



MODULE 8: CISCO PLATFORMS AND DEVELOPMENT

DevNet Associate v1.0





Module Objectives

- Module Title: Cisco Platforms and Development
- Module Objective: Compare Cisco platforms used for collaboration, infrastructure management, and automation.
- It will comprise of the following sections:

Topic Title	Topic Objective
8.1 Introduction to Cisco Platforms	Describe the Cisco API platform.
8.2 Cisco SDKs	Explain how Cisco SDKs assist in the development of applications.
8.3 Understanding Network Programmability and Device Models	Compare network programmability models.
8.4 Cisco Network Management	Compare Cisco network management platforms.
8.5 Cisco Compute Management	Describe Cisco compute management solutions.
8.6 Cisco Collaboration Platforms	Describe Cisco collaboration platforms.
8.7 Cisco Security Platforms	Describe Cisco security platforms.



8.1 INTRODUCTION TO CISCO PLATFORMS





Understanding the Cisco API Platform

- Exploring the categories of Cisco technology
 - DevNet creates Dev Centers for each technology group and these Dev Centers make it very convenient for grouping technologies together.
 - List of Cisco Dev Centers is given below:
 - Cloud
 - Collaboration
 - Data center
 - Internet of Things (IoT) and edge computing
 - Networking
 - Security
 - Wireless and mobile
 - Application developers



8.2 Cisco SDKs





What is an SDK?

- Software Development Kit (SDK) contains a set of software development tools integrated for developing applications for a specific device or system.
- Most SDKs are a package, integrated with libraries, documents, code examples, and so on.
- SDKs often help with pagination or rate limiting constraints on responses for a particular API.
- Cisco SDKs
 - Cisco provides a wide range of SDKs on different Cisco platforms, such as:
 - Webex Teams Android SDK
 - Jabber Web SDK
 - Jabber Guest SDK for Web
 - Jabber Guest SDK for iOS
 - Jabber Guest SDK for Android
 - Cisco DNA Center Multivendor SDK
 - UCS Python SDK
 - Cisco APIC Python SDK (Cobra SDK)
 - Cisco IMC Python SDK
 - Cisco Instant Connect SDK
 - Webex Teams Python SDK



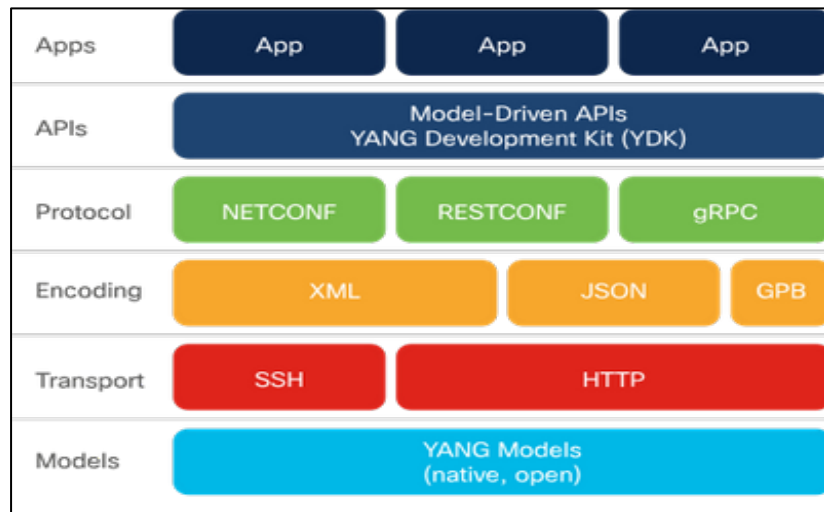
8.3 UNDERSTANDING NETWORK PROGRAMMABILITY AND DEVICE MODELS





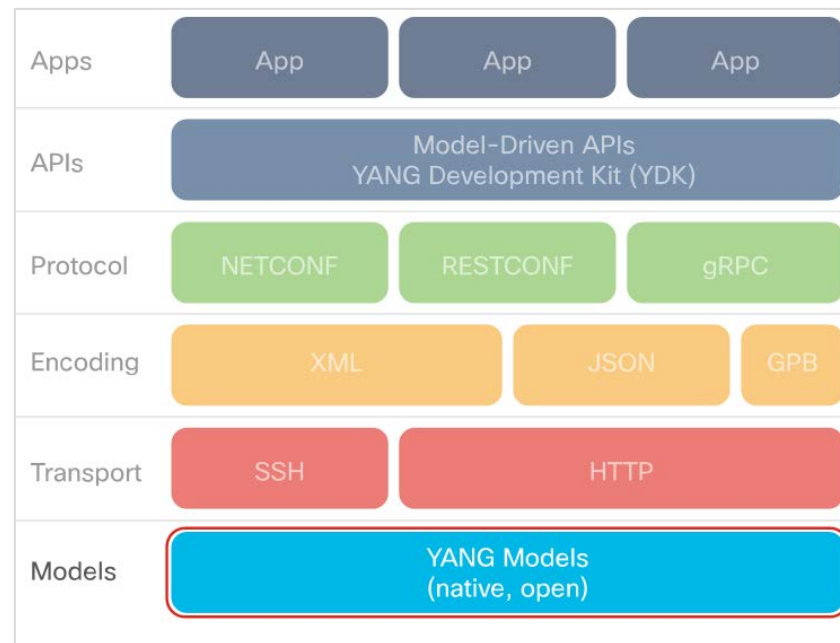
What is Model-Driven Programmability?

- Model-driven programmability:
 - Provides configuration language that is human-readable
 - Is Model-based, structured, and computer-friendly
 - Includes support for multiple model types, including native, OpenConfig, and IETF
 - Uses specification that is decoupled from transport, protocol end encoding.
 - Uses model-driven APIs for abstraction and simplification.
 - Leverages open-source and enjoys wide support.



What is YANG?

- Yet Another Next Generation, (YANG) as defined in RFC7519, is “a data modeling language used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF), NETCONF remote procedure calls, and NETCONF notifications.”
- In the real world, there are two types of YANG models:
 - **Open**
 - **Native**





What is YANG?

- Why we need YANG for device modeling?
 - YANG allows different network device vendors to describe their device type, configuration, and state to map to the device operation in programmatic way.
 - YANG defines four types of nodes for data modeling: **Leaf Nodes**, **Leaf-List Nodes**, **Container Nodes**, and **List Nodes**.

Yang Terms				
anyxml	data model	derived type	leaf	module
augment	data node	grouping	leaf-list	RPC
container	data tree	identifier	list	state data



What is YANG?

■ Yang in Action

- As the content of the Yang file increases, **pyang** is one of the tools used to extract the content in a more readable and concise way.
- You can install **pyang** using the **pip** command in a virtual environment.
- Using the **pyang** tool, you can convert a YANG file into an easy-to-follow tree structure as shown in the Pyang output example.

```

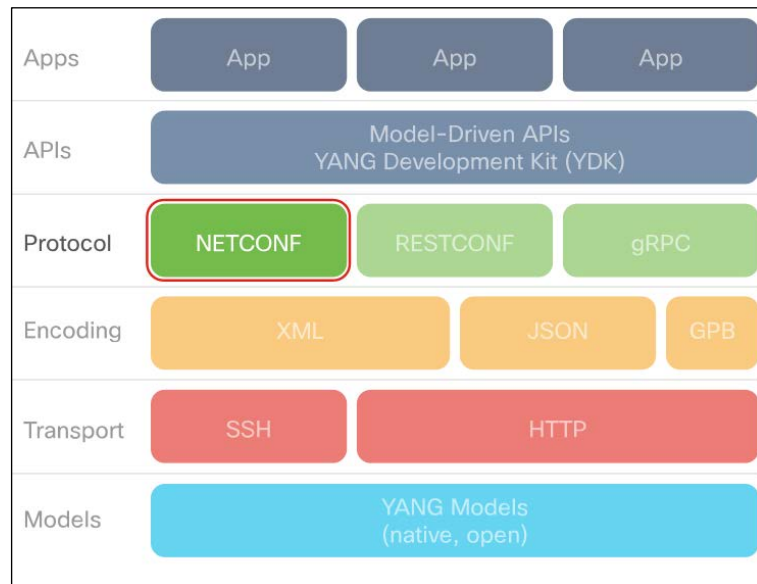
module: ietf-interfaces
+--rw interfaces
|
| +--rw interface* [name]
| |
| | +--rw name                string
| | +--rw description?        string
| | +--rw type                 identityref
| | +--rw enabled?            boolean
| | +--rw link-up-down-trap-enable? enumeration {if-mib}?
|
+--ro interfaces-state
+--ro interface* [name]
|
| +--ro name                string
| +--ro type                 identityref
| +--ro admin-status        enumeration {if-mib}?
| +--ro oper-status         enumeration
| +--ro last-change?        yang:date-and-time
| +--ro if-index            int32 {if-mib}?
| +--ro phys-address?       yang:phys-address
| +--ro higher-layer-if*    interface-state-ref
| +--ro lower-layer-if*     interface-state-ref
| +--ro speed?              yang:gauge64
|
+--ro statistics
|
| +--ro discontinuity-time   yang:date-and-time
| +--ro in-octets?          yang:counter64
| +--ro in-unicast-pkts?    yang:counter64
| +--ro in-broadcast-pkts? yang:counter64
| +--ro in-multicast-pkts? yang:counter64
| +--ro in-discards?       yang:counter32
| +--ro in-errors?         yang:counter32
| +--ro in-unknown-protos? yang:counter32
| +--ro out-octets?         yang:counter64
| +--ro out-unicast-pkts?   yang:counter64
| +--ro out-broadcast-pkts? yang:counter64
| +--ro out-multicast-pkts? yang:counter64
| +--ro out-discards?      yang:counter32
| +--ro out-errors?        yang:counter32

```



What is NETCONF?

- **Network Configuration (NETCONF)**, a protocol defined by the IETF RFC7519, is designed to install, manipulate, and delete the configuration of network devices.
- It is the primary protocol used with YANG data models today.
- **The NETCONF protocol uses XML-based data encoding for both the configuration data and the protocol messages.**
- It provides a small set of operations to manage device configurations and retrieve device state information.





What is NETCONF?

▪ NETCONF Protocol Operations

- The NETCONF protocol provides a set of operations to manage device configurations and retrieve device state information. The base protocol includes the following operations:

NETCONF Protocol Operation		
get	copy-config	unlock
get-config	delete-config	close-session
edit-config	lock	kill-session



What is NETCONF?

▪ NETCONF versus SNMP

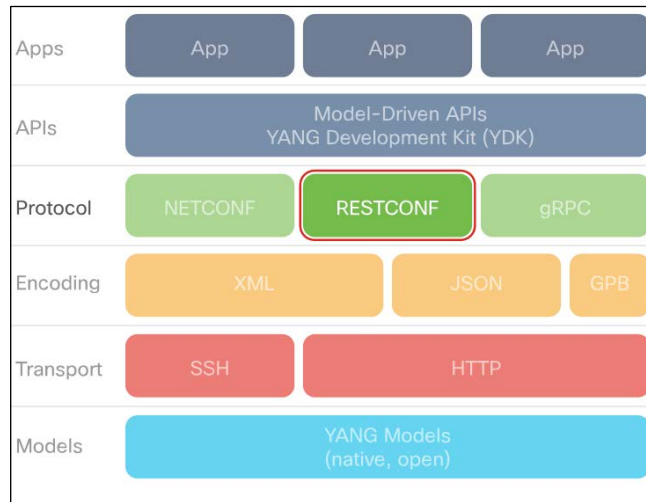
- NETCONF and SNMP protocols are both defined to remotely configure devices.
- Features of both are given below:

NETCONF	SNMP
<ul style="list-style-type: none"> • Multiple configuration data stores (candidate, running, startup) • Device-level and network-wide transactions • Configuration testing and validation • Extensible remote procedure calls • Built-in capability exchange 	<ul style="list-style-type: none"> • Uses pull model when retrieving data from device • Does not have a discovery process for finding Management Information Base (MIBs) supported by a device • Does not support the concept of transactions • Lacks backup-and-restore of element configuration • Limited industry support for configuration MIBs



What is RESTCONF?

- The RESTCONF RFC 8040 defines a protocol and mechanism for REST-like access to configuration information and control.
- RESTCONF uses datastore models and command verbs defined in NETCONF, encapsulated in HTTP messages.
- **RESTCONF uses structured data (XML or JSON) and YANG to provide REST-like APIs, enabling programmatic access to devices.**
- RESTCONF is not intended to replace NETCONF, instead provide an HTTP interface that follows the REST principles and is compatible with the NETCONF datastore model.





What is RESTCONF?

▪ RESTCONF vs. NETCONF

- Overall, NETCONF is more comprehensive, flexible, and complex than RESTCONF. The differences between NETCONF and RESTCONF:
 - RESTCONF is easier to learn and use for engineers with previous REST API experience.
 - NETCONF supports running and candidate data stores, while RESTCONF supports only a running data store as any edits of candidate data store are immediately committed.
 - RESTCONF does not support obtaining or releasing a data store lock. If a data store has an active lock, the RESTCONF edit operation will fail.
 - A RESTCONF edit is a transaction limited to a single RESTCONF call.
 - RESTCONF does not support transactions across multiple devices.
 - Validation is implicit in every RESTCONF editing operation, which either succeeds or fails.



What is RESTCONF?

- NETCONF operations and RESTCONF methods mapping

Description	NETCONF	RESTCONF
Create a data resource	<code><edit-config>, </edit-config></code>	POST
Retrieve data and metadata	<code><get-config>, <get> , </get-config></code>	GET
Create or replace a data resource	<code><edit-config> (nc:operation="create/replace")</code>	PUT
Delete a data resource	<code><edit-config> (nc:operation="delete")</code>	DELETE

- The RESTCONF RFC 8040 states that RESTCONF base URI syntax is `/restconf/<resource-type>/<yang-module:resource>`.
- `<resource-type>` and `<yang-module:resource>` are variables and the values are obtained using specific YANG model files.
- The basic format of a RESTCONF URL is `https://<hostURL>/restconf<resource><container><leaf><options>` where any portion after `restconf` could be omitted.



