware.

How to test? Micro/macrobenchmarks? http://hci.stanford.edu/cstr/reports/2012-02.pdf

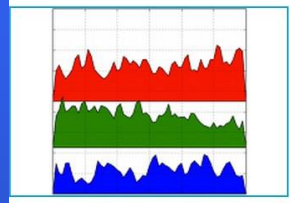
Better (and bigger) idea: Grad students! reproducingnetworkresearch.wordpress.com

REPRODUCING NETWORK RESEARCH

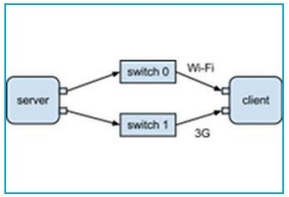
network systems experiments made accessible, runnable, and reproducible

projects / about / contribute

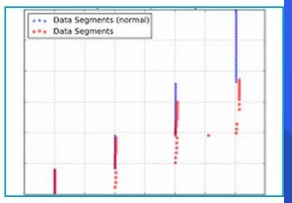
Posts by CS244 Spring 2012 Students



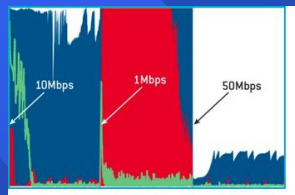
Exploring Outcast



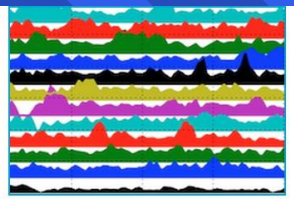
Multipath TCP over WiFi and 3G links



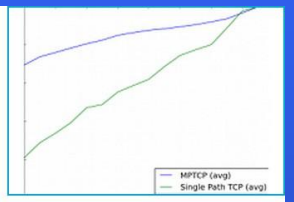
TCP Daytona: Congestion Control with a Misbehaving Receiver



Solving Bufferbloat – The CoDel Way

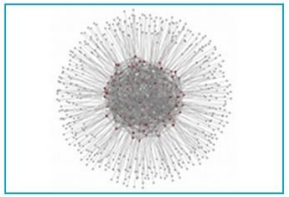


Life's not fair, neither is TCP (… under the following conditions)

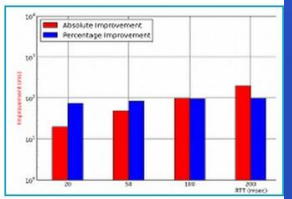


Fairness of Jellyfish vs. Fat-Tree

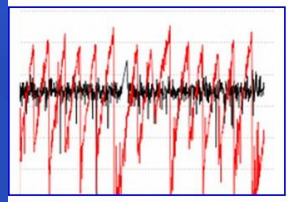
DCell: A Scalable and Fault-Tolerant Network Structure for Data Centers



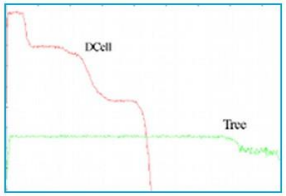
Jellyfish vs. Fat Tree



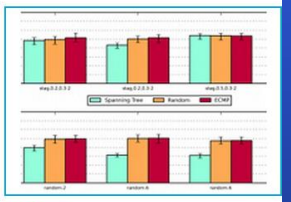
Choosing the Default Initial Congestion Window



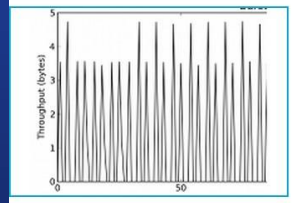
DCTCP and Queues



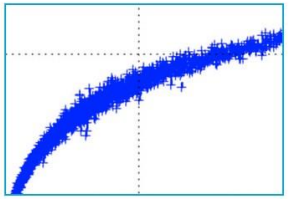
DCell : A Scalable and Fault-Tolerant Network Structure for Data Centers



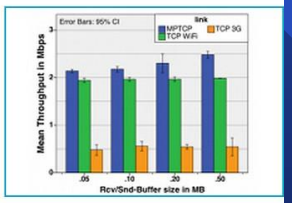
Hedera



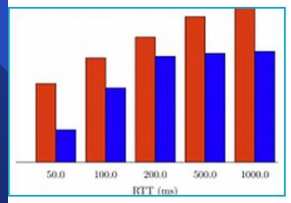
Seeing RED



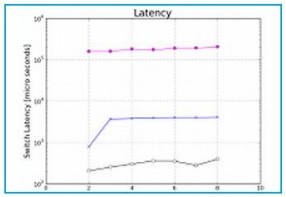
Why Flow-Completion Time is the Right Metric for Congestion Control



