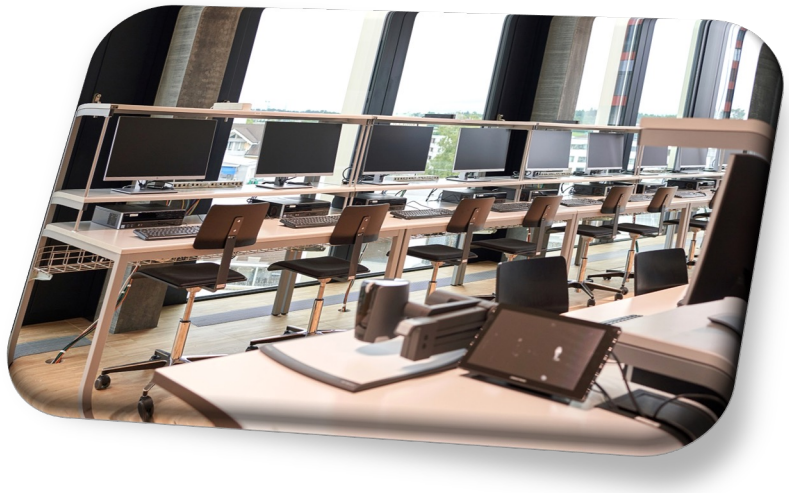
Lucerne University of
Applied Sciences and Arts

HOCHSCHULE
LUZERN

Informatik

Networking III – SW4: Network Virtualization & Automation



FH Zentralschweiz

Ausbildung

Dozent, **Diego Ortiz Yepes**
diego.ortizyepes@hslu.ch

Our Plan for today

Time (ca.)	Topic	Activity Type
5 min	Administrivia	Plenum
70 min	Network Virtualization (ENSA13)	Plenum + Live Quiz
15 min	Pause	
30 min	Network Automation (ENSA14)	Plenum + Live Quiz
20 min	Netacad Review, Q&A	Individual + Plenum

Administrativia

- Nächste Woche (21.03.2024)
  - Software Defined Networking (SDN)
  - Gastvortrag Florian Wamser
  - Online
- Übernächste Woche (28.03.2024)
  - Labor 1 - **Vor Ort** (Raum/Labor 404)
  - Betreuung: Stefan Küng
  - Abgabe in 2er Gruppen via ILIAS bis 31.03.2024 (Testatbedingung)

Our Plan for today

Time (ca.)	Topic	Activity Type
5 min	Administrivia	Plenum
70 min	Network Virtualization (ENSA13)	Plenum + Live Quiz
15 min	Pause	
30 min	Network Automation (ENSA14)	Plenum + Live Quiz
20 min	Netacad Review, Q&A	Individual + Plenum

Live Quiz: Mentimeter

https://www.menti.com/al3a7r6vzu6h

www.menti.com Code: ***1541 0965***

Mentimeter

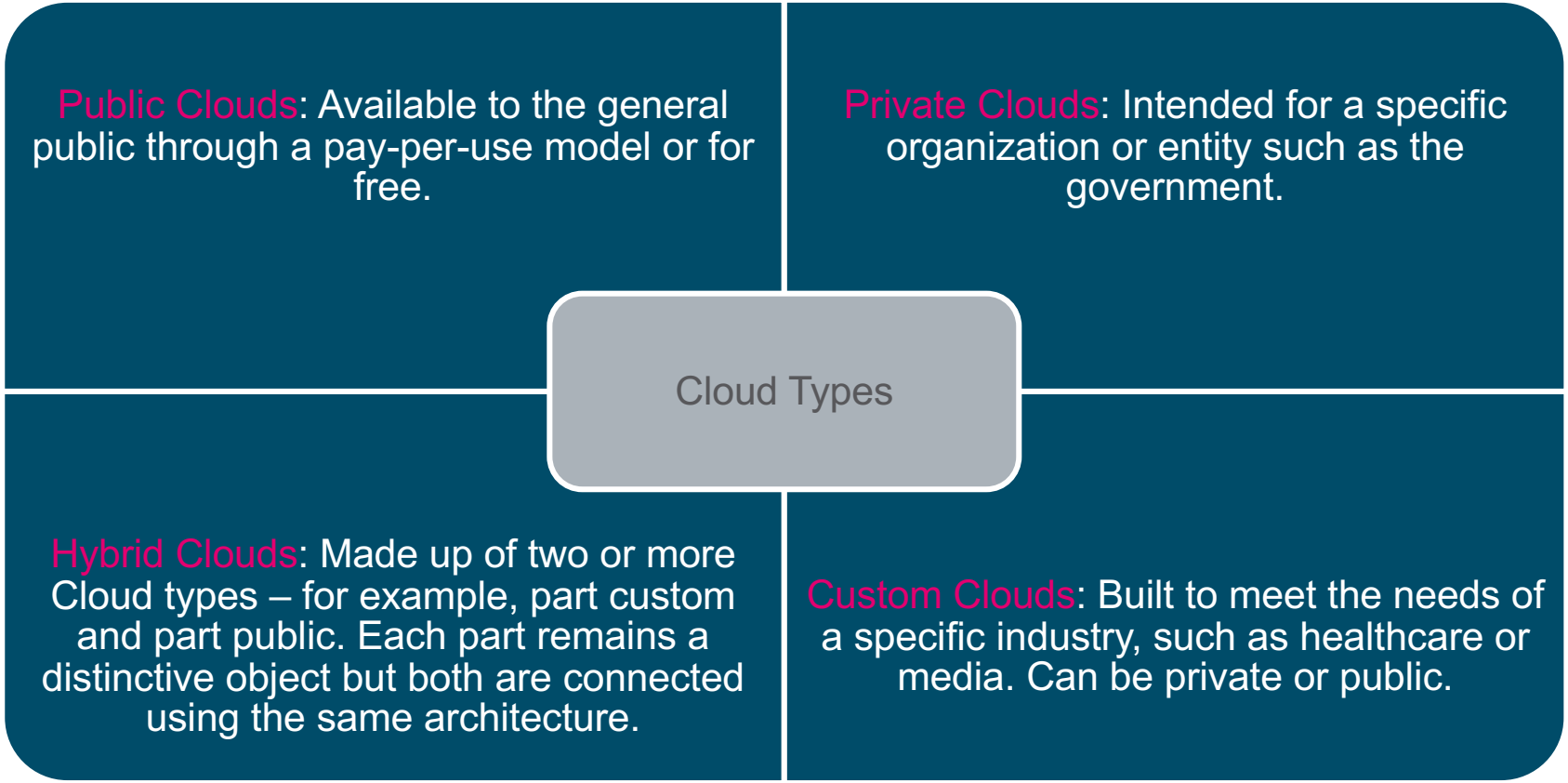
Network Trends

Cloud Computing

- Enables access to organizational data anywhere and at any time
- Streamlines the organization's IT operations: subscribing only to needed services
- Eliminates or reduces the need for onsite IT equipment, maintenance, and management
- Reduces cost for equipment, energy, physical plant requirements, and personnel training needs
- Enables rapid response to increasing data volume or capacity

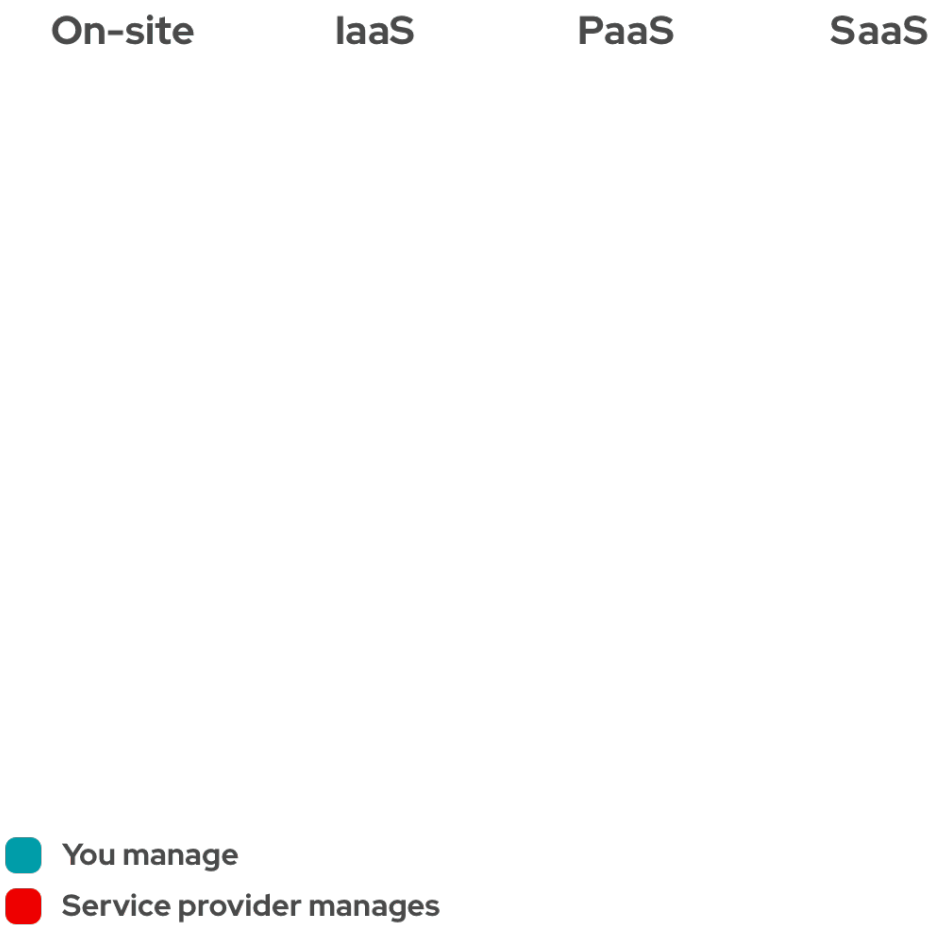
Network Trends

Cloud Computing (Cont.)