8.4 CISCO NETWORK MANAGEMENT





Network Management Platforms

- Network automation plays a crucial part in simplifying day-to-day operations and maintenance.
- Network automation is used for various common tasks in an organization:
 - Device Provisioning
 - Device software Management
 - Compliance Checks
 - Reporting
 - Troubleshooting
 - Data collection and telemetry
- Network programmability helps reduce Operational Expenses (OPEX), which represents a significant portion of overall network cost.
- Teams can speed up service delivery by automating tasks that are typically done by hand using a Command-Line Interface (CLI).



Cisco IOS XE

▪ What is Cisco IOS XE?

- With IOS XE, you have access to standards-based, consistent, programmable interfaces, standard data models for configuration, deployment, and rollback, as well as services integration with the network.
- IOS XE is based on Linux.
- Model-driven programmability support in Cisco IOS XE
- Cisco device YANG models can be obtained from the Cisco Device YANG models repository.
- IOS XE is used on most high-end Cisco platforms including switches, routers and gateways.
- NETCONF/YANG is supported as of IOS XE 16.3.1 software.



Cisco IOS XE

▪ Enabling model-driven programmability

- On some devices, Model-driven programmability services must first be enabled.
- Enter the configuration mode using the **configure terminal** command.

▪ Enable NETCONF on IOS XE

- NETCONF connections should be authenticated using AAA credentials.
- The CLI command to enable NETCONF is: **netconf-yang**

```
csr1000v-1#netconf-yang
% Bad IP address or host name% Unknown command or computer name, or unable to find computer address
```

▪ Troubleshooting:

- If there is an 'Unknown command' response to the **netconf-yang**, as shown below, then double-check the device configuration.
- Cisco IOS XE supports two datastores: running and candidate. It also supports locking datastores, as well as configuration rollback.



Cisco IOS XE

▪ Enable RESTCONF on IOS XE

- RESTCONF connections should be authenticated using AAA credentials.
- RESTCONF runs over HTTPS.
- The command enabled to support RESTCONF over port 443 is: **ip http secure-server**
- The CLI command to enable RESTCONF is: **restconf**

▪ Accessing YANG models

- For a complete look at Cisco YANG models, browse or clone the GitHub repository at <https://github.com/YangModels/yang>.
- The **vendor/cisco** subdirectory contains models for Cisco platforms.



Cisco DNA Center

- **What is Cisco DNA Center?**

- A Cisco DNA Center is a foundational controller and analytics platform for large and midsize organizations
- It supports full 360-degree services and integration: North, East, Southbound and Westbound

- **Cisco DNA Center dashboard (GUI):** The GUI organizes services and activities into Design, Policy, Provision, Assurance, and Platform.
- **Role-Based Access Control (RBAC):** The initial user, 'admin', is assigned the SUPER-ADMIN-ROLE allowing complete control of the DNA Center and access to all sections.
- **Product hardware:** The Cisco DNA Center appliance runs on dedicated Cisco Rack Servers in various scalable configurations.
- **Cisco DNA Center Intent API:** The Intent API provides the means to programmatically access the Cisco DNA Center services and functionality.



Cisco DNA Center – Intent API

■ Intent API documentation:

- The Intent API documentation is found in the DevNet Cisco DNA Center, under section **Cisco DNA Center - Intent API**.
- Additional documentation and a built-in 'TryIt' capability is available in the product GUI, under **Platform › Developer Toolkit**.

Documentation > DNA Center Platform

API Lifecycle

The Cisco DNA Center platform deploys and supports APIs according to Cisco Secure Development Lifecycle (CSDL) best practices.

- **Early Field Trial (EFT)** - Unsupported; experimental. Released for feedback from early adopters. Functionality is not guaranteed. API design is not final:
 - A future versions of the API may implement (incompatible) changes to request/response payload, request path or parameters.

Cisco DNA Center DESIGN POLICY PROVISION ASSURANCE **PLATFORM**

Overview Manage Developer Toolkit Runtime Dashboard

Welcome to the DNA Center Platform. Programmatically access your network through Intent APIs, integrate with your preferred IT systems to create end-to-end solutions and add support for multi-vendor devices.

Bundles
Bundles are easy to use feature sets for consuming Intent APIs, integrations, events and notifications. View all the available bundles, enable relevant bundles and customize the configuration preferences to consume events as per your application(s) or IT system(s) needs.

Developer Toolkit
Discover APIs to manage your network, configure integration flows and access network data to analyze, export and visualize complex reports.

Runtime Dashboard
Get insights into API usage, view events published to IT systems such as number of API calls, response time(s), events published, bundles activated etc.

Configurations
View and set global or bundle specific settings to manage your integration configurations and modify event specific settings.



Cisco DNA Center – Intent API

■ Method description, URI, and Parameters

- The DevNet documentation is organized by subdomain groups. The method URI is given in an abbreviated format, showing only the URL suffix.
- The GUI Platform Developer Toolkit is organized by Domain: Subdomain. It includes specific expanded URL using the IP address of the DNA Center.
- Clicking any subdomain name displays list of methods associated with that subdomain.

Intent API <small>RESTCONF/JSON/NETCONF</small> <small>Cisco DNA Center Platform v. 1.3.3.x</small>	
Filter by tag	
Authentication	Access Token Request
Sites	Create site, assign devices to them and get site health
Topology	Get topology details and overall network health
Devices	Manage network devices
Clients	Get client (by MAC Address) health, status, and information
Users	Obtain information about Users and associated connections and devices
Issues	Obtain issue details, impacted hosts, and suggested actions for remediation
Site Design	Design/prepare NFP device to achieve building floor
Network Settings	Manage Network Settings
Software Image Management (SWIM)	Manage activation and distribution of software images
Device Onboarding (PnP)	Zero-touch deployment of network devices
Configuration Templates	Configure and manage CLI templates
SDA	(BETA) Configure and manage SDA related fabric border devices
Non-Fabric Wireless	Configure and manage SD-WAN, Wireless, and RF profiles in non-fabric wireless network
Command Runner	Retrieve real-time device configuration and CLI keywords
Network Discovery	Discover network devices and manage discovery jobs
Path Trace	Network route and flow analysis
File	Get configuration files by namespace and ID
Task	Get information about asynchronous tasks
Tag	Assign administrator-defined tags to network devices
Application Policy	Create and manage applications, application data, and application policies
Event Management	Event based notification to external handlers



Cisco DNA Center – Create (POST) - Update (PUT)

- Intent API PUT and POST methods require a JSON request payload.
- Correspond to the UPDATE function in CRUD.
- Both POST and PUT requests are handled within the Cisco DNA Center asynchronously, which means that the request to add, create, or modify a resource is initiated with a correctly formed request, but not necessarily completed before responding.
- A request, which has been successfully initiated, responds with a structure containing `executionId`, `executionStatusUrl`, and a status `message`.



Cisco ACI

- The **Cisco Application Centric Infrastructure (ACI)** platform that runs on Nexus 9000 hardware is the Cisco solution for Software-Defined Networking (SDN).
- The centralized management system is the Application Policy Infrastructure Controller (APIC), a cluster of controllers. With APIC, you get a unified operation of both physical and virtual, or software-based infrastructure.
- Instead of opening a subset of the network functionality through programmatic interfaces, the entire ACI infrastructure is opened up for programmatic access. This entirety is achieved by providing access to Cisco ACI object model.
- The ACI object model represents the complete configuration and run-time state of every software and hardware component in the entire infrastructure.
- Note: Rather than ACI mode, Nexus 9000 switches may be configured in NX-OS mode to access device-level APIs. In NX-OS mode you can manage switches as a Linux device.



Cisco ACI

■ ACI use cases

- Common use cases include:
 - Programmability as a single fabric, with access to read and write object models representing all attributes in the system.
 - Desired state defined and enforced.
 - Extension to AWS or Azure public clouds (via Multi-Site Orchestrator (MSO) and its API)

■ Tools used with ACI

- ACI toolkit, Cobra (Python), ACIrb (Ruby), Puppet, and Ansible
- Note: The above tools are available for accessing the Multi-Site Orchestrator (MSO) for hybrid clouds as well. MSO's API is different from ACI's, the latter is used to access a private ACI fabric via its Application Policy Interface Controller (APIC).

Cisco ACI

■ API Inspector

- When a task is performed in the Cisco APIC GUI, the GUI creates and sends internal API messages to the operating system to execute the task.
- By using the API Inspector, which is a built-in tool of the Cisco APIC, you can view and copy these API messages.
- An administrator can replicate these messages to automate key operations.

The screenshot shows the API Inspector window with the following content:

```

API Inspector
about:blank

Filters: ☒ trace ☒ debug ☒ info ☒ warn ☒ error ☒ fatal ☒ all
Search:  Reset ☐ Regex ☐ Match case ☐ Disable
Options: ☒ Log ☐ Wrap ☐ Newest at the top ☒ Scroll to latest Clear Close

method: GET
url: https://sandboxapicdc.cisco.com/api/node/class/infraWNode.json?query-target-f
response: {"totalCount":"1","subscriptionId":"72058229710192643","imdata":[{"infraW
timestamp: 15:40:19 DEBUG
method: GET
url: https://sandboxapicdc.cisco.com/api/node/class/fvTenant.json?query-target-filt
response: {"totalCount":"13","subscriptionId":"72058229710192642","imdata":[{"fvTen
timestamp: 15:40:19 DEBUG
method: GET
url: https://sandboxapicdc.cisco.com/api/node/class/aaaRbacRule.json?query-target-f
response: {"totalCount":"0","subscriptionId":"72058229710192644","imdata":[]}
timestamp: 15:40:23 DEBUG
method: GET
url: https://sandboxapicdc.cisco.com/api/aaaRefresh.json
response: {"totalCount":"1","imdata":[{"aaaLogin":{"attributes":{"token":"oQcAAAAA
timestamp: 15:40:25 DEBUG
method: GET
url: https://sandboxapicdc.cisco.com/api/node/mo/topology/HDfabricOverallHealth5min
response: {"totalCount":"1","imdata":[{"fabricOverallHealthHist5min":{"attributes":
timestamp: 15:40:36 DEBUG
method: GET
url: https://sandboxapicdc.cisco.com/api/node/mo/topology/HDfabricOverallHealth5min
response: {"totalCount":"1","imdata":[{"fabricOverallHealthHist5min":{"attributes":

```

Cisco ACI - Ansible Modules

Ansible modules for ACI (and MSO)

- Cisco and the wider community have collaborated on a broad suite of open source modules for Ansible. Thus, enabling configuration and management of ACI fabrics as code, alongside other Ansible-managed inventory.
- Modules have also been created to address multi-site and hybrid cloud resources via the Multi-Site Orchestrator (MSO) APIs.
- The ACI/MSO modules permit the simple creation of playbook elements to perform inquiry, administration, and management tasks upon an ACI fabric.
- For example, the task in the figure, drawn from Ansible documentation, idempotently ensures that a given tenant account exists (creating it, if not). It uses the `aci_tenant` module, which provides a range of tenant-management functionality.

```
- name: Ensure tenant customer-xyz exists
  aci_tenant:
    host: my-apic-1
    username: admin
    password: my-password
    tenant: customer-xyz
    description: Customer XYZ
    state: present
```



Cisco ACI - Ansible Modules

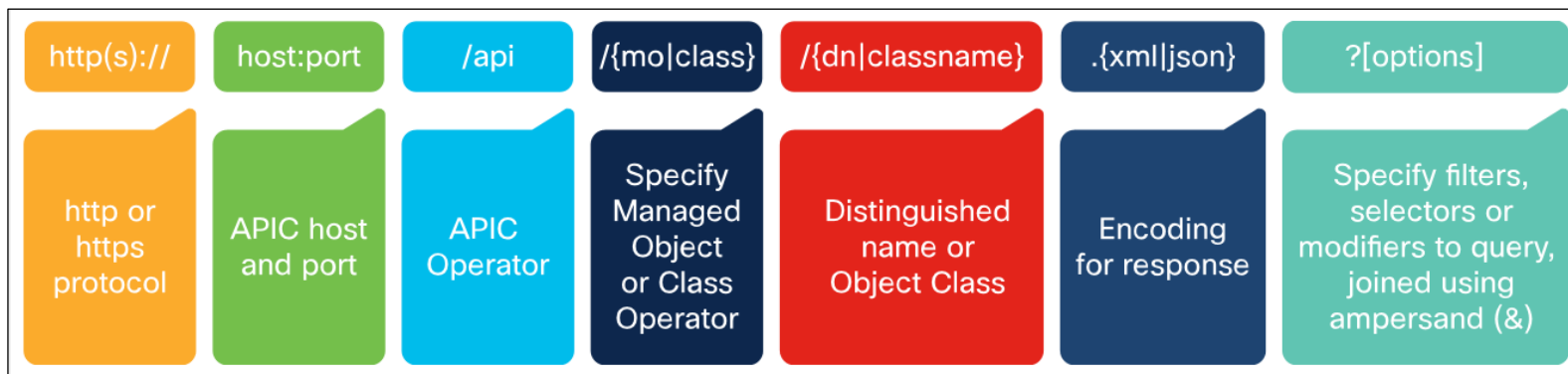
▪ Cisco ACI REST API Example

- The APIC CLI, GUI, and SDK use the same REST API interface so that whenever information is displayed and configuration changes are made, the data is read and written through the REST API, respectively.
- **Distinguished Name (DN)** – Identifies a specific target object, letting you map the name directly to URLs.
- **Relative Name (RN)** – Names the object apart from its siblings within the context of a parent object.
- Object instances are referred to as **Managed Objects (MOs)**.
- Every MO in the system can be identified by a unique DN.
- With the MO and its DN, you can refer to any object globally.
- In addition to a DN, you can refer to each object by its RN.
- The RN identifies an object relative to its parent object.