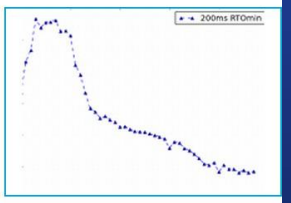
MPTCP Wireless Performance



Increasing TCP's Initial Congestion Window



HULL: High Bandwidth, Ultra Low Latency



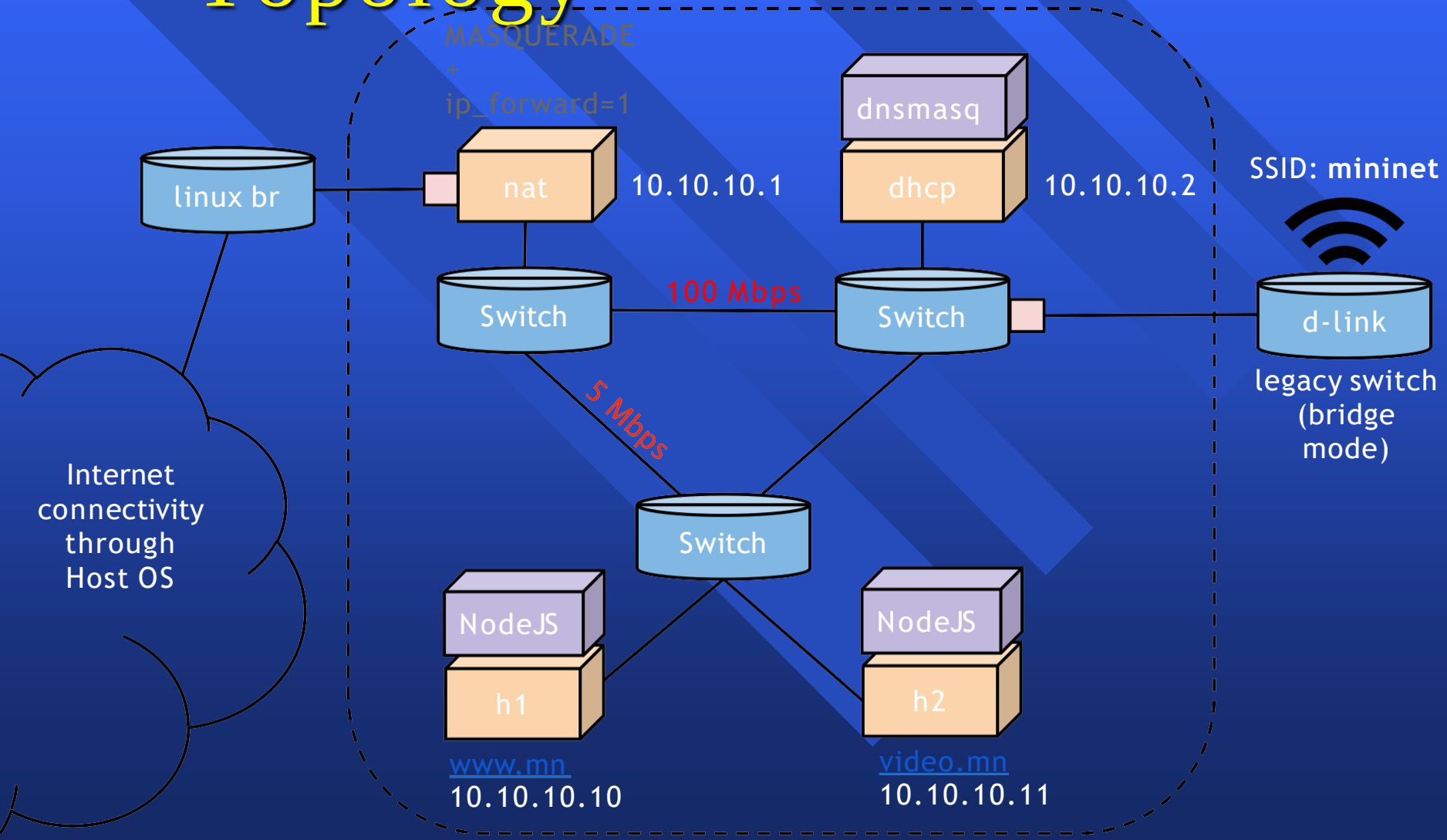
TCP Incast Collapse

Demo: Connect to Mininet with your phone or laptop!

Interactive demo here in Gates 104:

1) **connect to "mininet" wi-fi network**
2) check your IP address if you like
3) go to www.mn
4) go to google.com or any other site

Demo Topology



```
class DemoTopo( Topo ):
  def __init__(self, inetIntf, wlanIntf, **opts):
    Topo.__init__(self, **opts)
    s1 = self.addSwitch( 's1' )
    s2 = self.addSwitch( 's2' )
    s3 = self.addSwitch( 's3' )
    # connect switches in a triangle
    self.addLink( s1, s2, cls=TCLink, bw=5, delay='1ms', max_queue_size=20 )
    self.addLink( s2, s3)
    self.addLink( s1, s3, cls=TCLink, bw=100, delay='1ms', max_queue_size=20 )
    # add servers and connect to s2
    h1 = self.addHost( 'h1', ip='10.10.10.10/24', defaultRoute='via 10.10.10.1' )
    h2 = self.addHost( 'h2', ip='10.10.10.11/24', defaultRoute='via 10.10.10.1' )
    self.addLink( s2, h1)
    self.addLink( s2, h2 )
    # add NAT and connect it to s1
    nat = self.addNode( 'nat', ip='10.10.10.1/24', cls=NAT,
                        subnet=10.10.10.0/24, inetIntf=inetIntf, inNamespace=False )
    self.addLink( s1, nat )
    # add DHCP server and connect it to s3
    dhcp = self.addNode( 'dhcp', ip='10.10.10.2/24', cls=DHCPServer, defaultRoute='via 10.10.10.1' )
    self.addLink( s3, dhcp)
    # add the WiFi interface to s3
    self.addLink( s3, s3, cls=HWIntfLink, intfName1=wlanIntf, cls2=NoneIntf )
```

Enjoy Mininet!

mininet.org

docs.mininet.org

Network Emulation

-Why it's awesome

Challenges

- Scalability (demo)
- Performance (demo)

-Ease of use (demo)

Interactive Demo