Cloud Computing

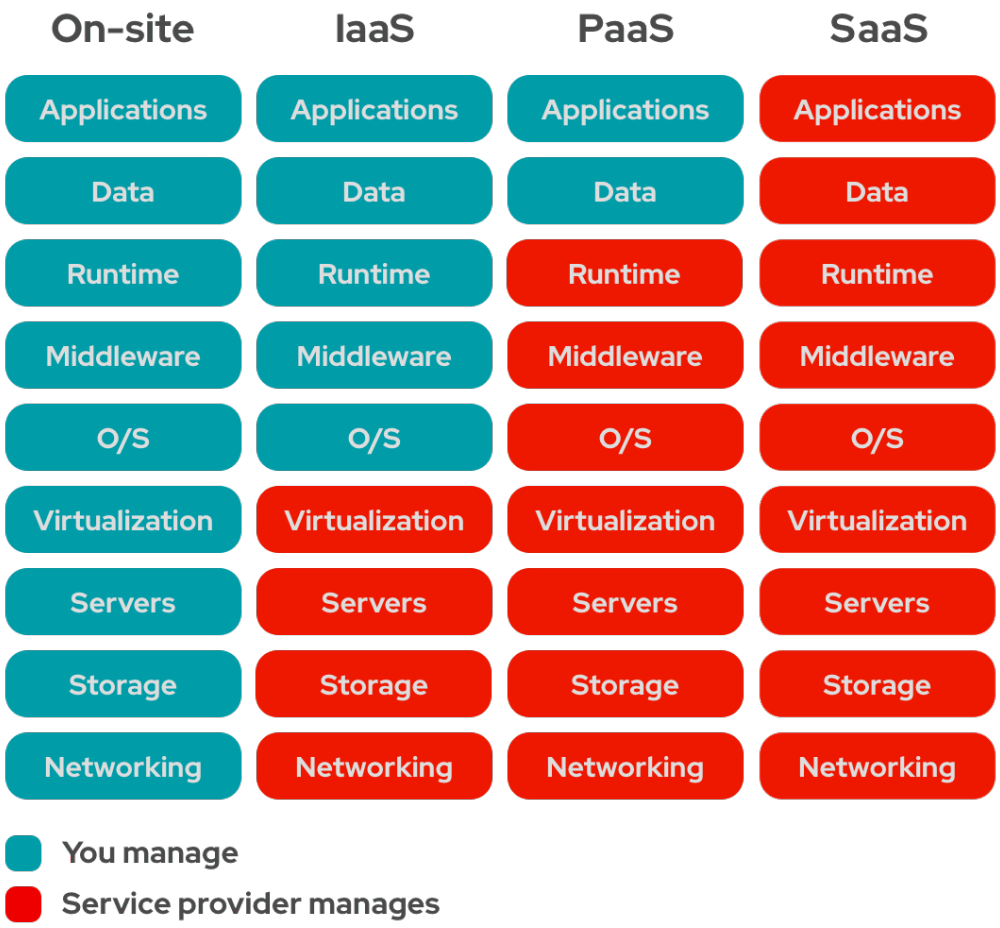
Cloud Services



https://www.redhat.com/de/topics/cloud-computing/iaas-vs-paas-vs-saas

Cloud Computing

Cloud Services



https://www.redhat.com/de/topics/cloud-computing/iaas-vs-paas-vs-saas

Cloud Computing versus Data Center

- Data center: Physical facility, usually very expensive to build and maintain.

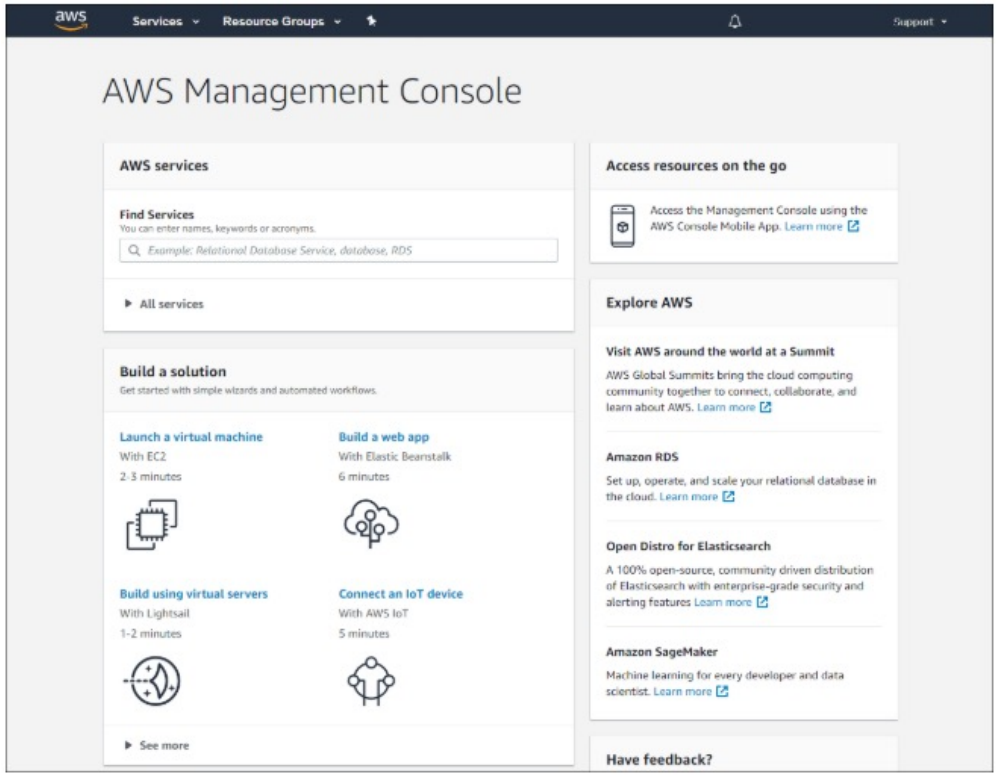
- Cloud computing: computing service/model. Typically:
  - off-premise
  - on-demand
  - shared pool of configurable computing resources
  - Rapidly provisioned and released with minimal management effort

The cloud does not run on the cloud: Cloud service providers use data centers to host their cloud services and cloud-based resources

- s. Wolkenbildung: Die Schweiz als neuer Hotspot für Datenzentren (https://news.hslu.ch/datenzentren-boom/)

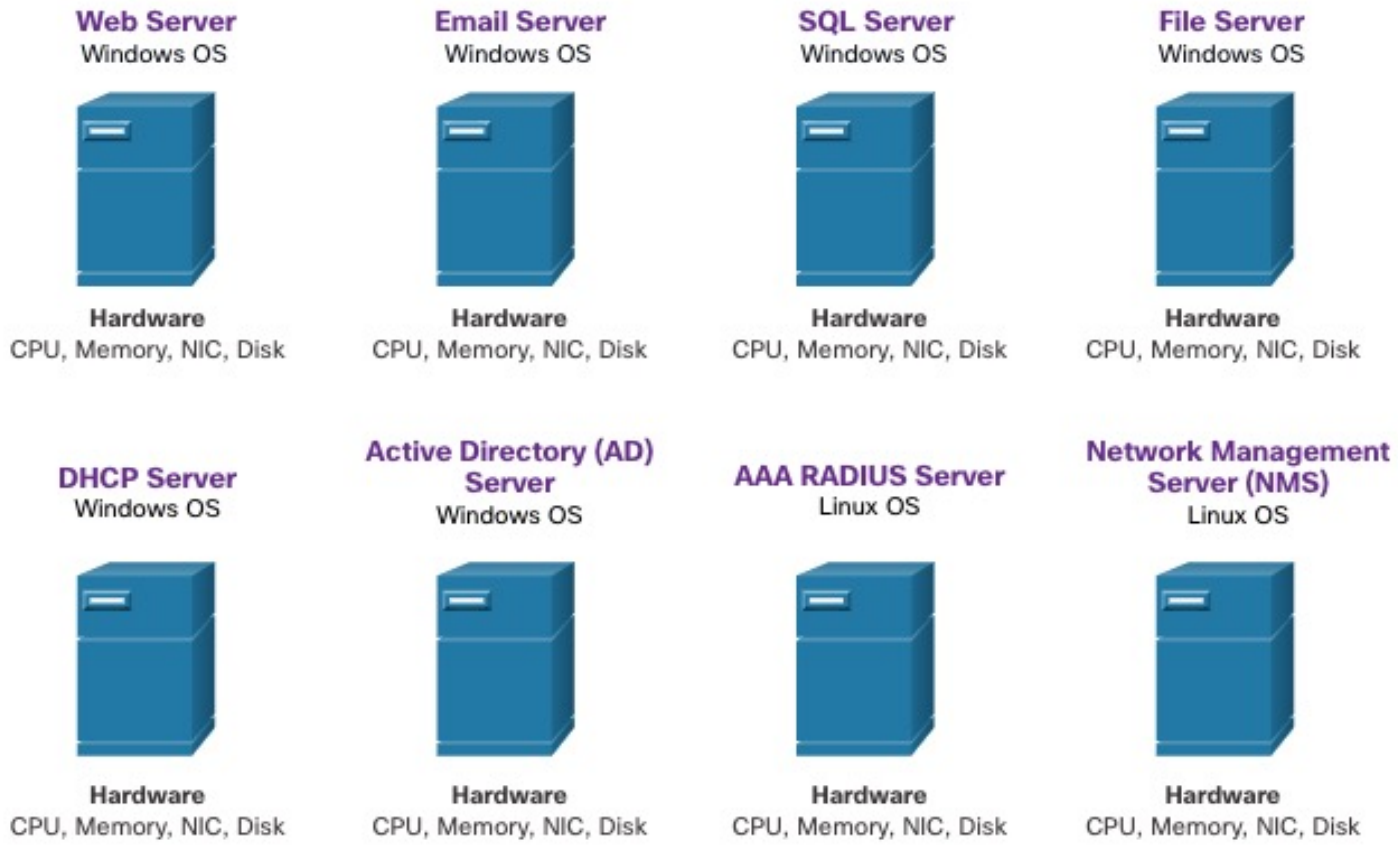
Cloud Computing and Virtualization

- Virtualization:
  - foundation of cloud computing
  - separates the operating system (OS) from the hardware
  - virtualized instances of servers are created on demand