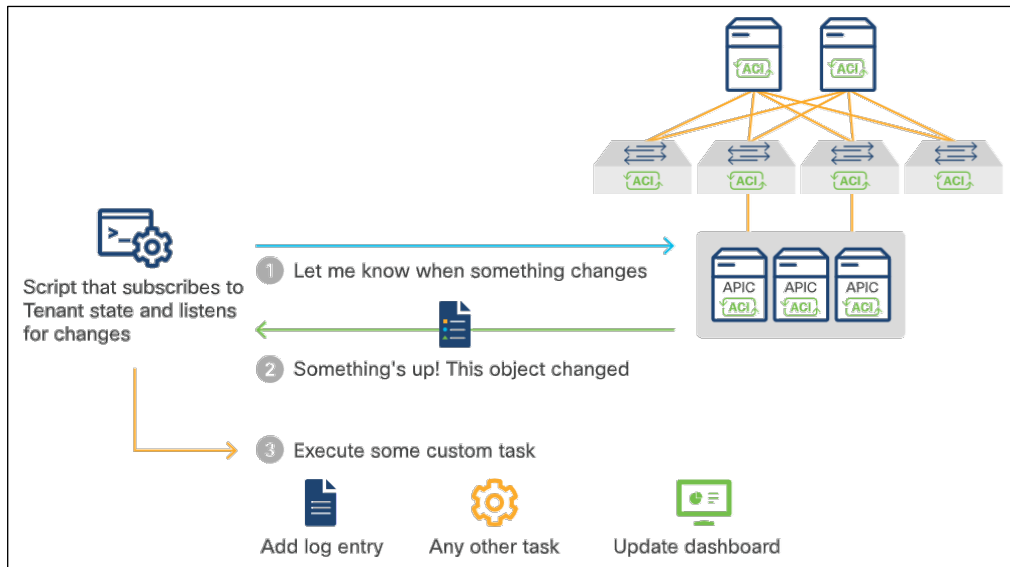
Cisco ACI URL Format

- A URI provides access to a target resource.
- The literal string **api**, indicates that the API is to be invoked.
- The final mandatory part of the request URI is the encoding format: either **.xml** or **.json**.
- The REST API supports a wide range of flexible filters.
- One of the most common use cases for this API is for monitoring the Cisco ACI Fabric.



Cisco ACI URL Format

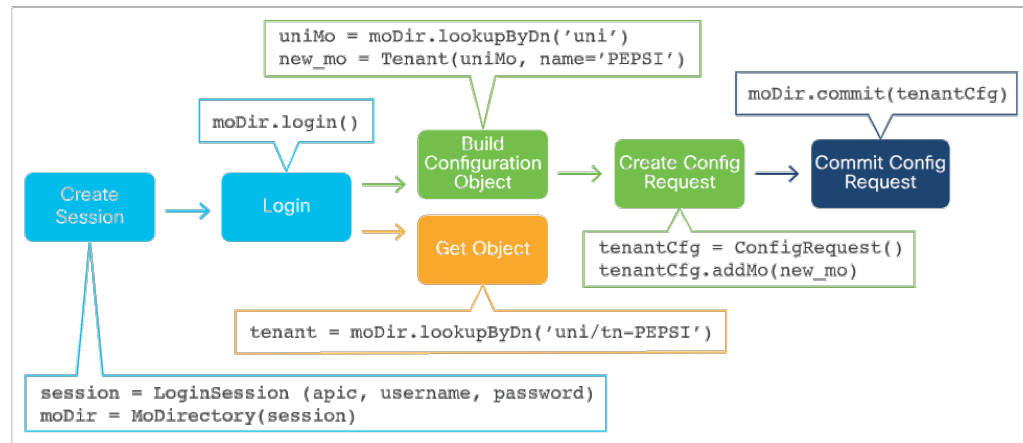
- Besides manual querying, you can gather information automatically and then apply policies, minimizing the opportunities for human error.
- When you start using automation when monitoring the ACI fabric, you can build various applications that can execute different tasks if there is a specific change in the network.





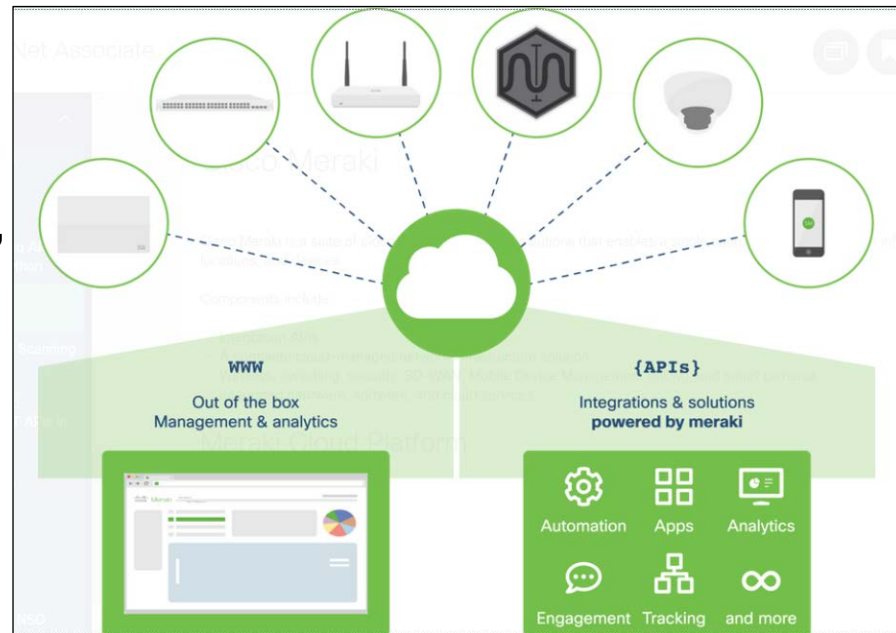
Cisco ACI Cobra

- To simplify application development with ACI, Cisco has developed Cobra, a robust Python library for the APIC REST API.
- Cobra SDK can be used to manipulate the MIT generally through the below workflow:
 - Create a Session Object.
 - Log in to Cisco APIC.
 - Create a Configuration Object.
 - Create a Configuration Request Object.
 - Add your Config Object to the Request.
 - Commit



Cisco Meraki

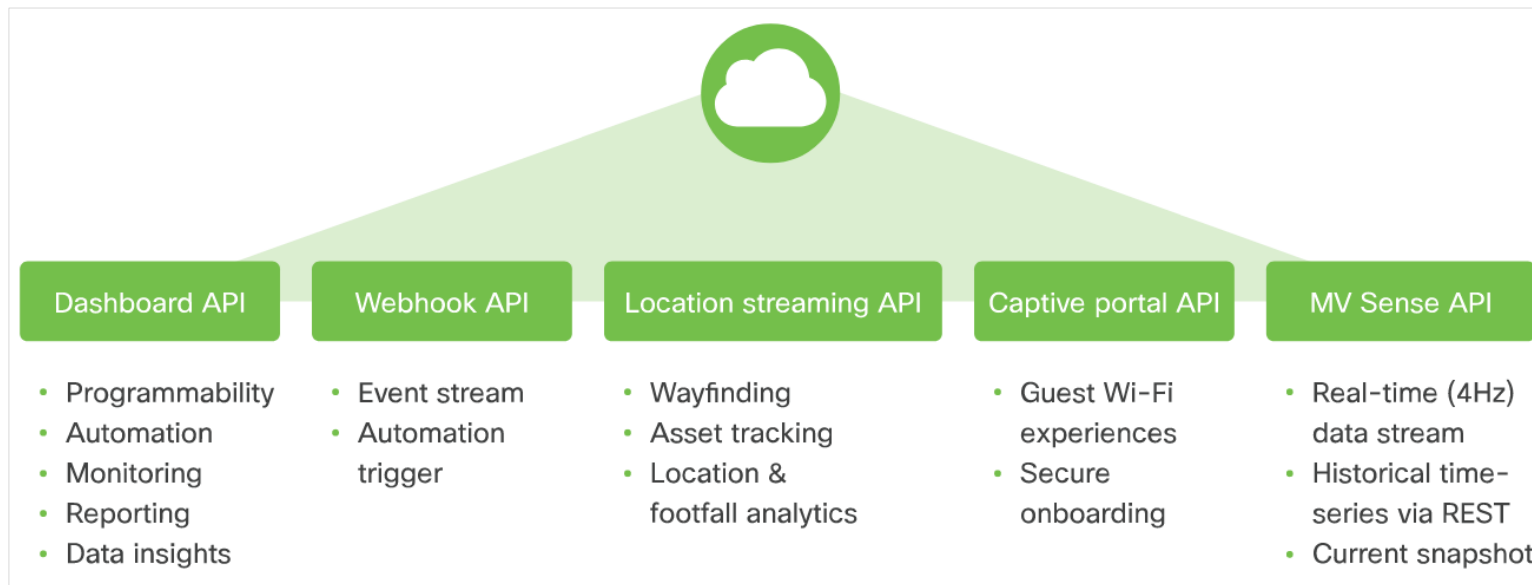
- Cisco Meraki is a suite of cloud-managed network solutions that enables a single source of management for infrastructure, locations, and devices.
- Components include:
 - Integration APIs
 - A complete cloud-managed network infrastructure solution
 - Wireless, switching, security, SD-WAN, Mobile Device Management (MDM), and smart cameras
 - Integrated hardware, software, and cloud services



Cisco Meraki

Meraki Integrations

- The Meraki enterprise cloud-managed networking infrastructure service has five different APIs for integration as seen below.



Cisco Meraki

Meraki Dashboard API

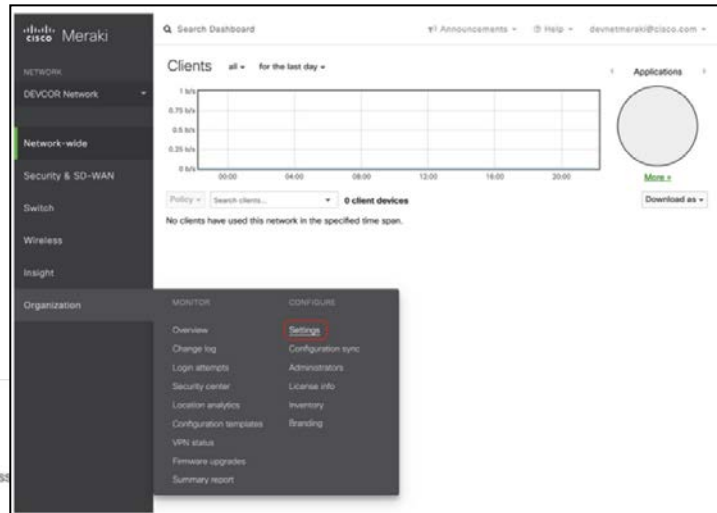
- The Cisco Meraki Dashboard API is a RESTful API that uses HTTPS for transport and JSON for object serialization.
- To provide access to the API for an organization, first enable the API for the organization under **Organization** > **Settings**.
- Scroll down.

Dashboard API access

API Access ⓘ

☒ Enable access to the Cisco Meraki Dashboard API

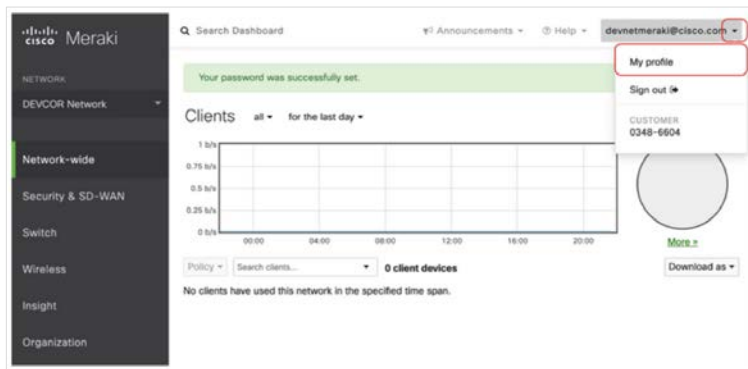
After enabling the API here, go to your [profile](#) to generate an API key. The API will return 404 for requests with a mis





Cisco Meraki

- After enabling the API, go to the **My Profile** page to generate an API key. The API key is associated with a Meraki Dashboard administrator account. **The API key needs to be kept safe**, as it provides authentication to all of the organizations that have the API enabled.



API access			
API keys			
Key	Created at	Last used	
*****2656	Dec 08 2019 08:34 UTC	Dec 08 2019 12:41 UTC	Revoke
*****272d	Dec 10 2019 06:08 UTC	Dec 10 2019 07:12 UTC	Revoke

- Note: The Meraki dashboard API will return a 404 code preventing the system from indicating the existence of resources to unauthorized users.**
- Note: These instructions are an example only. To complete these steps, you need to be a user with administrator access in a Meraki organization. The Meraki organization in this course has API access already and the API key is provided in the examples.



Cisco Meraki Location Scanning API

- **Location analytics**

- The Location Scanning API can be used by retail stores with multiple locations, conference deployments where location information can be useful for attendees, or deployments where the business wants to know trends in user engagement.

- **Bluetooth Scanning API**

- Meraki Access Points (APs) can detect and locate Bluetooth Low Energy (BLE) devices when they are nearby.

- **Location and privacy**

- Meraki Smart Cameras can perform object detection, classification, and direct tracking on the edge of the network, thus placing the computing needs in the endpoint.

- **Camera API categories**

- The available camera APIs are MV Sense, Live Link API, and Snapshot API



Cisco NX-OS Platform

■ What is Cisco NX-OS Platform?

- Nexus Operating System (NX-OS) is a data center operating system for the Nexus switch.
- Nexus switches are highly performant in data centers and integrate with a lot of systems.

■ Architecture

- Cisco Open NX-OS leverages the native Linux networking stack, instead of a custom-built userspace stack (NetStack) that was used in prior versions of NX-OS.
- Nexus switch interfaces, including physical, port-channel, vPC, VLAN, and other logical interfaces, are mapped to the kernel as standard Linux netdevs.