Virtualization

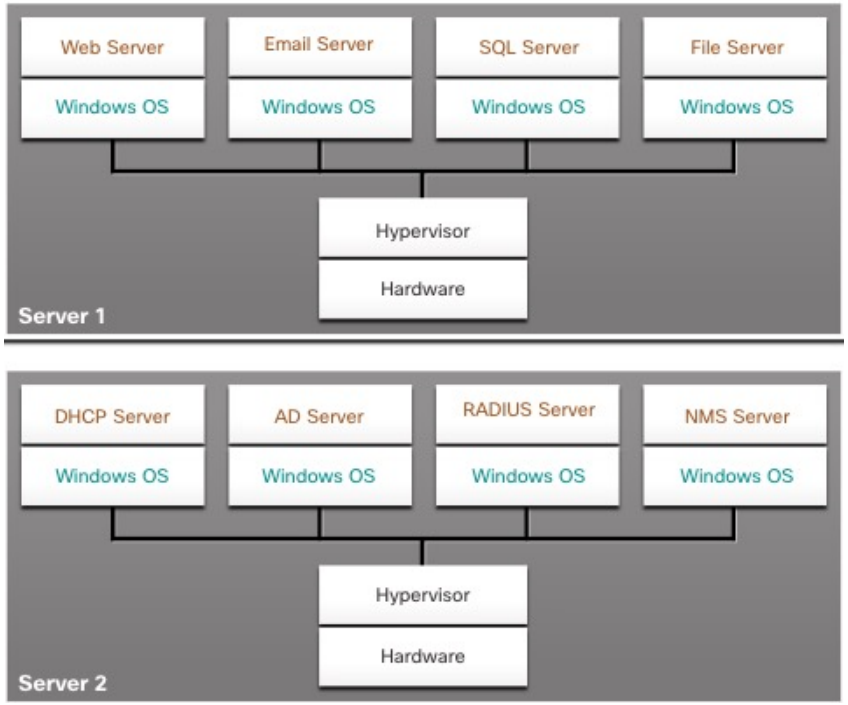
Dedicated Servers



Virtualization

Server Virtualization

- Consolidation
- Efficient resource usage
- Redundancy
- Replicability
- Hypervisor: program, firmware, or hardware that adds an abstraction layer for the VMs to have access to shared resources



Virtualization

Advantages of Virtualization

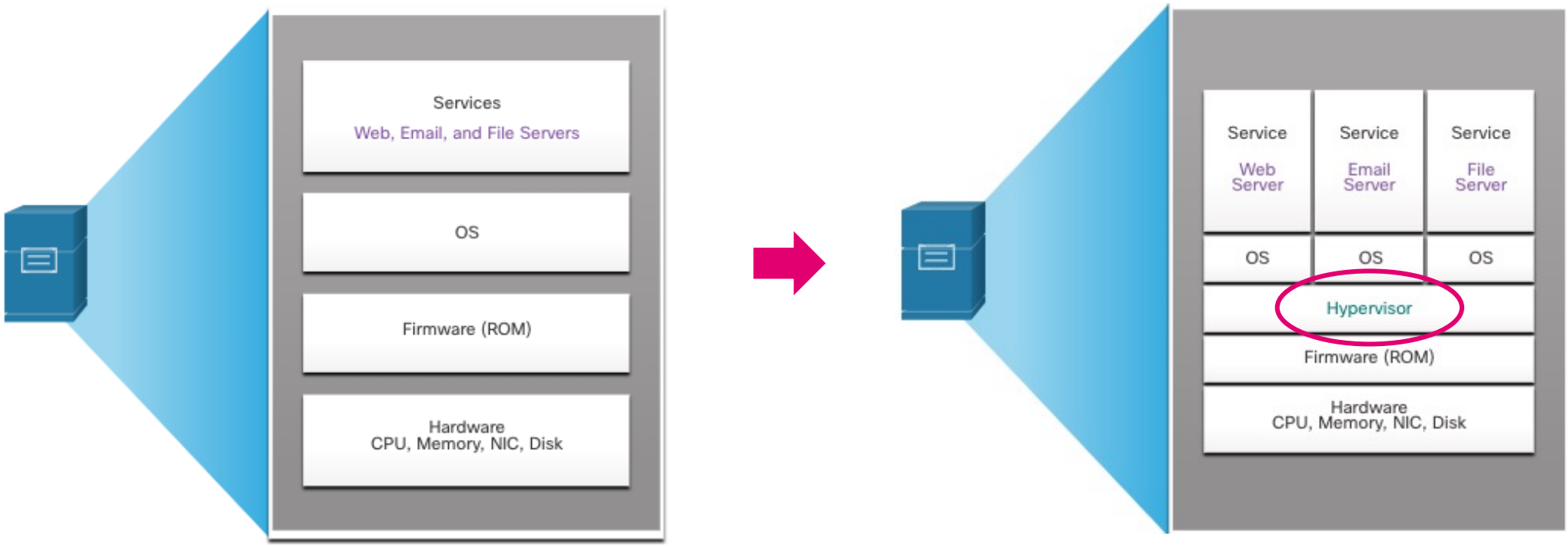
- Less equipment is required
- Less energy is consumed
- Less space is required
- ➔ Reduced cost **(?)**

Additional benefits:

- Easier prototyping
- Faster server provisioning
- Increased server uptime
- Improved disaster recovery
- Legacy support

Virtualization

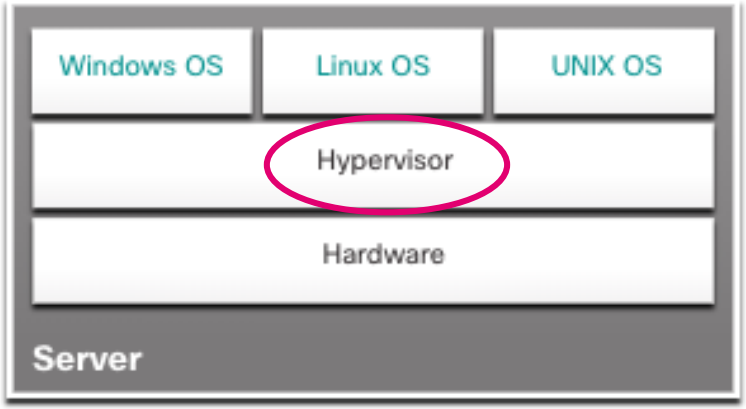
Type I Hypervisor



Virtualization

Type I Hypervisor

- "bare metal": installed directly on the hardware
- enterprise servers and data center networking devices
- Type I hypervisors have direct access to hardware resources → more efficient than hosted architectures (Type II)



Management Console & Type I Hypervisor functions

- A *Management console* is required to manage the lifecycle of the VMs executing on the Hypervisor

- Additional functions of a Type I Hypervisor:
  - Recover from hardware failure
  - Server over allocation

- Examples of Type I Hypervisors:
  - Cisco Unified Computing System (UCS)
  - VMware ESXi
  - Microsoft Hyper-V
  - Oracle VM Server for x86
  - KVM
  - …

Type II Hypervisor

- A Type 2 hypervisor (hosted hypervisor) is software that creates and runs VM instances. The computer, on which a hypervisor is supporting one or more VMs, is a host machine.

Live Quiz: Mentimeter

https://www.menti.com/al3a7r6vzu6h

www.menti.com Code: ***1541 0965***

Mentimeter

Virtual Network Infrastructure

Network Virtualization

Network functions can (also) be virtualized:

- Each network device: segmented into multiple virtual & independent devices
  - Examples: subinterfaces, virtual interfaces, VLANs, and routing tables.
  - Virtualized routing is called virtual routing and forwarding (VRF).

For real life enterprise networks this only works when automated to a high degree

How do we “split” Network Devices in order to virtualize them?

Software-Defined Networking

Network Virtualization Technologies

- **Software-Defined Networking (SDN)** - A network architecture that virtualizes the network, offering a new approach to network administration and management that seeks to simplify and streamline the administration process.

- **Cisco Application Centric Infrastructure (ACI)** - A purpose-built hardware solution for integrating cloud computing and data center management.

Software Defined Networking

- **OpenFlow** - This approach was developed at Stanford University to manage traffic between routers, switches, wireless access points, and a controller. The OpenFlow **protocol** is a basic element in building SDN solutions

- **OpenStack** - This approach is a virtualization and orchestration **platform** designed to build scalable cloud environments and provide an IaaS solution. OpenStack is often used with Cisco ACI.

- Orchestration in networking is the **process** of automating the provisioning of network components such as servers, storage, switches, routers, and applications.

Software-Defined Networking

Control Plane and Data Plane

Control plane (brains)

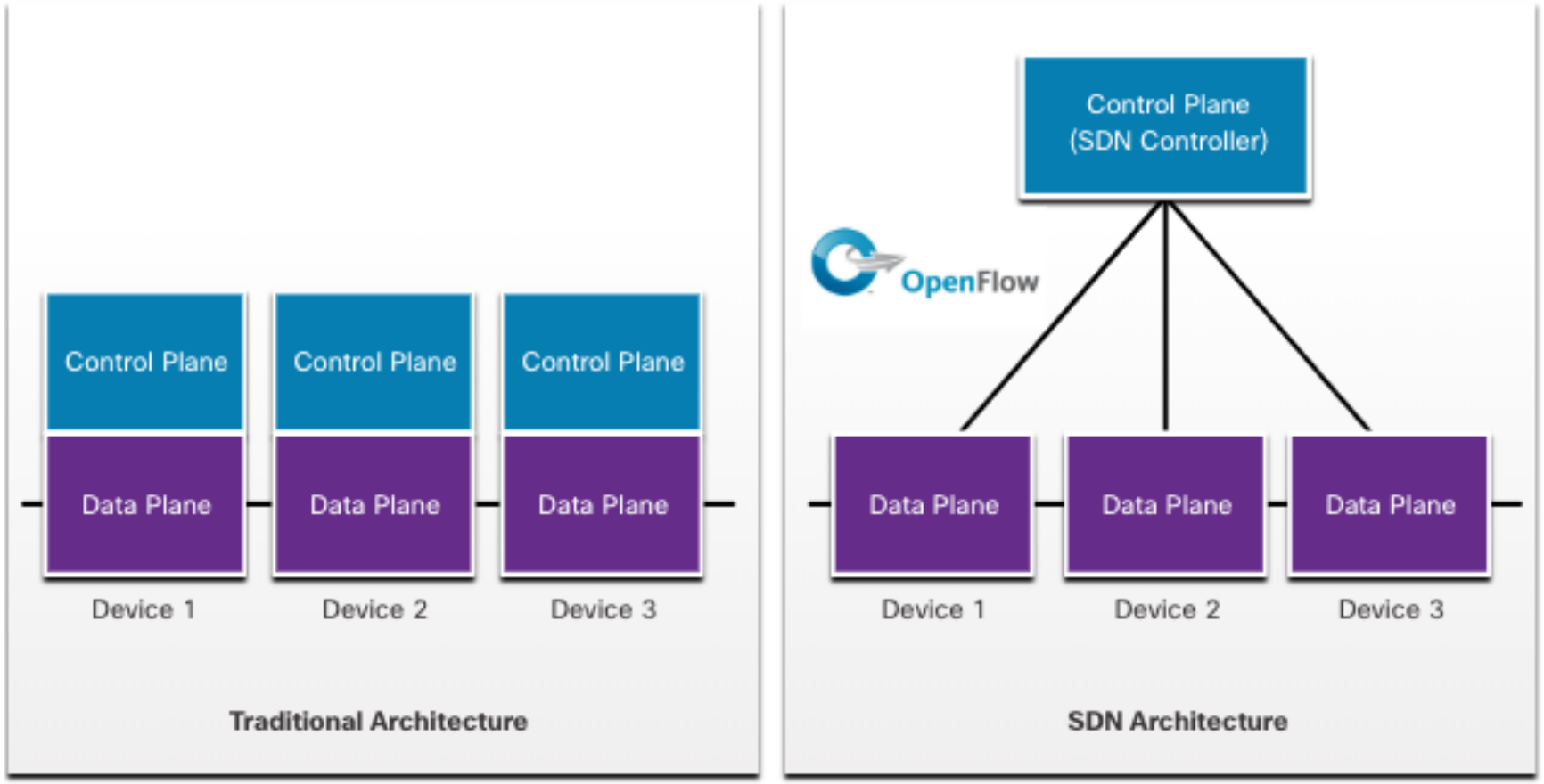
- Used to make forwarding **decisions**
- Layer 2 and Layer 3 route forwarding mechanisms
- Processed by the CPU
- Examples: routing protocol neighbor tables, routing protocol topology tables, IPv4 and IPv6 routing tables, STP, ARP table

Data plane (forwarding plane)

- Switch fabric connecting the various network ports on a device
- Implements/executes the forwarding
- Information processed by a special data plane processor (**no CPU involvement**)

Software-Defined Networking

Traditional and SDN Architectures

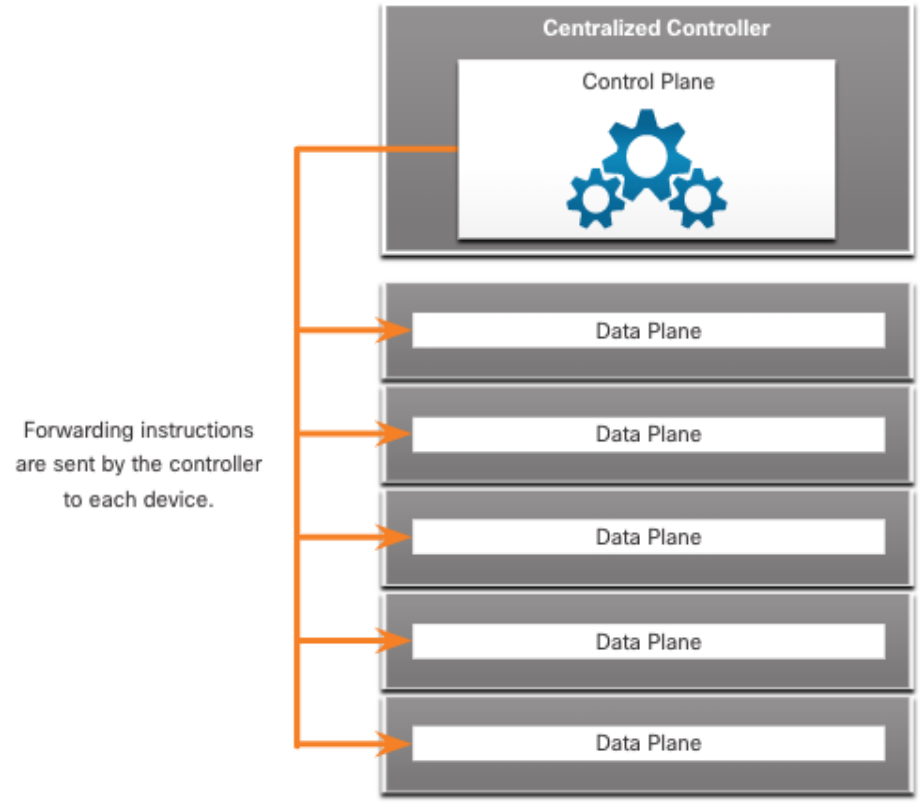


Software-Defined Networking

Control Plane, Data Plane & Management Plane

SDN: separation of the control plane and data plane

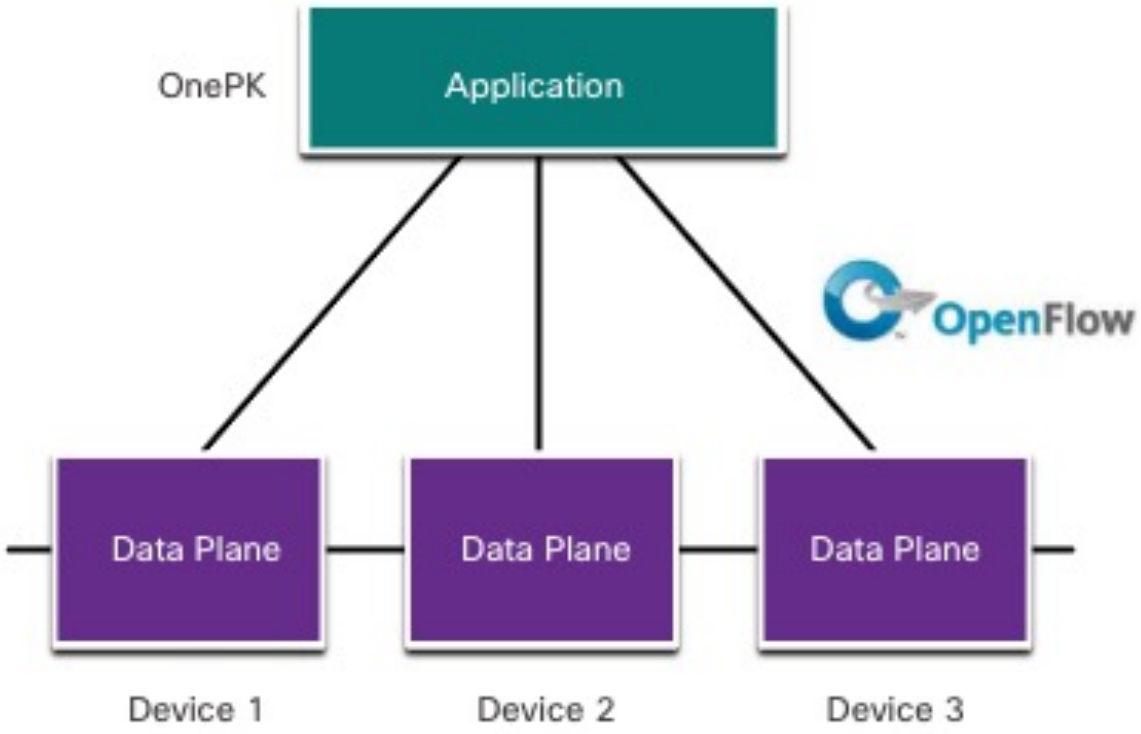
- Control plane: removed from each device and performed by a centralized controller
- Centralized Controller: communicates control plane functions to each device
- Devices: focus on forwarding data
- Management Plane: managing a device through its connection to the network
  - Accessed by Network administrators to configure a device



Controllers

SDN Types

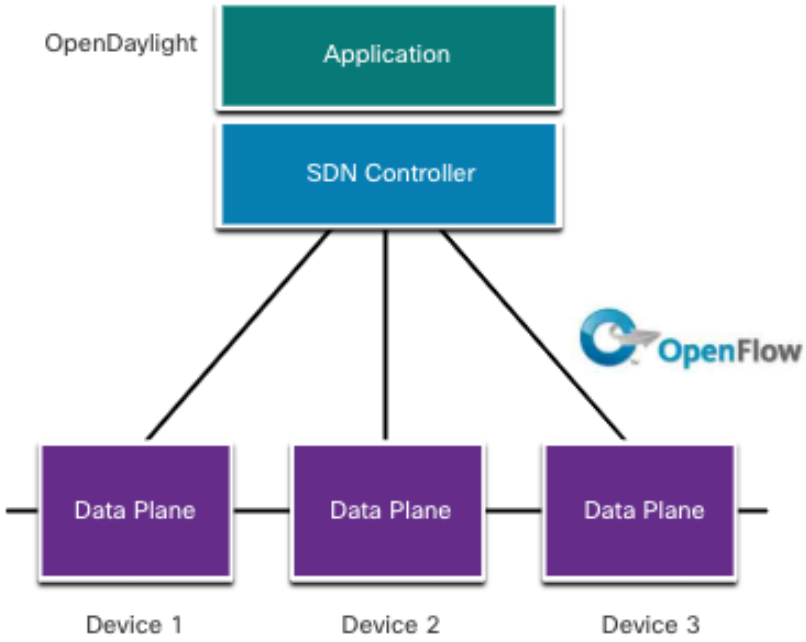
- **Device-based SDN:** Devices are programmable by applications running on the device itself or on a server in the network



Controllers

SDN Types

Controller-based SDN: Uses a centralized controller that has knowledge of all devices in the network. The applications can interface with the controller responsible for managing devices and manipulating traffic flows throughout the network. The Cisco Open SDN Controller is a commercial distribution of OpenDaylight.