■ Environment and scale

- The Nexus family is performant and preferred by developers, DevOps professionals, and other users who want to provide self-service network infrastructure models.
- The Cisco Nexus 9000 Series NX-OS Verified Scalability Guide document describes the Cisco NX-OS configuration limits for Cisco Nexus 9000 Series switches.



Cisco NX-OS Platform

▪ **Container support**

- NX-OS supports running Linux Containers (LXC)s directly on the platform. It provides access to a CentOS 7-based Guest Shell, which supports custom functionality directly on the device in a secure, isolated shell.

▪ **Telemetry**

- You can integrate different telemetry applications such as Ganglia, Splunk, or Nagios on the switch with NX-OS.

▪ **Open NX-OS programmatic interfaces**

- Open NX-OS is the set of software used to provide the APIs, data models, and programmatic access including the NX-API REST service and model-driven programmability using YANG modeling.
- Open NX-OS includes the native NX-OS data model in the software image itself. If you are only using Open NX-OS, you do not have to download additional models.



Cisco NX-OS Platform

■ YANG, NETCONF, and RESTCONF

- Cisco NX-OS supports YANG models through the interfaces of NETCONF, RESTCONF on Open NX-OS.
- The Cisco NX-OS employs a NETCONF agent as a client-facing interface to provide secure transport for the client requests and server responses.

■ Enabling Model Driven Programmability features in NX-OS

- Enable the following features on the switch using CLI or another method (such as NX-API). Enable the transport protocols that you want to leverage.

```
feature bash-shell
feature netconf
feature restconf
```

- Note: These instructions apply to Open NX-OS 9.2.1+. Previous versions of Open NX-OS require manual installation and activation of RPMs for the protocol agents. See the Programming Guides for your platform if you are using an earlier version.



Cisco NX-OS Platform

■ OpenConfig

- OpenConfig models supported by Open NX-OS can be downloaded from Cisco Artifactory Open NX-OS Agents.
- Download the desired models to your local workstation, and then copy them to the Open NX-OS switch where you want to install them.

```
nx-osv9000-1#copy scp://developer@10.10.20.20/home/developer/Downloads/mtx-openconfig-all-1.0.0.0-9.2.1.lib32_n9000.rpm bootflash: vrf management
```

- Use the **bash-shell** feature on Open NX-OS to install the newly copied RPM files. Ensure that the feature **bash-shell** must be enabled on your switch.

```
nx-osv9000-1(config)#run bash sudo su
bash-4.2#
bash-4.2# cd /bootflash/
bash-4.2# yum install mtx-openconfig-all-1.0.0.0-9.2.1.lib32_n9000.rpm
```



Cisco NSO

- What is Cisco NSO?
 - **Network Services Orchestrator (NSO)** enables operators to adopt the service configuration solution dynamically, according to changes in the offered service portfolio.
 - NSO has three components:
 - Device and service abstraction layers
 - Configuration database
 - Model-driven programmable interface (YANG models)



Cisco NSO

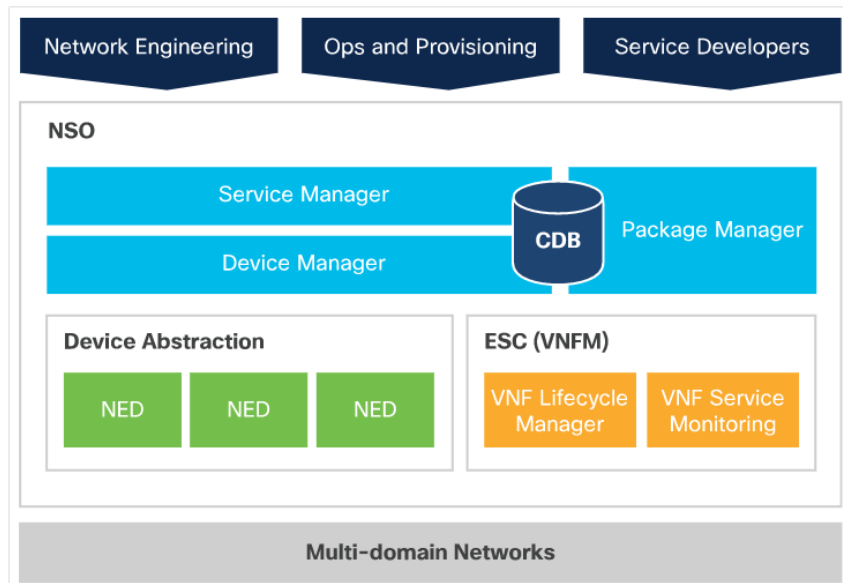
- **Device and service abstraction layers**

- NSO uses the standardized YANG modeling language to model and automate any type of device.
- The data model for a service correlates service definitions with network operations.
- For device data models, NSO recognizes and can work across the physical devices in a data center, including firewalls and other OSI model Layer 4 through Layer 7 devices.
- NSO can automate the launch, configuration, monitoring, and license management of **Virtual Network Functions (VNFs)**.

Cisco NSO

▪ Functional architecture

- The NSO architecture is logically comprised of two layers:
 - **Device Manager** – Simplifies device integration and manages different devices using a YANG and NETCONF view. Whether the interface is SNMP or CLI, the Device Manager creates a transactional change sequence for the target devices.
 - **Service Manager** – Lets you develop service-aware applications to configure devices. The Service Manager handles the complete lifecycle (creating, modifying, and deleting) of service instances.





Cisco NSO

■ Configuration Database

- NSO uses an internal Configuration Database (CDB) to store its own configuration, the configuration of all services, as well as a copy of the configuration of all managed devices.
- The NSO CDB provides:
 - A model on how to handle configuration data in network devices.
 - An internal API for locating network element configurations.
 - Automatic support for upgrade and downgrade of configuration data.
- CDB client applications need to be able to read configuration data from the database and react when configurations are updated.
- Note: The CDB journaling requires file system providing sufficient minimal performance and primitives must include file synchronization and truncation. NFS and other network file systems are unsuitable and unsupported for use with CDB in production deployment.



Cisco NSO

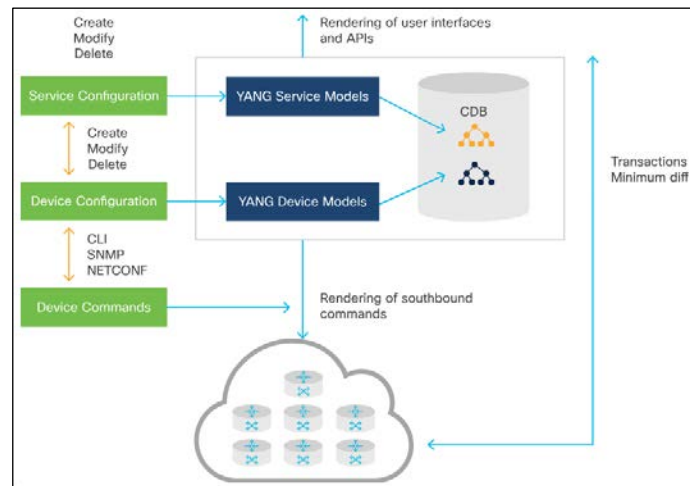
■ YANG models

- In YANG, data models are represented by definition hierarchies called schema trees. In NSO, there are three primary YANG sources:
 - **NSO data model** – This defines the built-in functions of NSO.
 - **Data models from devices** – Data models from devices such as native YANG modules from NETCONF devices, generated YANG modules from SNMP MIBs, or reverse engineered YANG modules from a CLI device. These YANG modules then specify the functions of the devices that are integrated to NSO.
 - **YANG service models** – When developing service applications, a developer specifies the service model, such as a BGP peer, firewall setting, or MPLS VPN, in YANG.

Cisco NSO

■ Service model design

- A Service in NSO consists of the following:
 - **A YANG service model** – This defines the attributes of the service.
 - **A device configuration mapping** – When the service is created, corresponding changes must be made to the devices. These are defined using either service templates or programmatically using Java.
- Note: While most of the validation can be expressed in YANG, in some cases, the configuration data validation will require external code, such as when performing look-ups in databases. This step is developed using MAAPI.





Cisco NSO Services

▪ What is Cisco NSO Service?

- An NSO service is a function provided by network devices.

▪ Northbound interfaces

- Allow integration of NSO with applications and portals.
- Available northbound interfaces: REST API, CLI, NETCONF, Web UI, SNMP, Java, and so on
- Note: The legacy REST API has been deprecated since NSO 5.1 and is scheduled to be removed in NSO 5.3.

▪ Command line interface (CLI)

- NSO CLI is a single interface for network devices and network services. It comes in two flavors: Juniper-style and Cisco XR-style.

▪ Web user interface

- The NSO Web UI is a YANG model browser with additional device and service functionality. The interface is built with pure client-side JavaScript.



Cisco NSO Services

- **Managing services (southbound)**

- NSO requires a YANG model, device address, management port, and authentication credentials for each device to be managed.
- A name identifies each managed device.

- **Southbound interfaces**

- Allow the configuration and management of network elements.
- Available Southbound interfaces are: NETCONF, SNMP, CLI, IOS, IOS XR, and so on.

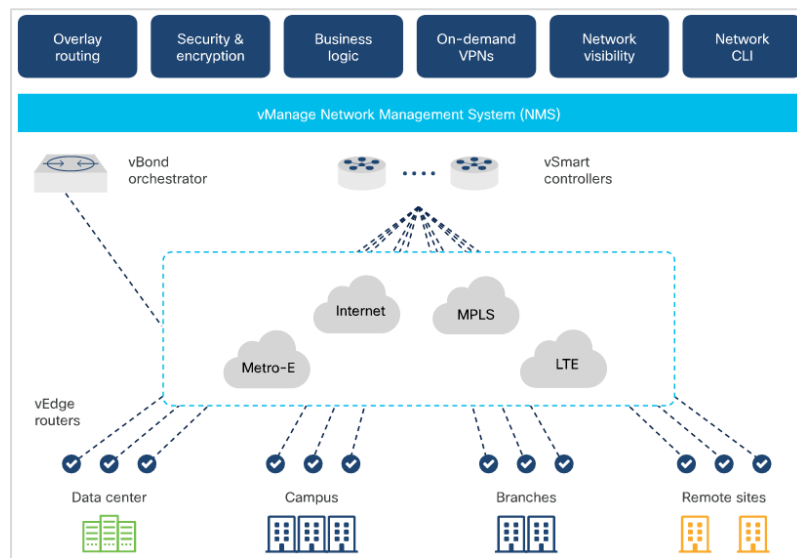
- **Adding devices**

- A new device can be added into NSO using any of the following methods:
 - Discovery
 - Manually
 - Cloning
 - Templates



Cisco SD WAN

- Through a dashboard called vManage, Cisco SD-WAN provides:
 - Transport independence** - Guaranteeing zero network downtime, Cisco SD-WAN automates application flexibility over multiple connections.
 - Network services** - Deliver rich networking and security services with clicks on the dashboard.
 - Endpoint flexibility** - Simplify connectivity across branches, campuses, data centers, or cloud environments, extending the SD-WAN fabric wherever you need it to go.





Cisco SD WAN

▪ Key features of Cisco SD-WAN

- Cloud-first architecture
- Embedded security
- Predictable application experience

▪ Cisco SD-WAN components

- **vManage Network Management System (NMS):** Centralized network management system, so that you can configure overlay networks from a dashboard.
- **vSmart Controller:** Controls the flow of data traffic by working with the vBond orchestrator to authenticate SD-WAN devices as they join the network.
- **vBond Orchestrator:** Orchestrates connectivity between vEdge routers and vSmart controllers.
- **vEdge Routers:** Provisioned at the perimeter of a site and delivered as hardware, software, cloud or virtualized components, vEdge Routers secure virtual overlay network over a mix of WAN transports.



Cisco SD WAN

▪ Cisco SD WAN APIs

- The Cisco SD-WAN software provides a REST API, which is a programmatic interface for controlling, configuring, and monitoring the devices in an overlay network.
- The Cisco SD-WAN REST API calls expose the functionality of the software and hardware features or of the normal operations you perform to maintain SD-WAN devices and the overlay network itself. Each of these features or operations is called a resource.
- Resources are grouped into collections. The SD-WAN REST API resources are grouped into the following collections:
 - Administration
 - Certificate Management
 - Configuration
 - Device Inventory
 - Monitoring
 - Real-Time Monitoring
 - Troubleshooting Tools



8.5 CISCO COMPUTE MANAGEMENT



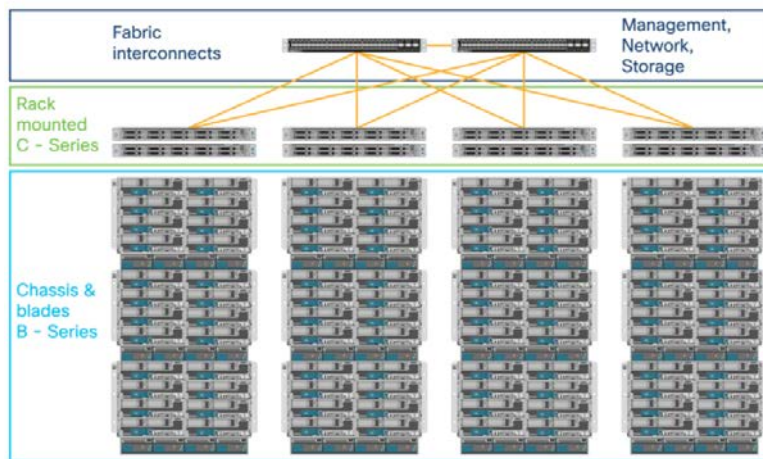


Cisco Compute Solutions

- The Cisco Unified Computing System (UCS), along with its software and SaaS adjuncts provides a complete physical and logical plant for compute, networking, and storage in the modern datacenter.
- **UCS tools and services:** UCS-Manager, UCS-Director, SaaS global infrastructure management system, and Intersight.
- UCS is 'hyperconverged' infrastructure.
- In very abstract terms, UCS virtualizes physical infrastructure and makes it uniformly software-definable.