Controllers

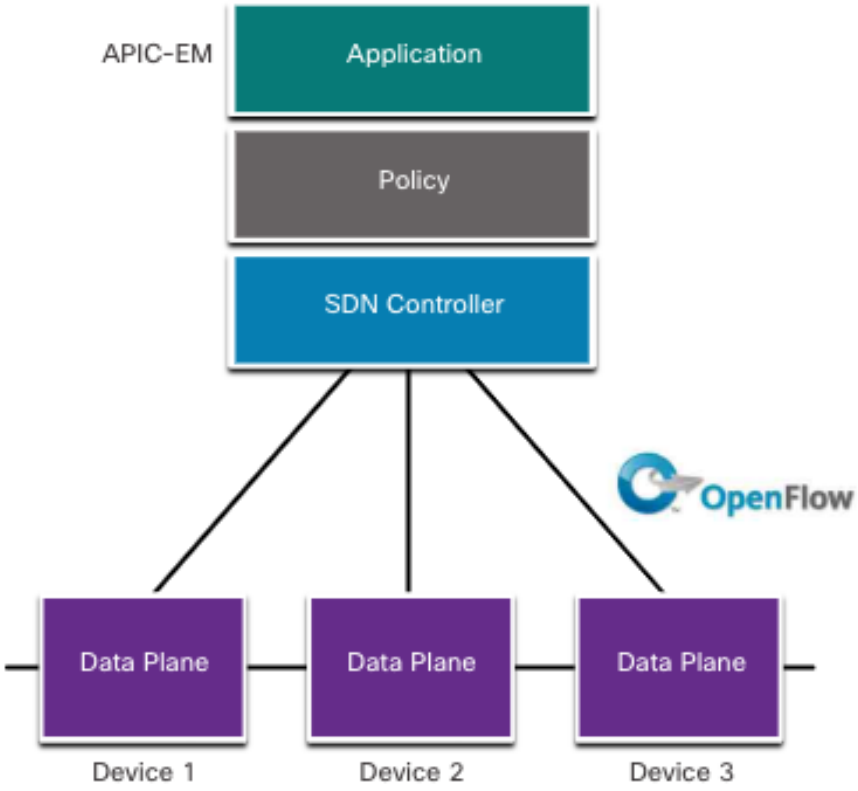
SDN Types (Cont.)

Policy-based SDN: Similar to controller-based SDN: centralized controller.

Additional Policy layer that operates at a higher level of abstraction.

Automate advanced configuration tasks via a guided workflow and user-friendly GUI

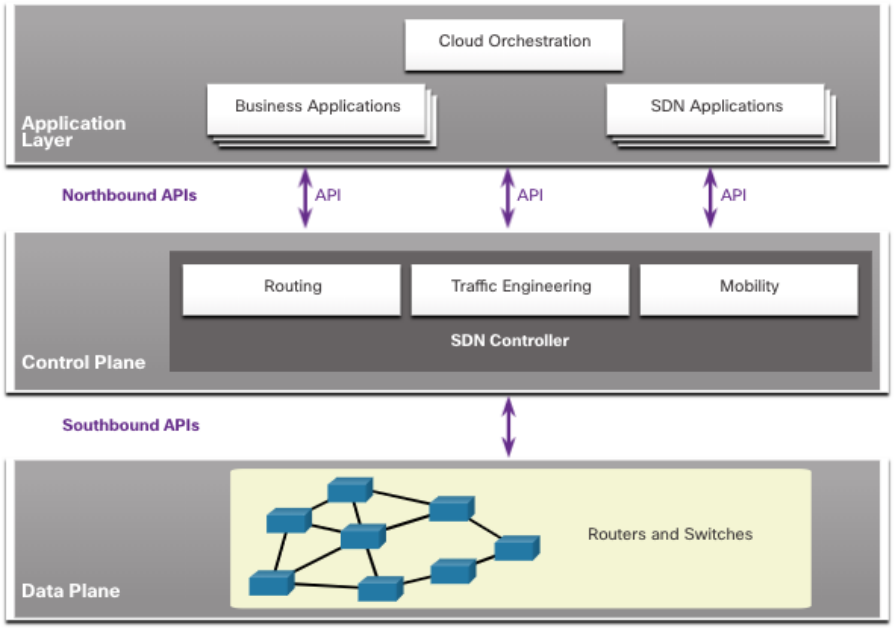
Example: Cisco APIC-EM



Software-Defined Networking

The SDN Controller

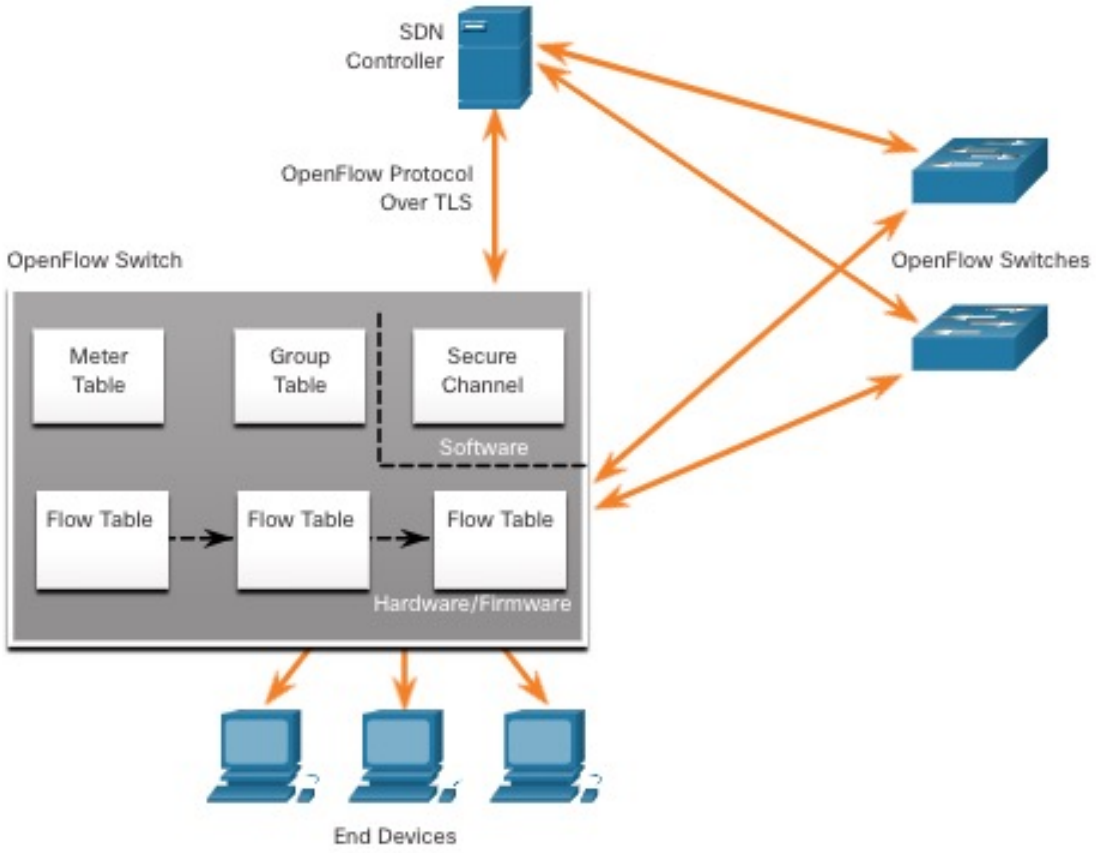
- Logical entity
- Enables management of how the data plane of switches and routers should handle network traffic
- Orchestrates, mediates, and facilitates communication between applications and network elements
- Uses northbound APIs to communicate with the upstream applications, helping network administrators shape traffic and deploy services
- Uses southbound APIs to define the behavior of the data planes on downstream switches and routers
- OpenFlow is a widely implemented southbound API



Controllers

SDN Controller and Operations

- SDN controller: defines the data flows between the centralized control plane and the data planes on individual routers and switches
- Each flow: must get permission from the SDN controller (e.g. security policy check)
- Complex functions performed by the controller

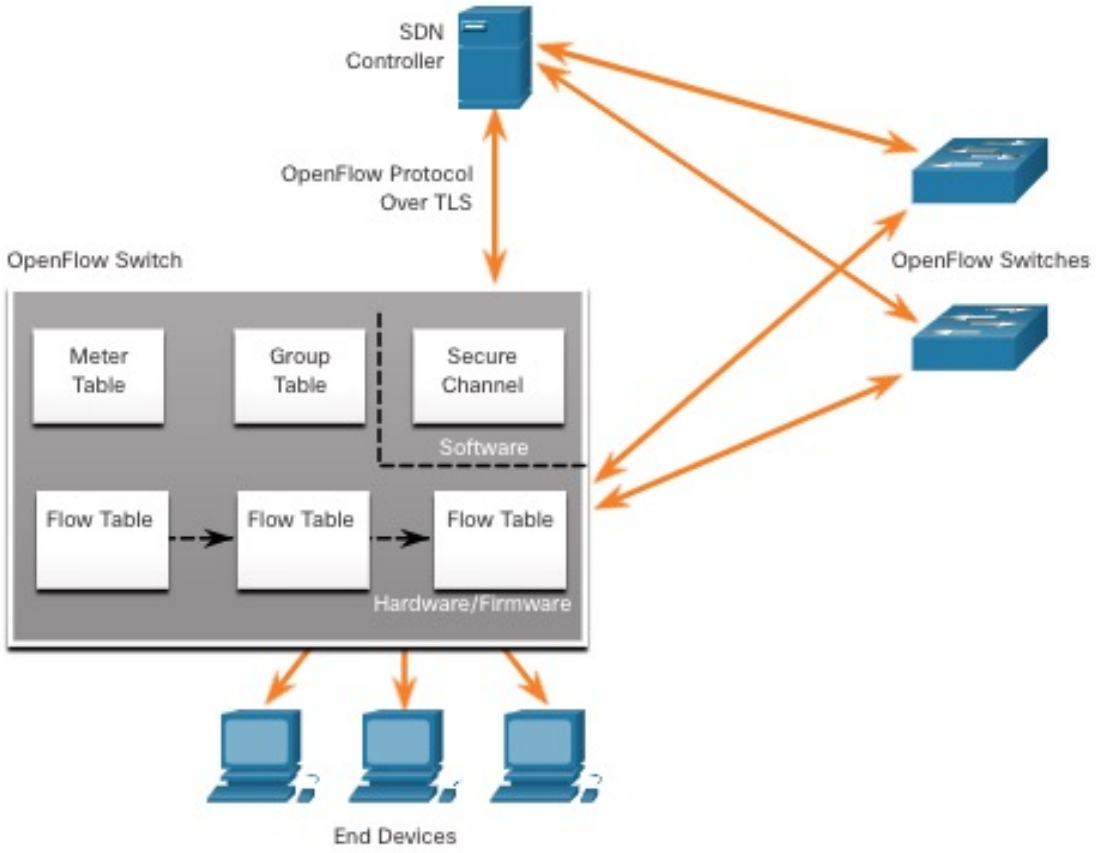


Controllers

SDN Controller and Operations

Switch tables are used to manage the flows of packets through the switch:

- **Flow Table** - This table matches incoming packets to a particular flow and specifies the functions that are to be performed on the packets.
- **Group Table** - Actions that affect one or more flows (aggregation).
- **Meter Table** - This table triggers a variety of performance-related actions on a flow including the ability to rate-limit the traffic.



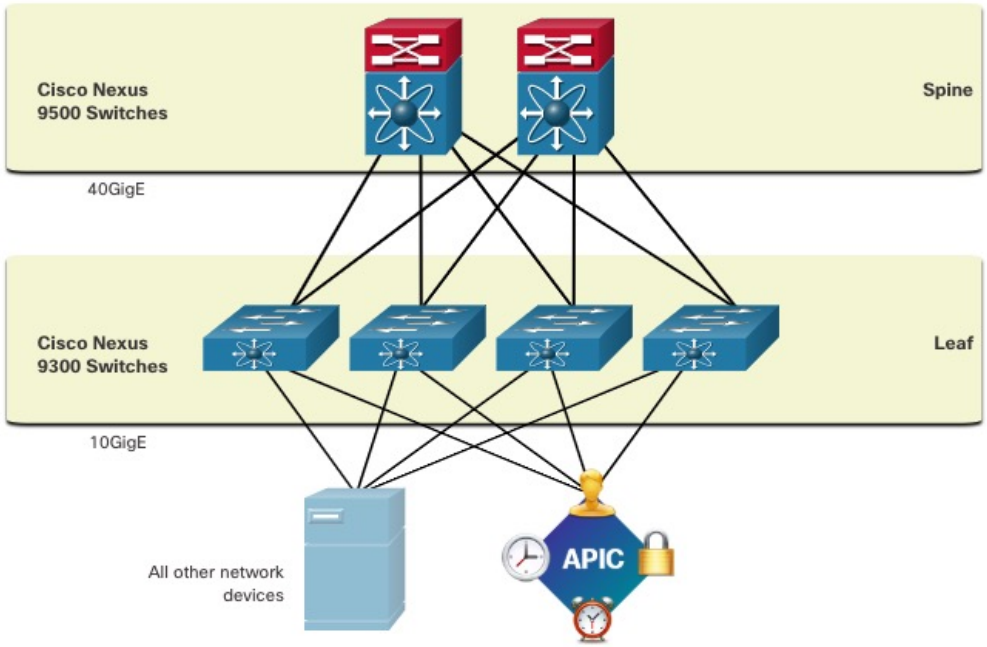
Controllers

Spine-Leaf Topology

- Two-tier spine-leaf topology:
- Leaf switches: attach to the spines, never to each other
- Spine switches: only attach to the leaf and core switches (not shown)

- The APIC controller does not manipulate the data path directly

- The APIC centralizes the policy definition and programs the leaf switches to forward traffic based on the defined policies



IBN and Cisco DNA Center

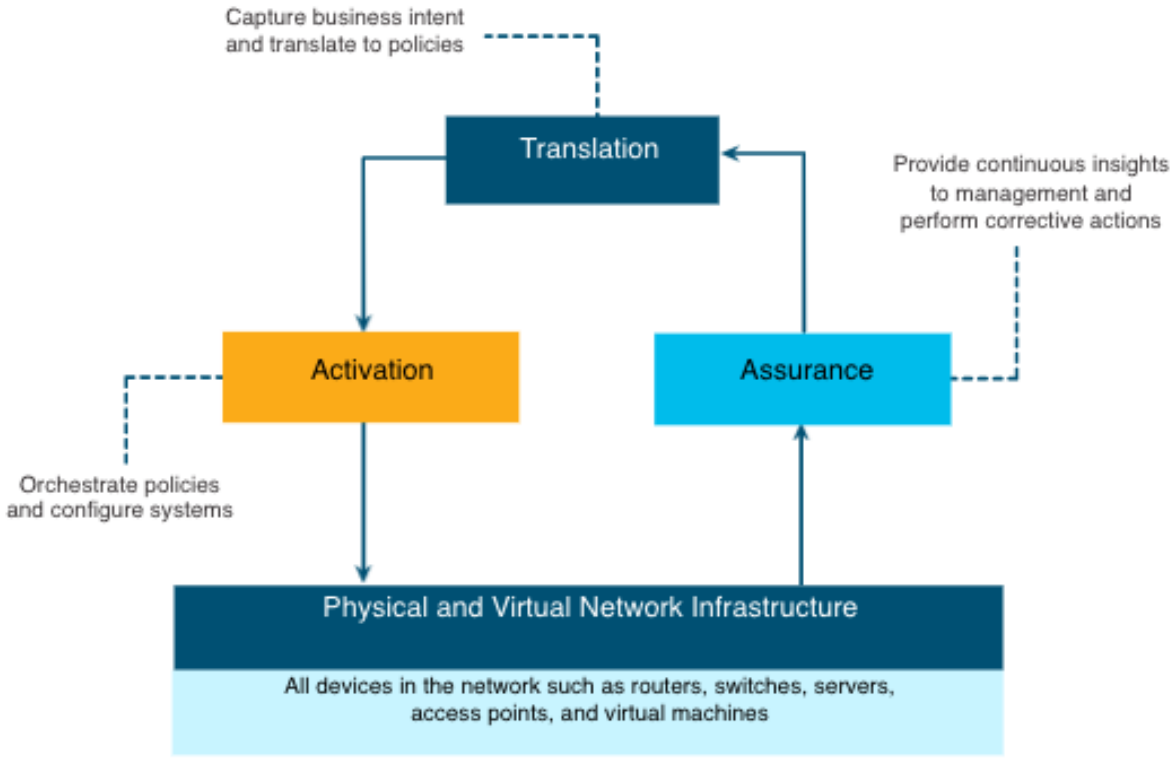
Intent-Based Networking (IBN)

- Emerging industry model for next generation of networking
- Builds on Software-Defined Networking (SDN)
- Designing and operating networks: Hardware-centric and manual → software-centric and fully automated
- Intent: business objectives for the network
- IBN captures intent and uses analytics, machine learning, and automation to align the network continuously and dynamically as business needs change
- IBN translates business intent into network policies that can be automated and applied consistently across the network

IBN and Cisco DNA Center

Intent-Based Networking Overview (Cont.)

Cisco views IBN as having three essential functions: translation, activation, and assurance.



Software Defined Networking

- Today we have covered an introduction to SDNs
- We will go deeper in this topic next week with Florian Wamser: SDNs from a Research and Practical standpoint
- Please prepare for Florian's guest the lecture:
  - Review the introduction covered today
  - Small Research: How are SDNs being used in practice today?
    - Do you have examples from your own companies (BB-Studenten) ?
    - Are Openflow-based SDN prevalent or do vendors push their own proprietary solutions?

Live Quiz: Mentimeter

https://www.menti.com/al3a7r6vzu6h

www.menti.com Code: ***1541 0965***

Mentimeter

Our Plan for today

Time (ca.)	Topic	Activity Type
5 min	Administrivia	Plenum
70 min	Network Virtualization (ENSA13)	Plenum + Live Quiz
15 min	Pause	
30 min	Network Automation (ENSA14)	Plenum + Live Quiz
20 min	Netacad Review, Q&A	Individual + Plenum

Our Plan for today

Time (ca.)	Topic	Activity Type
5 min	Administrivia	Plenum
70 min	Network Virtualization (ENSA13)	Plenum + Live Quiz
15 min	Pause	
30 min	Network Automation (ENSA14)	Plenum + Live Quiz
20 min	Netacad Review, Q&A	Individual + Plenum

Automation Overview

The Benefits of Automation

- Machines can work 24/7
- Machines provide a more uniform product
- Collection of vast amounts of data to be quickly analyzed
- Decision-making
- Robots: dangerous conditions / reduce risk to humans.
- Fast reaction
- **Dealing with complexity and unanticipated changes**

Automation Overview

Smart Devices

- Whenever a device takes a course of action based on an outside piece of information
- Ability to alter its behavior depending on its environment
- **Programmed** to do so!

Common Data Formats

- Data formats: way to store and exchange data in a structured format.

- Common data formats:

  - eXtensible Markup Language (XML)
  - JavaScript Object Notation (JSON)
  - YAML Ain't Markup Language (YAML)

Data Format Rules

```
{"message": "success", "timestamp": 1560789216, "iss_position": {"latitude": "25.9990",
"longitude": "-132.6992"}}
```

- Syntax: types of brackets used, such as [ ], ( ), { }, the use of white space, or indentation, quotes, commas, etc.
- Data type representation and support: how objects are represented
- Hierarchy: How objects relate to each other

Data Formats

Compare Data Formats

```
{
    "message": "success",
    "timestamp": 1560789260,
    "iss_position": {
        "latitude": "25.9990",
        "longitude": "-132.6992"
    }
}
```

JSON Format

```
message: success
timestamp: 1560789260
iss_position:
    latitude: '25.9990'
    longitude: '-132.6992'
```

YAML Format

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
<message>success</message>
<timestamp>1560789260</timestamp>
<iss_position>
   <latitude>25.9990</latitude>
   <longitude>-132.6992</longitude>
</iss_position>
</root>
```

XML Format

Data Formats

Java Script Object Notation (JSON)

- Human readable
- Very popular
- Easy to parse
- Can be used with most modern programming languages

Data Formats

JSON (Cont.)

```
GigabitEthernet0/0/0 is up, line protocol is up (connected)
   Description: Wide Area Network
   Internet address is 172.16.0.2/24
```

```
{
"ietf-interfaces:interface": {
    "name": "GigabitEthernet0/0/0",
    "description": "Wide Area Network”,
    "enabled": true,
    "ietf-ip:ipv4": {
        "address": [
            {
            "ip": "172.16.0.2",
            "netmask": "255.255.255.0"
            }
        ]
    }
  }
}
```

Data Formats

JSON Syntax Rules

- hierarchical structure, nested values
- { } hold objects
- [ ] hold arrays (, as separator)
- Key/value pairs:
  - Keys must be strings within double quotation marks " ". (≠ Python Dict!)
  - Values must be a valid JSON data type (string, number, array, Boolean, null, or another object)
  - Separated by a colon
- Multiple key/value pairs within an object separated by commas
- White space is irrelevant

Data Formats

YAML Data Format

- Superset of JSON
  - Comments
  - Extensible Data Types
  - Reational Anchors
  - …
- Focus on readability: no brackets, parentheses , …
- Indentation: used to define structure

Data Formats

YAML Data Format (Cont.)