Cisco UCS Manager

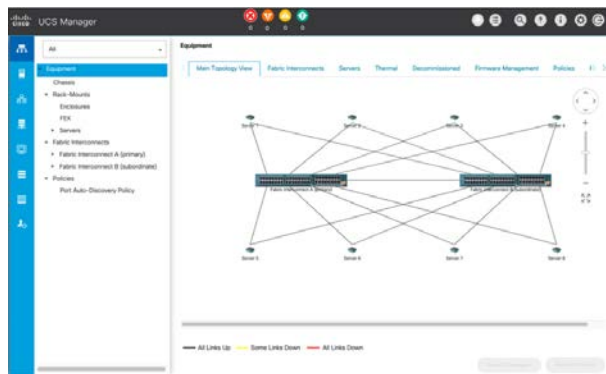
- The **Cisco Unified Computing System (UCS)** is a data center server computer product line composed of computing hardware, switching fabric, embedded management software, and virtualization support.
- The products provide scalability by integrating many components of a data center. This lets users manage them as a single unit through UCS Manager, UCS Central, and the Cisco Integrated Management Controller.





Cisco UCS Manager

- Cisco UCS Manager runs on the primary fabric interconnect and is assigned a Virtual IP address (VIP) with failover capability to the subordinate fabric interconnect.
- Cisco UCS Manager mediates all communication within the system
- Cisco UCS Manager is aware of the current configuration and performs automated device discovery whenever a new resource is installed.
- After a resource is detected, Cisco UCS Manager adds it and its characteristics to the system inventory.



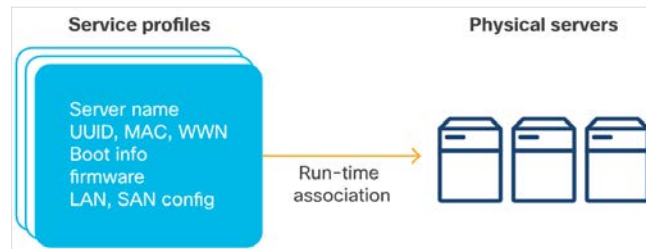
Cisco UCS Manager

■ Cisco UCS Service Profiles

- Cisco UCS Manager uses Service Profiles to assign a unique identity to a server when a Service Profile is associated with a server.
- Each UCS server can only have a single Service Profile association at a time.
- Service Profiles can be disassociated from one server and associated to another.

■ Applying abstraction to servers

- Cisco UCS Management software provides an abstraction layer between server component interfaces and the administrator.
- The abstraction layer is presented as a web-based GUI, an SSH CLI, and API.





Cisco UCS Manager Servers

■ Cisco UCS Manager servers

- Cisco UCS Manager servers are either blades that reside in chassis (B-Series Servers) or rack-mounted (C-Series servers). Both are connected to a redundant pair of switches called UCS Fabric Interconnects (FIs).
- When the servers in a UCS system are configured, there should not be a single point of failure.
- UCS Manager keeps track of events, alerts, errors, and statistics. It is able to report these items using various industry-standard management protocols like SNMP and Syslog.

■ Cisco UCS Unified API

- Cisco UCS Unified uses the same API methodology that is used for the CIMC, Cisco UCS Manager, and Cisco UCS Central.



Cisco UCS Manager Servers

■ Cisco UCS Management Information Model

- All the physical and logical components that comprise Cisco UCS are represented in a hierarchical Management Information Model (MIM), also referred to as the MIT.
- MIT is a tree structure with nodes, where each node in the tree represents a Managed Object (MO) or a group of objects that contains the nodes' administrative state and its operational state.
- The hierarchical structure starts at the top (**sys**) and contains the parent and child nodes.

```
Tree (topRoot):-----Distinguished Name:

|--sys----- (sys)
  |--chassis-1----- (sys/chassis-1)
  |--chassis-2----- (sys/chassis-2)
  |--chassis-3----- (sys/chassis-3)
  |--chassis-4----- (sys/chassis-4)
  |--chassis-5----- (sys/chassis-5)
    |--blade-1----- (sys/chassis-5/blade-1)
      |--adaptor-1----- (sys/chassis-5/blade-1/adaptor-1)
    |--blade-2----- (sys/chassis-5/blade-2)
      |--adaptor-1----- (sys/chassis-5/blade-2/adaptor-1)
      |--adaptor-2----- (sys/chassis-5/blade-2/adaptor-2)
    |--blade-3----- (sys/chassis-5/blade-3)
      |--adaptor-1----- (sys/chassis-5/blade-3/adaptor-1)
      |--adaptor-2----- (sys/chassis-5/blade-3/adaptor-2)
    |--blade-4----- (sys/chassis-5/blade-4)
      |--adaptor-1----- (sys/chassis-5/blade-4/adaptor-1)
    |--blade-5----- (sys/chassis-5/blade-5)
      |--adaptor-1----- (sys/chassis-5/blade-5/adaptor-1)
      |--adaptor-2----- (sys/chassis-5/blade-5/adaptor-2)
    |--blade-6----- (sys/chassis-5/blade-6)
      |--adaptor-1----- (sys/chassis-5/blade-6/adaptor-1)
    |--blade-7----- (sys/chassis-5/blade-7)
      |--adaptor-1----- (sys/chassis-5/blade-7/adaptor-1)
    |--blade-8----- (sys/chassis-5/blade-8)
      |--adaptor-1----- (sys/chassis-5/blade-8/adaptor-1)
```



Cisco UCS Managed Objects

■ What are Cisco UCS Managed Objects?

- Cisco UCS Managed Objects are XML representations of a physical and logical entities in the UCS system.
- A UCS managed object can have many attributes and can contain many children, the children objects can also contain many children, and so on.

■ Object naming

- A specific object can be identified by its Distinguished Name or by its Relative Name (RN).
- The DN has the following format consisting of a series of RNs: $dn = \{rn\}/\{rn\}/\{rn\}/\{rn\}$.
- The Relative Name (RN) identifies an object within the context of its parent object. The DN is composed of a sequence of RNs.

■ Object classes

- All managed objects belong to a class indicating the type of UCS resource the object represents.



Cisco UCS Managed Objects

▪ UCS XML API

- The Cisco UCS Manager XML API, like other APIs, provides methods to authenticate, query, and configure.

▪ Authentication methods

- Authentication methods authenticate and maintain an API session with UCS Manager:
 - **aaaLogin** - Initial method for logging in and retrieving an authentication cookie.
 - **aaaRefresh** - Refreshes the current authentication cookie.
 - **aaaLogout** - Exits the current session and deactivates the authentication cookie.



Cisco UCS Managed Objects

▪ Query methods

- Query methods obtain information on the current configuration state of an object. The following are query examples:
 - **configResolveDn** – Retrieves objects by DN.
 - **configResolveDns** – Retrieves objects by a set of DNs.
 - **configResolveClass** – Retrieves objects of a given class.
 - **configResolveClasses** – Retrieves objects of multiple classes.
 - **configFindDnsByClassId** – Retrieves the DNs of a specified class.
 - **configResolveChildren** – Retrieves the child objects of an object.
 - **configResolveParent** – Retrieves the parent object of an object.
 - **configScope—Performs** – Performs class queries on a DN in the MIT.



Cisco UCS Managed Objects

- Configuration methods
 - There are several methods to make configuration changes to managed objects.
 - These changes can be applied to the whole object tree, a subtree rooted at a specified object, or an individual object. The following are examples of configuration methods:
 - **configConfMo** - Affects a single managed object.
 - **configConfMos** - Affects multiple subtrees.
 - **configConfMoGroup** - Applies the same configuration changes to multiple subtree structures or managed objects.



Cisco UCS Managed Objects

■ Cisco UCS API documentation

- Cisco UCS API documentation is typically referred to as the UCS Object Model documentation.
- The Object Model documentation is available with the UCS Platform Emulator or online.
- Every UCS object class and method is listed in the documentation along with UCS types, events, faults, errors, and Syslog messages.

The screenshot displays the Cisco DevNet website's UCS Manager Information Model Reference for Release 4.0(2d). The left sidebar lists various classes under 'All Packages' and 'Classes'. The main content area shows the 'Class fabric:Vlan (CONCRETE)' with details such as Class ID:133, Encrypted: false, Exportable: true, Persistent: true, and Privileges: [admin, ext-lan-config, ext-lan-policy]. It also includes a description of the class as a user-created object representing a named Layer 2 bridge. Below this, the 'Naming Rules' section shows the RH FORMAT: net-[name] and the DN FORMAT: [0] fabric/eth-ctrls/[id]/net-[name].



Cisco UCS Manager Documentation

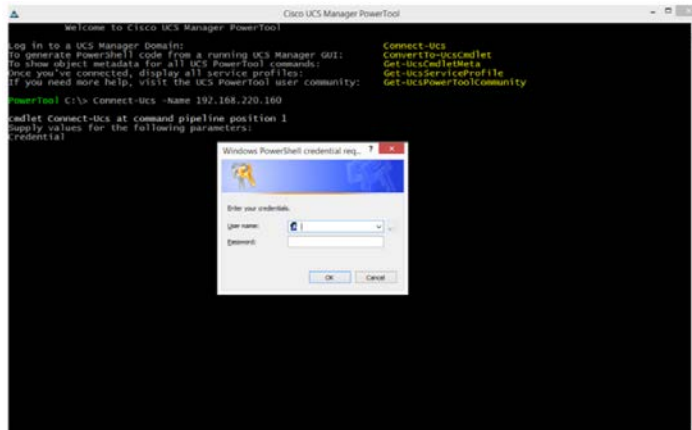
- The left navigation menu enables you to select an object.
- The right pane displays the object information divided into the following sections:
 - **Overview:** Indicates if the object is Abstract or Concrete.
 - **Naming Rules:** Indicate the object prefix.
 - **Containers Hierarchies:** Displays where the object can reside in the MIM.
 - **Contained Hierarchies:** Displays what objects the fabricVlan can contain and what objects those contained objects can contain, and so on.
 - **Inheritance:** Displays all the objects prior to the fabricVlan object that the fabricVlan inherited attributes from.
 - **Events, faults, and FSMs:** Objects that are attached to the parent object for which they are generated.
 - **Properties Summary:** Lists all the properties or attributes of the object.
 - **Properties Detail:** Each object property is defined completely in the Properties Detail.



Cisco UCS Power Tool

- UCS PowerTool is a library of PowerShell Cmdlets that enable the management of UCS environments from Microsoft Operating Systems, via the UCS XML API.
- Cmdlets ensure that they are completely aware of objects, their containment, properties and the details associated with each property.
- The Cmdlet to authenticate with a UCS Manager is as follows:

Connect-Ucs -Name <ip-address-ucs-manager>





Cisco UCS Power Tool

- The Cmdlet to query UCS Blades and view their DNs is:

Get-UcsBlade | Select-Object Dn

- The Cmdlet to create UCS VLAN 100 is:

Get-UcsLanCloud | Add-UcsVlan-Name vlan100 -Id 100

```

Cisco UCS Manager PowerShell
PowerTool C:\> Get-UcsBlade | Select-Object Dn
Dn
---
sys/chassis-3/blade-1
sys/chassis-3/blade-3
sys/chassis-3/blade-7
sys/chassis-4/blade-1
sys/chassis-4/blade-2
sys/chassis-5/blade-1
sys/chassis-5/blade-2
sys/chassis-5/blade-3
sys/chassis-5/blade-4
sys/chassis-5/blade-5
sys/chassis-6/blade-1
sys/chassis-7/blade-1
sys/chassis-7/blade-5
sys/chassis-7/blade-7
PowerTool C:\>
  
```



Cisco UCS Power Tool

- **UCS PowerTool OS support**

- UCS PowerTool is a library of PowerShell Cmdlets.
- UCS PowerTool has nearly 6000 Cmdlets to manage every aspect of Cisco UCS Systems.

- **UCS Python SDK**

- UCS Python SDK is a set of Python modules, each containing one or more classes, developed specifically to automate UCS Manager via the UCS XML API.
- Similar to UCS PowerTool, the UCS Python SDK provides hundreds of Python modules to interact with UCS objects.



Cisco UCS Power Tool

■ UCS Ansible

- UCS Ansible is available for UCS Manager and the Cisco Integrated Management Controller. UCS Ansible combines the UCS Manager authentication with the object mutation or object query.
- UCS Ansible modules are called as tasks in an Ansible playbook.
- Ansible playbook that queries a UCS Manager for UCS VLANs is shown in the figure.

```
---
- name: UCS Query
  hosts: ucs
  connection: local
  gather_facts: no
  tasks:
    - name: Query UCS
      ucs_query:
        hostname: "{{ ucs_hostname }}"
        username: "{{ ucs_username }}"
        password: "{{ ucs_password }}"
        class_ids: fabricVlan
      register: response
```



Cisco UCS Director

▪ What is Cisco UCS Director?