```
{
   "ietf-interfaces:interface": {
      "name": "GigabitEthernet2",
      "description": "Wide Area Network",
      "enabled": true,
      "ietf-ip:ipv4": {
         "address": [
            {
               "ip": "172.16.0.2",
               "netmask": "255.255.255.0"
            },
            {
               "ip": "172.16.0.3",
               "netmask": "255.255.255.0"
            },
            {
               "ip": "172.16.0.4",
               "netmask": "255.255.255.0"
            }
         ]
      }
   }
}
```

```
ietf-interfaces:interface:
   name: GigabitEthernet2
   description: Wide Area Network
   enabled: true
   ietf-ip:ipv4:
      address:
         - ip: 172.16.0.2
           netmask: 255.255.255.0
         - ip: 172.16.0.3
           netmask: 255.255.255.0
         - ip: 172.16.0.4
           netmask: 255.255.255.0
```

Data Formats

Extensible Markup Language (XML)

- Like HTML
- Self-descriptive: encloses data within a related set of tags: **<tag>data</tag>**
- Unlike HTML, no predefined tags or document structure.
- XML objects: **<key>value</key>**
- **Old/legacy**

Data Formats

XML Data Format (Cont.)

```
{
   "ietf-interfaces:interface": {
      "name": "GigabitEthernet2",
      "description": "Wide Area Network",
      "enabled": true,
      "ietf-ip:ipv4": {
         "address": [
            {
               "ip": "172.16.0.2",
               "netmask": "255.255.255.0"
            },
            {
               "ip": "172.16.0.3",
               "netmask": "255.255.255.0"
            },
            {
               "ip": "172.16.0.4",
               "netmask": "255.255.255.0"
            }
         ]
      }
   }
}
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<ietf-interfaces:interface>
   <name>GigabitEthernet2</name>
   <description>Wide Area Network</description>
   <enabled>true</enabled>
   <ietf-ip:ipv4>
      <address>
         <ip>172.16.0.2</ip>
         <netmask>255.255.255.0</netmask>
      </address>
      <address>
         <ip>172.16.0.3</ip>
         <netmask>255.255.255.0</netmask>
      </address>
      <address>
         <ip>172.16.0.4</ip>
         <netmask>255.255.255.0</netmask>
      </address>
   </ietf-ip:ipv4>
</ietf-interfaces:interface>
```

Live Quiz: Mentimeter

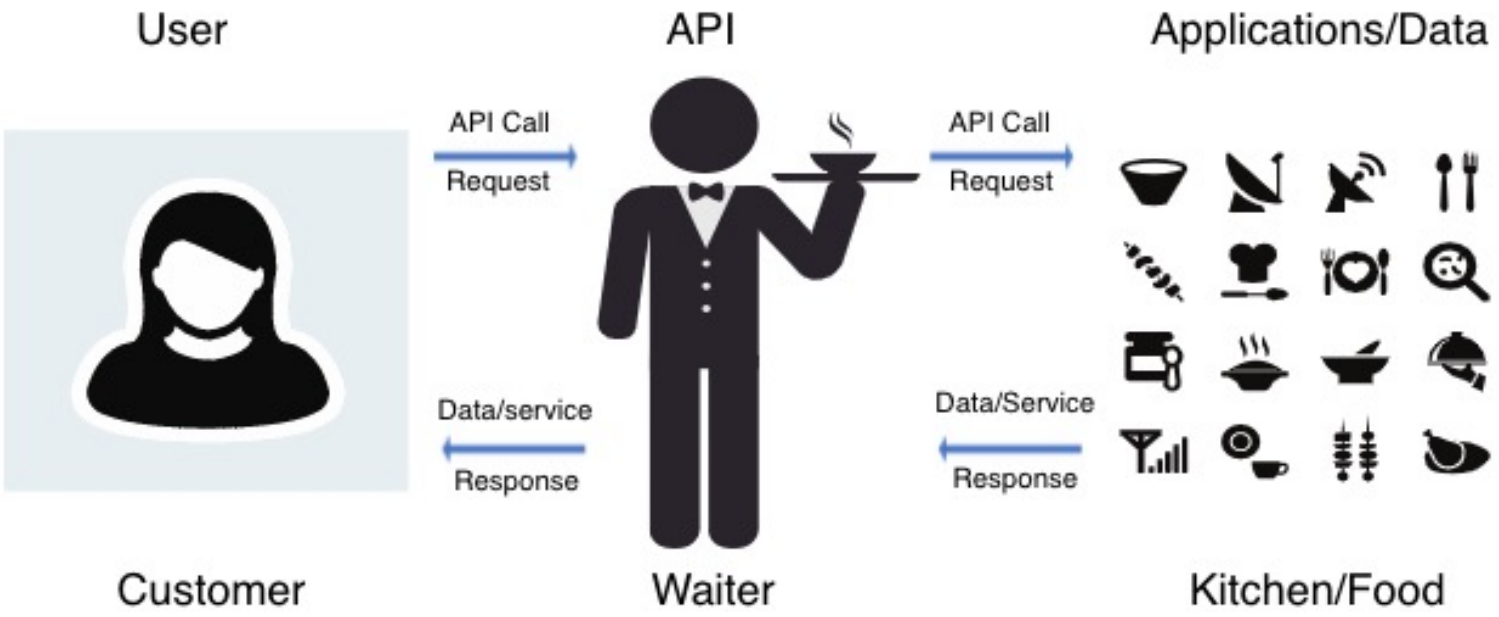
https://www.menti.com/al3a7r6vzu6h

www.menti.com Code: ***1541 0965***

Mentimeter

APIs

The API Concept



APIs

Open, Internal, and Partner APIs

- **Open APIs or Public APIs -** These APIs are publicly available and can be used with no restrictions.
- **Internal or Private APIs -** These are APIs that are used by an organization or company to access data and services for internal use only.
- **Partner APIs -** These are APIs that are used between a company and its business partners or contractors to facilitate business between them.

APIs

Types of Web Service APIs

Web service: service available over the internet using the World Wide Web. Types of APIs:

Characteristic	SOAP	REST	XML-RPC	JSON-RPC
Data Format	XML	JSON, XML, YAML, and others	XML	JSON
First released	1998	2000	1998	2005
Strengths	Well-established	Flexible formatting and most widely used	Well-established, simplicity	Simplicity

- Simple Object Access Protocol (SOAP)
- **Representational State Transfer (REST)**
- eXtensible Markup Language-Remote Procedure Call (XML-RPC)
- JavaScript Object Notation-Remote Procedure Call (JSON-RPC)

Software-Defined Networking

REST and RESTful API

- REST APIs work on top of the HTTP protocol
- REST APIs define a set of functions developers can use to perform requests and receive responses via HTTP protocol methods
- "RESTful" means:
  - **Client-Server** - The client handles the front end and the server handles the back end. Either can be replaced independently of the other.
  - **Stateless** - No client data is stored on the server between requests. The session state is stored on the client.
  - **Cacheable** - Clients can cache responses to improve performance.

RESTful Implementation

RESTful web service as a collection of resources with:

- Base Uniform Resource Identifier (URI)
- Data format supported by the web service. (JSON, YAML, or XML, …)
- Set of operations (using HTTP methods).

Common HTTP methods include POST, GET, PUT, PATCH and DELETE. As shown in the following table, these correspond to RESTful operations: Create, Read, Update, and Delete (or CRUD).

HTTP Method	RESTful Operation
POST	Create
GET	Read
PUT/PATCH	Update
DELETE	Delete

Software-Defined Networking

URI, URN, and URL

A URI is a string of characters that identifies a specific network resource. A URI has two specializations:

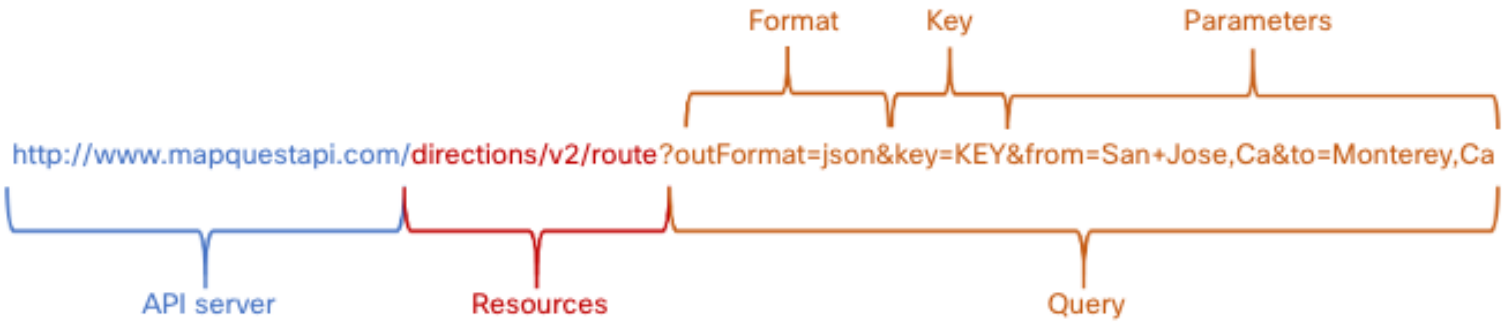
- **Uniform Resource Name (URN)** - identifies only the namespace of the resource (web page, document, image, etc.) without reference to the protocol or how to access it.
- **Uniform Resource Locator (URL)** - defines the network location of a specific resource. HTTP or HTTPS URLs are typically used with web browsers. Protocols such as FTP, SFTP, SSH, and others can use a URL. A URL using SFTP might look like: sftp://sftp.example.com.

These are the parts of the URI https://www.example.com/author/book.html#page155 :

- **Protocol/scheme** – HTTPS or other protocols such as FTP, SFTP, mailto, and NNTP
- **Hostname** - www.example.com
- **Path and file name** - /author/book.html
- **Fragment** - #page155

Anatomy of a RESTful Request (Cont.)

These are the different parts of the API request:



- **API Server** - This is the URL for the server that answers REST requests. In this example it is the MapQuest API server.
- **Resources** - Specifies the API that is being requested. In this example it is the MapQuest directions API.
- **Query** - Specifies the data format and information the client is requesting from the API service. Queries can include:
- **Parameters** - Parameters are used to send information pertaining to the request. In this example, the query parameters include information about the directions that the API needs so it knows what directions to return: "from=San+Jose,Ca" and "to=Monterey,Ca".
  - ~~**Format** – This is usually JSON but can be YAML or XML. In this example JSON is requested.~~
  - ~~**Key** - The key is for authorization, if required. MapQuest requires a key for their directions API. In the above URI, you would need to replace "KEY" with a valid key to submit a valid request.~~

Anatomy of a RESTful Request (Cont.)

Reasons why an API provider may require a key (aka. token):

- To authenticate the source to make sure they are authorized to use the API.

- To limit the number of people using the API.
- To limit the number of requests per user.

- To better capture and track the data being requested by users.
- To gather information on the people using the API.

Live Quiz: Mentimeter

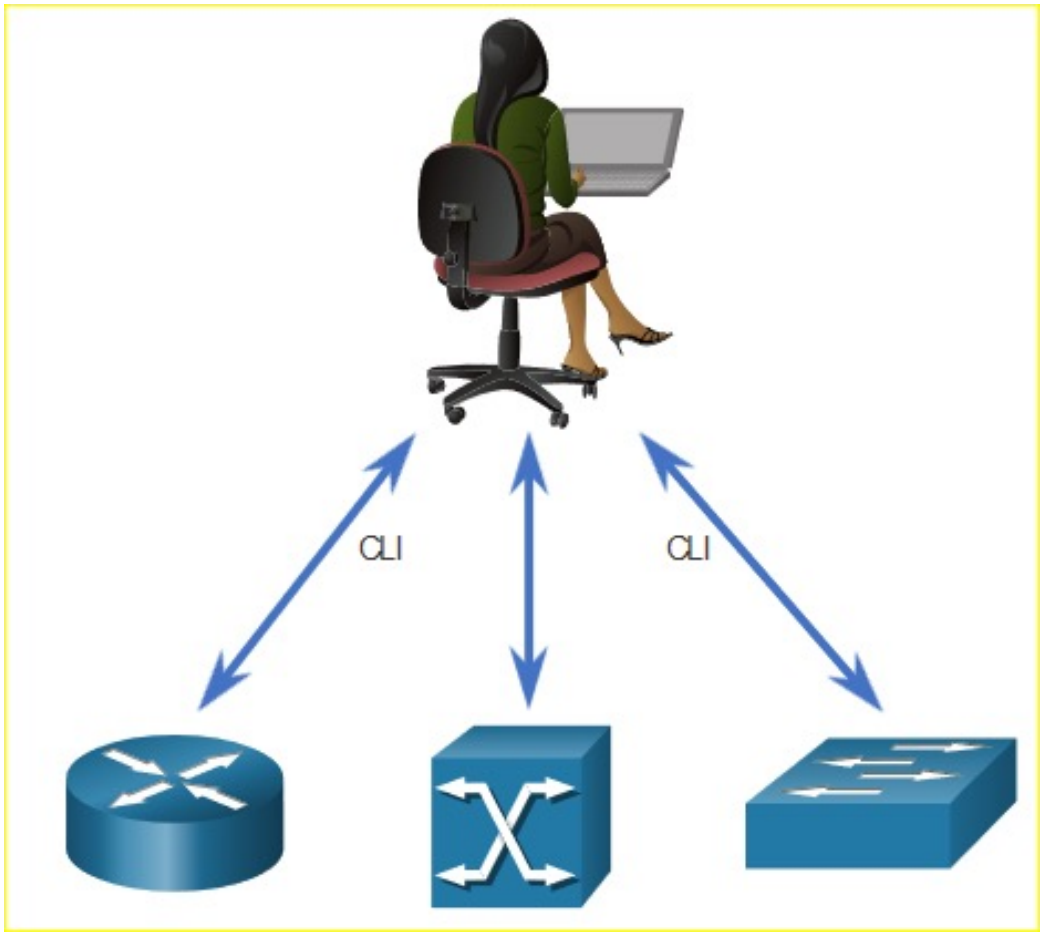
https://www.menti.com/al3a7r6vzu6h

www.menti.com Code: ***1541 0965***

Mentimeter

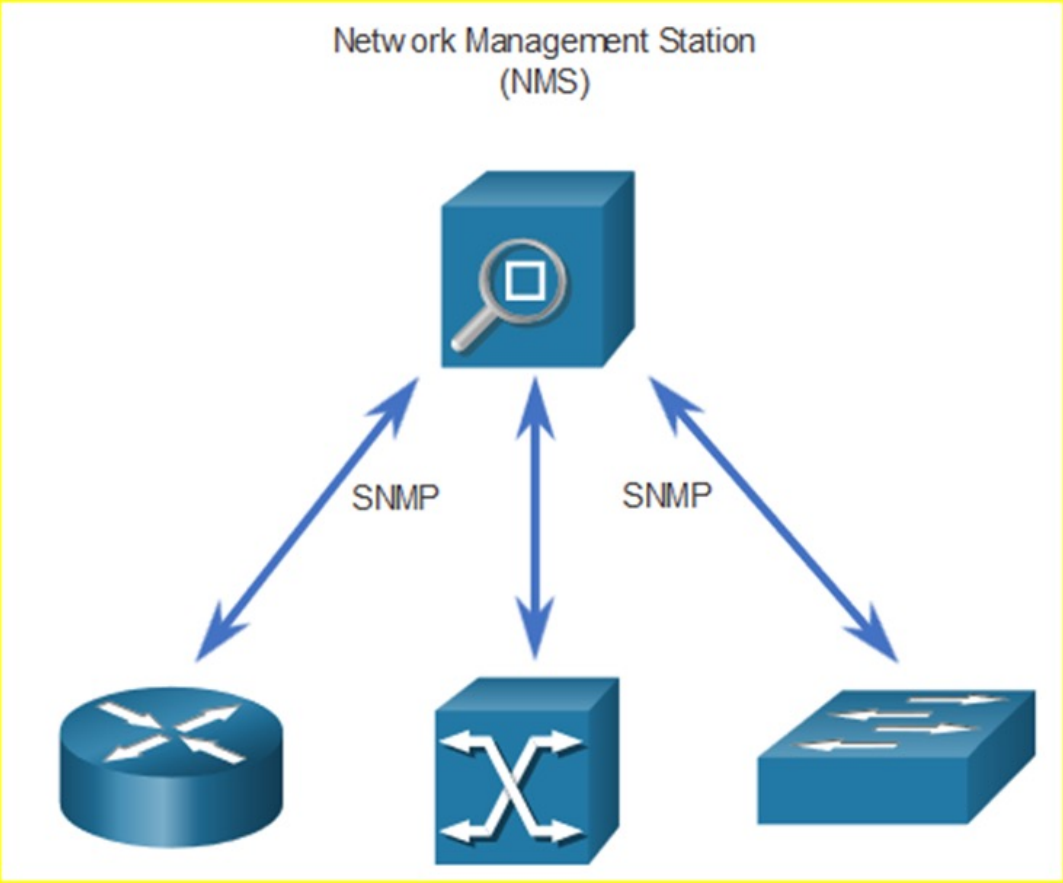
Configuration Management Tools

Traditional Network Configuration



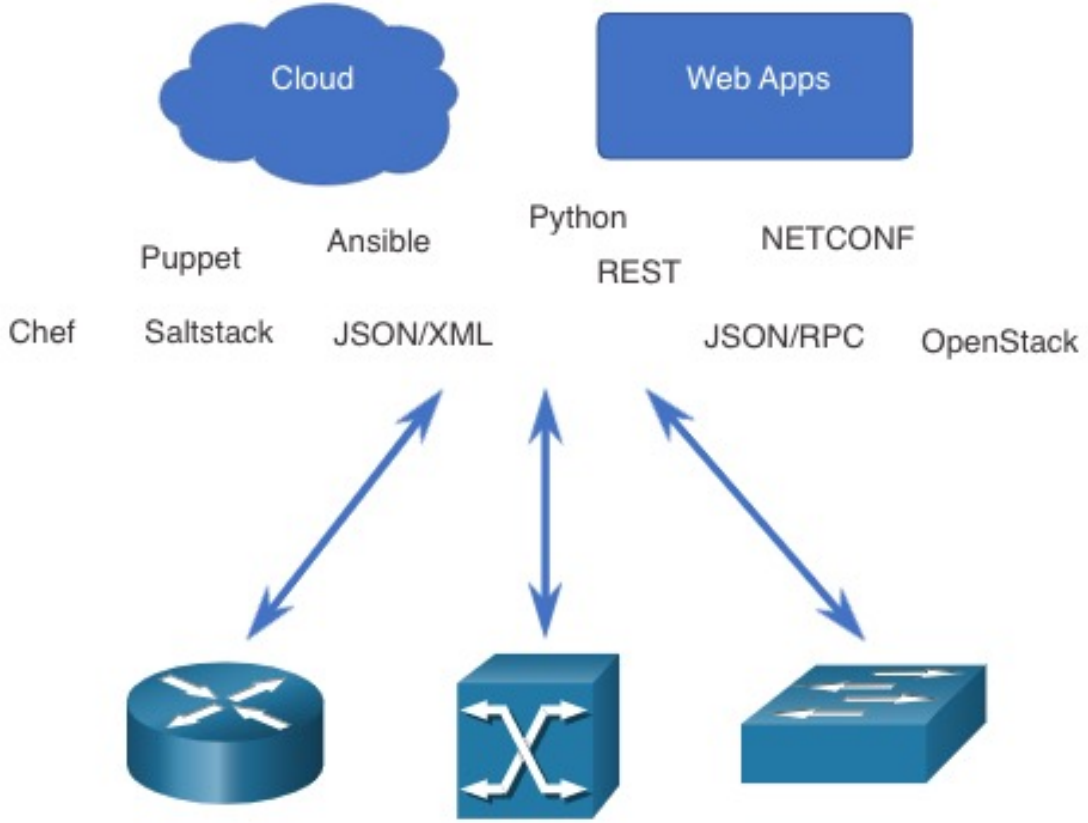
Configuration Management Tools

Traditional Network Configuration



Configuration Management Tools

Network Automation



Configuration Management Tools

- RESTful API requests to automate tasks and can scale across thousands of devices
- Automation:
  - Software and version control
  - Device attributes such as names, addressing, and security
  - Protocol configurations
  - ACL configurations
- Automation: tool automatically performs a task on a system
- Orchestration: arranging of the automated tasks that results in a coordinate process or workflow

Configuration Management Tools (Cont.)

There are several tools available to make configuration management easier:

- Ansible
- Chef
- Puppet
- SaltStack

Goal: reduce complexity and time involved in configuring and maintaining (large-scale) network infrastructures

Our Plan for today

Time (ca.)	Topic	Activity Type
5 min	Administrivia	Plenum
70 min	Network Virtualization (ENSA13)	Plenum + Live Quiz
15 min	Pause	
30 min	Network Automation (ENSA14)	Plenum + Live Quiz
20 min	Netacad Review, Q&A	Individual + Plenum