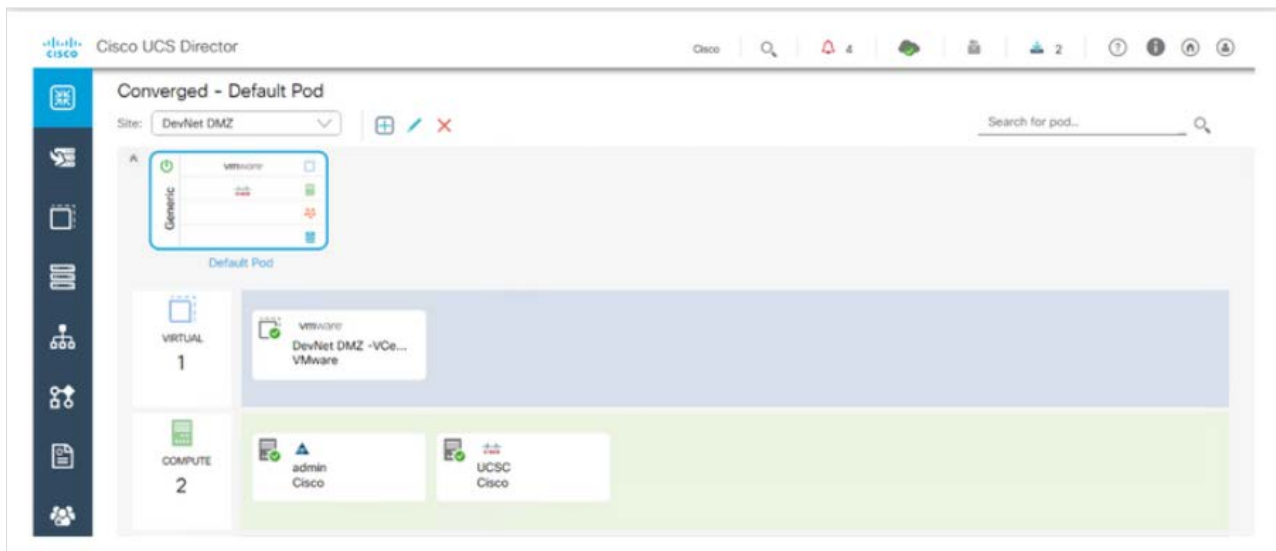
- Cisco UCS Director is a complete, highly secure, end-to-end management, orchestration, and automation solution for a wide array of Cisco and non-Cisco data center infrastructure components, and for converged infrastructure solutions based on the UCS and Cisco Nexus platforms.

▪ Management through Cisco UCS Director

- Cisco UCS Director extends the unification of computing and networking layers through Cisco UCS to provide visibility and management of data center infrastructure components.
- You can use Cisco UCS Director to configure, administer, and monitor supported Cisco and non-Cisco components.
- Network administrators can deploy and add capacity to converged infrastructures in a consistent and repeatable manner.
- Network administrators can create, clone, and deploy service profiles and templates for all Cisco UCS servers and compute applications.

Cisco UCS Director

- **Automation and orchestration with Cisco UCS Director**
 - Cisco UCS Director enables you to build workflows that provide automation services and to publish the workflows and extend their services to your users on demand.





Cisco UCS Director

- **Features and benefits:**

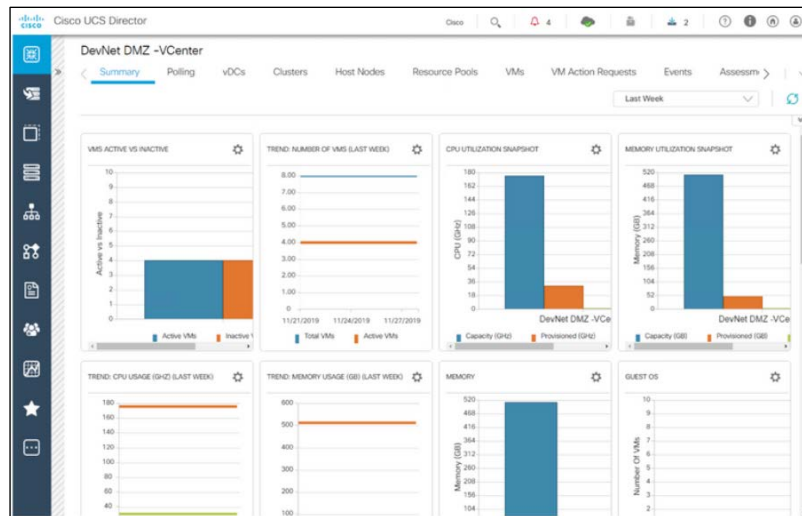
- **Central Management:** Provides a single interface for administrators.
- **Self-service Catalog:** Enables end users to order and deploy new infrastructure instances.
- **Adaptive Provisioning:** Provides a real-time available capability, internal policies, and application workload requirements.
- **Dynamic Capacity Management:** Provides continuous monitoring of infrastructure resources.
- **Multiple Hypervisor Support:** Supports VMware ESX, ESXi, Microsoft Hyper-V, and Red Hat hypervisors.
- **Computing Management:** Provisions, monitors, and manages physical, virtual, and bare metal servers, as well as blades.
- **Network Management:** Provides policy-based provisioning of physical and virtual switches and dynamic network topologies.
- **Storage Management:** Provides policy based provisioning and management of filers, virtual Filers (vFilers), Logical Unit Numbers (LUNs), and volumes.
- **Dashboards:** Provides multiple dashboards to track resource and policy utilization.



Cisco UCS Director

■ Model-based orchestration

- Cisco UCS Director includes a task library that contains over 1000 tasks and out-of-the-box workflows.
- Model-based orchestration and a workflow designer enable you to customize and automate the infrastructure administrative and operational tasks.





Cisco UCS Director – REST API

- To access the REST API through Cisco UCS Director, a valid Cisco UCS Director user account and an API access key are needed.
- Cisco UCS Director uses the API access key to authenticate API requests.
- This access key is a unique security access key code that is associated with a specific Cisco UCS Director user account.
- You must pass the REST API access key as an HTTP header in the following format:

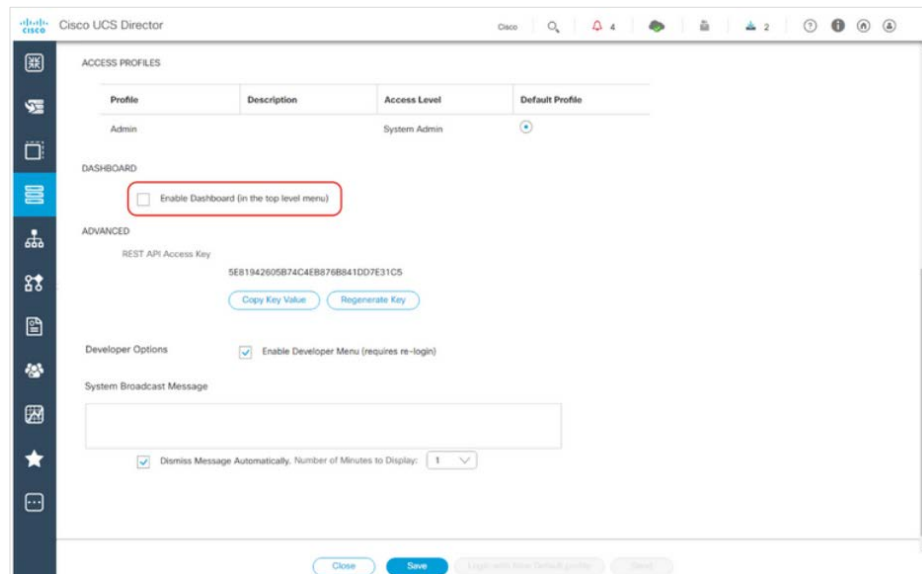
X-Cloupia-Request-Key: F90ZZF12345678ZZ90Z12ZZ3456FZ789

Cisco UCS Director – REST API

■ Using the GUI

- When you enable the developer menu, Cisco UCS Director GUI provides a developer menu option for developers to access the report metadata and REST API Browser. You can then access the following features:

- Report Metadata
- REST API Browser
- REST Client





Cisco UCS Director – REST API

▪ How to make a REST API request

- API clients use an HTTP request to interact with Cisco UCS Director.
- To pass the REST API access key, each request must be associated with an HTTP header called X-Cloupia-Request-Key with its value set to the current REST API access key.
- Requests made to the API have the following characteristics:
 - They are sent over HTTP.
 - Request format encoding can be either JSON or XML in UCS Director API Version 1.
 - The request must contain a valid URL.

▪ API VERSION 1

- <http://SERVER/app/api/rest?formatType=json&opName=operationName&opData=operationData>
- For details about encoding the URL, see the RFC at <http://www.ietf.org/rfc/rfc1738.txt>.

Cisco UCS Director – REST API

■ API VERSION 2

- Only XML is supported for Version 2 of the UCS Director API.

`http://server/cloupia/api-v2/group`

- HTTP method: POST

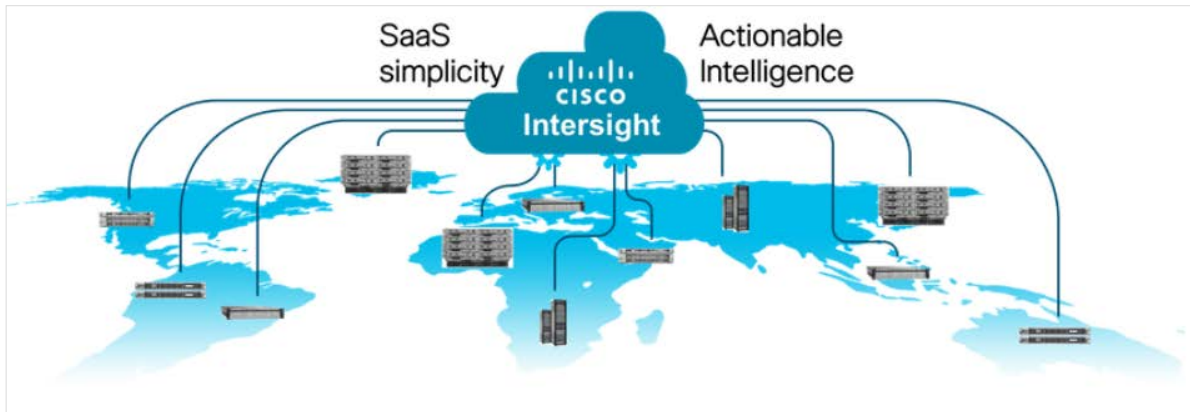
■ How to use cURL commands?

- cURL is a command line tool for getting or sending data using URL syntax.
- The cURL command is used to execute a REST API request.

```
<cuicOperationRequest>
  <payload>
    <![CDATA[
      <AddGroupConfig>
        <groupName>TestGroup</groupName>
        <groupDescription></groupDescription>
        <parentGroup>0</parentGroup>
        <groupCode></groupCode>
        <groupContact>jbesai@cisco.com</groupContact>
        <firstName></firstName>
        <lastName></lastName>
        <phone></phone>
        <address></address>
        <groupSharePolicyId></groupSharePolicyId>
        <allowPrivateUsers>>false</allowPrivateUsers>
      </AddGroupConfig>
    ]]>
  </payload>
</cuicOperationRequest>
```

Cisco Intersight

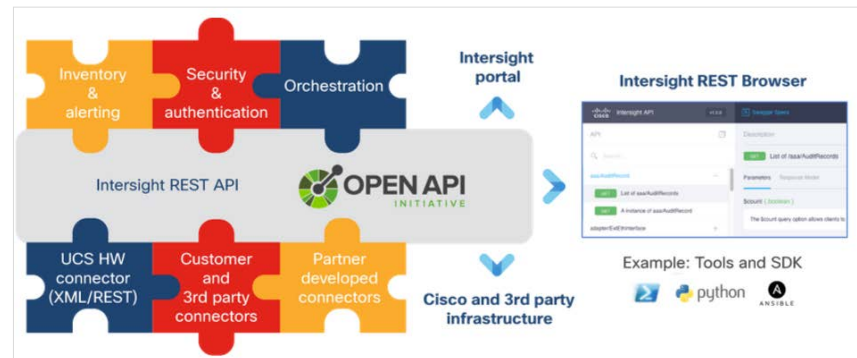
- Cisco Intersight is a **Software as a Service (SaaS)** systems management platform capable of managing infrastructure at the edge and remote locations as well as in the data center.
- The Intersight API is consistently available with a cloud-based management mode.
- New features can be added to the service without impacting existing automation systems.





Cisco Intersight

- **Intersight and Cisco Unified Computing System (UCS)**
 - Cisco UCS Manager software provides an abstraction layer between the server component interfaces and the administrator.
 - The abstraction layer is presented as a web-based graphical user interface (GUI) and an application programming interface (API).
 - Cisco Intersight builds on the UCS Unified Fabric and UCS Management experience with a Cisco-hosted and maintained management platform
 - With the Intersight API you can build integrations with Cisco Intersight for additional tasks like monitoring, analysis, configuration, deployment, and orchestration.





Cisco Intersight

■ Intersight API Capabilities

- The Intersight API provides access to the Intersight MIM.
- The Intersight API accepts and returns messages that are encapsulated through JavaScript Object Notation (JSON) documents and uses HTTP over TLS as the transport protocol.
- All the physical and logical components visible in Cisco Intersight are represented in a hierarchical MIM, also referred to as the Management Information Tree (MIT).

■ Accessing the Intersight API

- There are a couple of ways to gain access to the Intersight API:
 - Use a web browser as an Intersight API REST Client
 - Use API keys for remote or service access



Cisco Intersight

■ Ansible Modules for Cisco Intersight

- Intersight supports several API integrations, including Ansible modules that enable inventory collection and configuration management of Intersight resources.
- Ansible is written in Python, and the modules for Intersight interact with the API using the code written in Python.
- Several example playbooks and Lab guides are hosted on GitHub.

■ Additional SDKs and resources

- In addition to Ansible Modules, Intersight provides Python SDK and PowerShell modules that are generated from Intersight OpenAPI schema (which is also referred to as the Intersight Swagger Spec).
- The intersight.com site hosts the latest OpenAPI specification and links to the SDKs.



8.6 CISCO COLLABORATION PLATFORMS





Introduction

- **Cisco Unified Communications Manager (Unified CM)**
 - Used to configure and automate device provisioning, call routing, and profiles and settings management in a single solution.
 - Deployed in hospitals, banks, universities, and government agencies to manage the increasing number of devices and user profiles.
- **Contact Center** - It provides robust customer support for call centers with agent desktop, supervisor, and reporting capabilities.
- **Finesse** - It is an agent and supervisor desktop.
- **Webex** - It encompasses Meetings, Teams, and Devices.
 - Webex Teams is a meetings and messaging app designed to improve collaboration.
 - Webex Devices include digital whiteboards, telepresence units, and room controls.



Cisco Unified Communications Manager

- **What is UCM?**

- **Cisco Unified Communications Manager**, also known as Unified CM, CUCM, or CallManager is:
 - an IP-based communications solution to support mobile and remote workers in small, medium, or enterprise-level businesses.
 - an integration for voice, video, and data into a single on-premise solution for call control and session management.
 - highly extensible, with various APIs for configuration, management, monitoring, and call control.



Cisco Unified Communications Manager

▪ Purpose

- The primary function of Unified CM is to manage phone users, IP phones, directory numbers, and to connect and manage calls to the desired destinations. With Unified CM you can:
 - Define gateways, trunks, remote destinations, and more telephony-related information.
 - Configure a full range of call handling tasks, including hold, transfer, call forward, and starting conference calls.



Cisco Unified Communications Manager

■ Unified CM Advanced Features

• Extensions

- Unified CM integrates with other services:
 - Cisco Instant Messaging and Presence (Cisco IM&P)
 - Voicemail
 - Contact Center

Unified CM Advanced Features		
AXL	UDS	Serviceability
Platform Administrative Web Services (PAWS)	Softphone compatibility	Extension Mobility
Cisco Jabber Voice & Video SDK	Java Telephony API (JTAPI) or Telephony API (TAPI)	Client Matter Codes (CMC)
Forced Authentication Codes	Hunt Lists	Music/Video on Hold



Administrative XML Layer

- **Administrative XML Layer (AXL)** is an XML/SOAP-based interface that provides a mechanism for inserting, retrieving, updating, and removing data from the Unified Communication configuration database.
- Developers can use AXL and the provided Web Services Description Language (WSDL) to create, read, update, and delete objects such as gateways, users, devices, route-patterns and so on.
- The AXL interface provisions and manages objects in the Unified Communication Management Administration Console.
- The AXL API is for administration and configuration.

Application	Application
API	AXL (Administrative XML)
Protocol	SOAP
Encoding	XML
Transport	HTTP(S)
Application server	Apache/Tomcat
Data storage	Database



Administrative XML Layer

▪ How it works?

- AXL is a SOAP interface.
- The methods for the objects begin with: **list**, **add**, **update**, **get**, and **remove**.
- For example, **listPhone** lists all phones, and **addUser** creates a new user.

▪ SQL queries

- You can also perform direct SQL queries to update or retrieve data in the Unified CM configuration database using **ExecuteSQLupdate** or **ExecuteSQLquery**.

▪ Versioning

- The AXL schema version is backwards-compatible for up to two major releases.
- This means developers can use AXL schema version 10.0(1) for Unified CM release 10.0(1) through 12.5(1).



Administrative XML Layer

▪ **Advanced features**

- One of the most advanced feature of AXL is the Change Notification Feature.
- You can use this SOAP request repeatedly to see what changes have been made to the system since the last time you ran the request.
- You can request to see changes about specific categories, like Phone or User, or just see all changes.

▪ **Related SOAP-based Administration APIs**

- UC Manager Serviceability includes similar SOAP-based API requests for retrieving information about phones such as the registration status, the IP address, and so on.
- It also includes a performance monitoring API, and an API to manage and get the status of CUCM services.



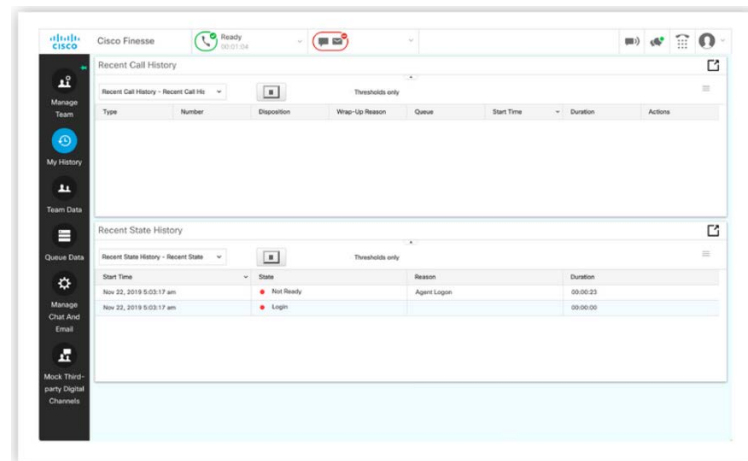
User Data Services

- **User Data Services (UDS)** is a REST-based API that provides a mechanism for inserting, retrieving, updating and removing data from the Unified Communication configuration database.
- **Purpose**
 - The UDS API is designed for end users to configure settings.
 - What you can do with UDS
 - UDS API can be used to create a directory search or manage user preferences and settings in the web application.
- **How it works**
 - UDS is a REST-based interface that sends and receives XML-formatted data.
 - UDS implements the four common HTTP request methods: **GET**, **POST**, **PUT**, and **DELETE**.



Cisco Finesse

- **Finesse** is Cisco's browser-based contact center agent and supervisor desktop.
- Finesse has REST APIs and JavaScript APIs that can be used to:
 - Build a script to automate tasks
 - Build custom agent desktops for the agent state workflow
 - Integrate contact center functionality into applications
 - Integrate applications into the Finesse agent and supervisor desktop
 - Integrate into existing applications to add contact center functionality
 - Embed existing web pages into a custom gadget



Cisco Finesse

■ What is a Contact Center?

- A contact center, also known as a call center, is typically a centralized location where a company handles the customer service for their business. A Customer service can be provided in the form of calls, text, email, and chat.
- Contact centers have been categorized into two tasks:
 - **Inbound tasks:** When customers initiate communication with customer service.
 - **Outbound tasks:** A customer interaction made from the contact center agent to a customer.





Cisco Finesse

▪ **Contact center systems**

- Contact center systems are very complex.
- Contact center systems perform skill-based routing using an Automatic Call Distributor (ACD)
- Contact center systems also provide agent and supervisor management.

▪ **Finesse deployments**

- Finesse has two different deployments:
 - Standalone Finesse used with a Contact Center Enterprise system
 - Co-resident Finesse used with a Unified Contact Center Express.
- Finesse communicates with the contact center system via the CTI protocol.



Cisco Finesse

■ Finesse APIs

- It provides an extensive list of REST and JavaScript APIs for developers to integrate Finesse's functionality into applications or applications to be integrated into the Finesse agent desktop.

■ Finesse REST APIs

- Finesse provides REST APIs for performing agent and supervisor actions programmatically.
- Because the APIs are HTTP-based, they can be used in both thick and thin applications.
- The Finesse REST APIs that use the GET verb are synchronous, whereas the remaining are asynchronous.

Webex Teams

■ What is Webex Teams?

- Cisco Webex Teams is an online collaboration solution to connect people and teams through chat, voice, and video.
- With the Webex Teams app, you gain access to secure virtual work spaces.
- Teams also integrates with Cisco Webex devices.

■ Features of Webex Teams

- Create spaces
- Create meetings
- Work inside and outside the company the same way
- Place calls





Webex Teams

▪ Security

- Users and sensitive information is kept safe with extensive controls to help you configure and control your security policies.

▪ Integrations

- Webex Teams includes pre-built solutions with third-party applications from vendors such as Microsoft, Google Cloud, and Salesforce.

▪ Features of the Webex Teams API

- The Webex Teams API is an extensive set of APIs that allow you to interact with the entire Webex Teams platform, including managing requests with pagination and providing error reporting.
- The capabilities include managing organizations, teams, people, rooms, memberships, and messages to creating conversational bots or embedded video calls



Webex Teams

Features of Webex Teams API	Description
Organizations	Represent a set of people in Webex Teams. Organizations may manage other organizations or be managed themselves.
Teams	Groups of people with a set of rooms visible to all members of that team. Teams API resources are teams to be managed, created, updated, and deleted.
People	Registered users of Webex Teams.
Rooms	Virtual meeting places where people post messages and collaborate to get work done. The Rooms API can manage, create, update, and delete rooms.
Memberships	Represent a person's relationship to a room. The Memberships API resources allow to list members of any room , create or revoke memberships, and update memberships to make someone a moderator of a room.
Team Memberships	Represent a person's relationship with a team.
Messages	Way of communication in a room. Use this API to list, create, and delete messages.



Webex Teams

- API methods can be used to manage, create, update, and delete different features of WebEx Teams.

Methods of Webex Teams API	Description	Example
GET	List and show Details	https://api.ciscospark.com/v1/teams
POST	Create and Send	https://api.ciscospark.com/v1/teams
PUT	Update	https://api.ciscospark.com/v1/teams/{teamId}
DELETE	Delete	https://api.ciscospark.com/v1/teams/{teamId}




Webex Devices

- Cisco Webex Devices provide access to all Webex's features.
- Webex Boards, Room Devices, and Desk Devices enable collaboration through video, calling, and programmability.

Meet Webex Devices.


Get the most out of Cisco Webex Meetings and Cisco Webex Teams with tools designed for better team collaboration.



Cisco Webex Board
All-in-one whiteboard, wireless presentation screen, and video conferencing system for smarter team collaboration.



Cisco Webex Room Devices
Intelligent video conferencing devices for meeting rooms of all sizes.



Cisco Webex Desk Devices
Simple-to-use and compact video conferencing devices designed for desktops.

Webex Devices

■ Webex Board Series

- All-in-one team collaboration devices for meeting rooms and spaces.

■ Webex Room Series

- Brings integrated video conferencing systems to every room.

■ Webex Desk Series

- Brings high-quality video conferencing to your desktop.

■ Cisco Touch 10

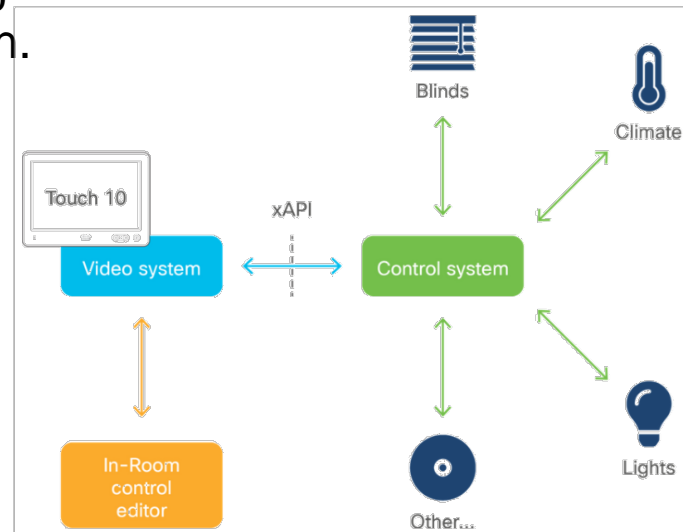
- Intuitive touchscreen device for interacting with Cisco conferencing systems.



Webex Devices

■ Programmability for Cisco Collaboration Devices:

- Webex Devices can be customized through the API, known as the xAPI. This enables bi-directional communication with third-party applications and control systems.
- There are multiple ways to access xAPI including Telnet/SSH, HTTP, and RS-232 serial connection.
- **xAPI supports both XML and JSON and JavaScript Macros for on-device customization.**



Webex Devices

- **UI Extensions (In-Room Controls) and macros**
 - UI Extensions describe custom widgets, buttons, and other virtual controllers that you can create and deploy. These used to be globally, referred to as "In-Room Controls".
 - Going forward there is only one tool UI Extensions Editor and In-Room Controls will be a subset of UI Extensions.

In-Room Control

You can customize our user interface to allow control of peripherals in a meeting room, for example lights and blinds. This allows for the powerful combination of an external control system's functionality and the Cisco user interface.

Create/Edit Interfaces

Launch Editor	Launch the editor to create custom user interface panels. Export them to the video system to make them appear in the user interface, or save the designs in local files for later.
Download Editor	Download the editor to create interface designs offline, for export to the video system later.
Launch Simulator	Launch a room simulator in your browser.

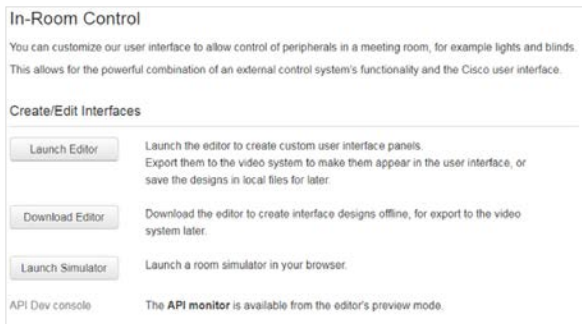
API Dev console

The **API monitor** is available from the editor's preview mode.

Webex Devices

■ What UI Extensions do

- UI Extensions enable you to add custom user interface elements to the Touch 10 display.
- Launching the simulator from “UI Extensions Editor” will load a virtual meeting room, equipped with several automated systems controlled from switches on the walls.
- The switch controls let you switch the projector on or off, interact with the projector canvas, close/open the blinds, and so on.

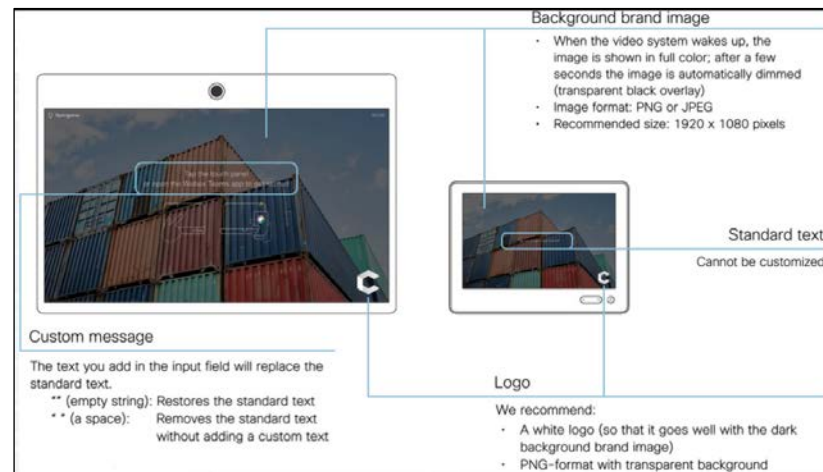




Webex Devices

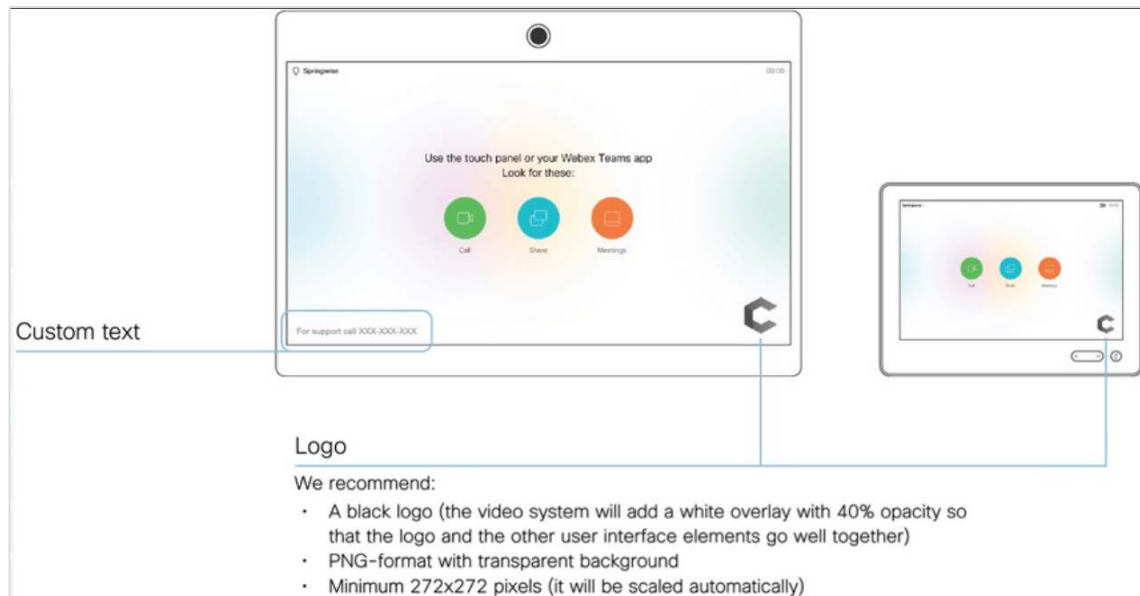
■ Personalizing Collaboration Devices

- As of version 9.2 of Cisco's Collaboration Endpoint software release, on-screen branding, signage and message customization options let you personalize the appearance of a room device and its Touch10 interface.
- Branding and customization of the room device "Halfwake" screen lets you:
- Add a background image
- Add a small logo image to the bottom right corner
- Customize or remove the default on-screen welcome message



Webex Devices

- In the 'Awake' state, you can:
 - Add a small logo image to the bottom right corner (screen/Touch10)
 - Add a label or message to the bottom left corner (screen only)





8.7 CISCO SECURITY PLATFORMS





Introduction

- To secure a network or an application, you need automated ways to identify threats and mitigate risks, as well as ways to configure systems with security in mind.
- Scripts can ingest data faster than a human.
- Cisco provides a large portfolio of security technologies and product families which are configurable and manageable via APIs. These mainly include:
 - Advanced Malware Protection (AMP) for Endpoints
 - Cisco Firepower Management Center (FMC)
 - Cisco Firepower Threat Defense (FTD)
 - Cisco Identity Services Engine (ISE)
 - Cisco Threat Grid
 - Cisco Umbrella



Cisco Advanced Malware Protection (AMP)

- **What is AMP?**

- Cisco Advanced Malware Protection (AMP) for Endpoints provides API access to automate security workflows and includes advanced sandboxing capabilities to inspect any file that looks like malware in a safe and isolated way.

- **Architecture**

- AMP has a collection of subscription-based products that are managed with a centralized web-based console. You can deploy AMP on mobile phones and email or web servers.

- **Integrations**

- AMP integrates across the Cisco security portfolio with deployment options such as Cisco Umbrella and Meraki MX.

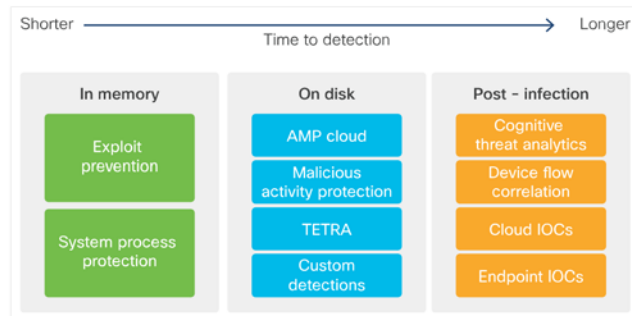
- **Environment and scale**

- AMP for Endpoints can be used in a university campus setting, within healthcare organizations, for government entities, or industrial and manufacturing environments.

Cisco Advanced Malware Protection (AMP)

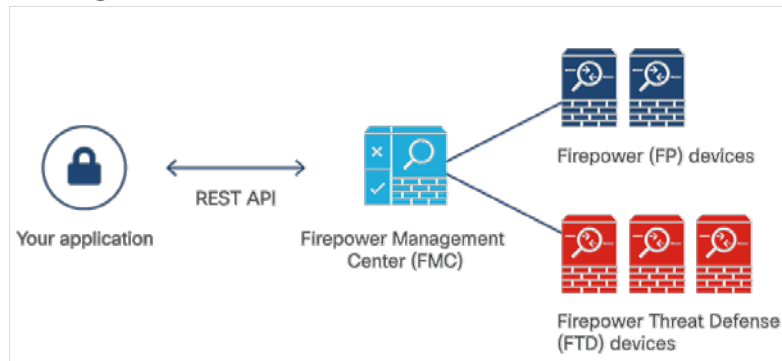
■ Capabilities

- There are three main categories of capabilities that AMP offers:
 - **Prevention:** AMP protects against identified threats in malware files by preventing breaches.
 - **Detection:** AMP continuously monitors and records all file activity to detect malware.
 - **Responses and automation:** AMP accelerates investigations and automatically remediates malware across PCs, Macs, Linux, servers, and mobile devices (Android and iOS).



Cisco Firepower Products

- Firepower Management Center (FMC) is a central management console that can configure all aspects of Firepower Threat Defense (FTD) including access control rules and policy object configuration.
- With FMC, devices on the network can be managed for controlling and filtering the network traffic based on various characteristics.
- The FMC API and FTD APIs cannot directly co-exist and hence you have to choose either:
 - Firepower Device Manager (FDM)/FTD-API/CDO or FMC





Cisco Firepower Products

■ Protection

- Firepower Threat Defense configuration with Firepower Device Manager also provides protective services that are listed here:
 - Track, backup, and protect CA Certificates.
 - Manage, backup, encrypt, and protect private keys.
 - Internet Key Exchange (IKE) key management.
 - Using security intelligence data to filter traffic, including IP addresses, address blocks, domain names, and URLs
 - Provide Access Control Lists to select traffic for services (controls which websites are available to the users on the network).
 - Two types of ACL can be configured: **Extended** and **Standard**

■ Architecture

- FMC can run on VMware vSphere or Amazon Web Services (AWS) and also on a range of physical devices like Cisco FMC 1000, 1600, 2000, 2500, 2600, 4000, 4500, and 4600.
- Firepower management tools are purpose-built for customers to manage and configure their Firepower Threat Defense devices. These tools include FMC and FDM.



Cisco Firepower Products

▪ Integrations

- Firepower Management Center (and Device Manager, though currently with a limited feature set) can integrate with Cisco Identity Services Engine (ISE).
- The integration of ISE with FMC enables movement of particular users in or out of quarantine after they start a VPN session.
- Other products, such as Threat Grid and Umbrella, can integrate with Firepower Threat Defense devices.

▪ Environment and scale

- Within FTD, FDM and Next Generation Firewall APIs help small and medium-sized businesses so that they do not have to hire security experts.

▪ API authentication

- For the FMC API, you use an access token to authenticate to the REST API. The token lasts for 30 minutes before the client must refresh it.
- To make the call, you use the header `X-auth-access-token:<authentication token value>`

Cisco Identity Services Engine (ISE)

■ What is ISE?

- The **Cisco Identity Services Engine (ISE)** is an integral part of the Cisco security portfolio.
- ISE provides a rule-based engine for enabling policy-based network access to users and devices. It automatically and securely places the device and user into the right part of the network.

■ Architecture

- Cisco ISE architecture consists of nodes with defined node types: Administration, Policy Service, Monitoring, or pxGrid.
- The remaining pieces of the architecture include network resources and endpoints, or devices connecting to the network.





Cisco Identity Services Engine (ISE)

■ Integrations

- Cisco ISE integrates with identity systems for identity management including role-based access control (RBAC), Okta/SAML Single-Sign On (SSO), Lightweight Directory Access Protocol (LDAP), Active Directory (AD).

■ Capabilities and use cases

- ISE's capabilities can be summarized as follows:
 - Asset visibility
 - Policy compliance
 - Secure wired access
 - Segmentation

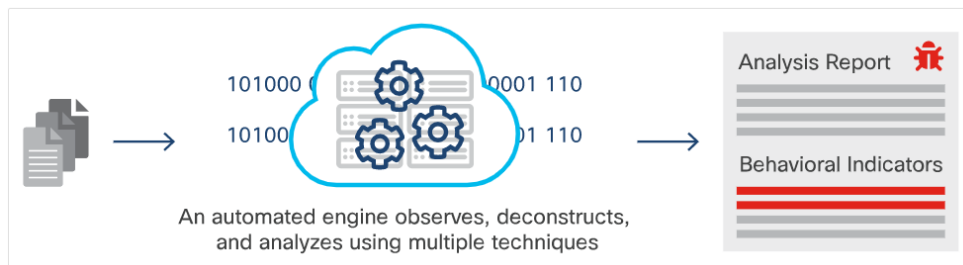
Cisco Threat Grid

■ What is Cisco Threat Grid?

- Threat Grid is a malware analysis platform that combines static and dynamic malware analysis with threat intelligence from global sources.

■ Benefits and Purpose

- Review and analyze potential threats or behavior indicators of malware activity.
- The user interface and API workflows are designed for Security Operations Center (SOC) analysts, malware analysts, security specialists, and forensic investigators.





Cisco Threat Grid

■ Integrations

- Threat Grid is also available through integrations with other Cisco Security products, such as Advanced Malware Protection, next-generation firewalls, and Cisco ASA with FirePOWER Services.

■ Environment and scale

- The content subscription license has the capacity to sample and analyze between 500 and 10,000 samples per day.

■ Capabilities

- Threat Grid offers malware analysis capabilities:
 - **Static analysis** – provides identifying information about the file, file headers, and its contents.
 - **Dynamic analysis** – executes the malware in a safe, specialized virtual environment called a "glovebox".



Cisco Umbrella

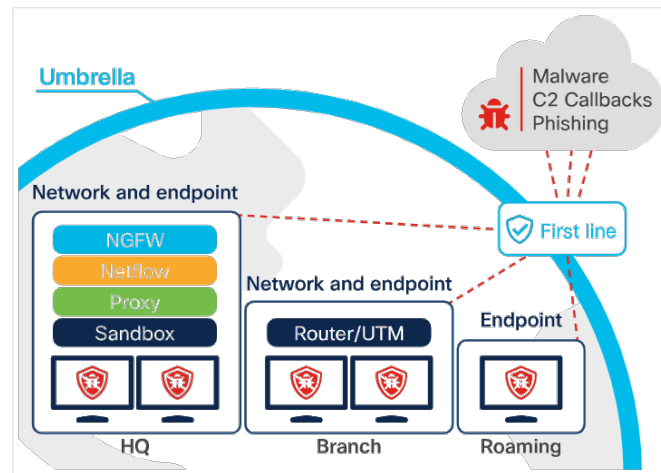
What is Cisco Umbrella?

- Umbrella uses Domain Name Servers (DNS) to enforce security on the network.
- Umbrella blocks access to malicious domains, URLs, IPs, and files.

Architecture

- APIs and data-driven architecture provide the base of the Umbrella service. There are a few APIs to use with Cisco Umbrella:

- Enforcement API
- Network Devices API
- Investigate API
- Reporting API





Cisco Umbrella

▪ Integrations

- You can integrate Meraki MR and Umbrella for wireless protection use cases.
- Use the Umbrella Enforcement API to take actions on a domain request, or use the Umbrella Investigate API to pull threat intelligence data programmatically.

▪ Environment and scale

- Umbrella is used in larger retailers, large hospital settings, and university campuses. It can protect hundreds, thousands, or tens of thousands of endpoints.

▪ Capabilities

- Umbrella's protection capabilities include:
 - Wi-Fi protection when guests are on your network
 - Selected application blocking
 - Endpoint security for off-network (not on VPN) devices
 - Web filtering



8.8 CISCO PLATFORMS AND DEVELOPMENT SUMMARY





What Did I Learn in this Module?

- To sort the Cisco developer offerings, DevNet creates Dev Centers for each technology group.
- Cisco Dev Centers include: Unified Communications Manager, Cloud, Collaboration, Data center, Internet of Things(IoT) and edge computing, Networking, Security, Wireless and mobile, and Application developers.
- Software Development Kit (SDK) contains a set of software development tools integrated for developing applications for a specific device or system.
- Model-driven programmability inherits the power of models. There are two types of YANG models: open and native.
- Network Configuration (NETCONF) is a protocol designed to install, manipulate, and delete the configuration of network devices.
- Network automation is used for various common tasks in an organization.



What Did I Learn in this Module?

- Internetwork Operating System (IOS) was original operating system and IOS-XE is the next-generation programmable platform.
- Cisco DNA Center is a foundational controller and analytics platform for large and midsize organizations providing a single dashboard for network management, network automation, network assurance, monitoring, analytics, and security.
- The Cisco Unified Computing System (UCS), along with its software and SaaS adjuncts, provides a complete physical and logical plant for compute, networking, and storage in the modern datacenter.
- Cisco Intersight is a Software as a Service (SaaS) systems management platform capable of managing infrastructure at the edge and remote locations as well as in the data center.



What Did I Learn in this Module?

- Cisco's suite of on-premise and cloud-based collaboration solutions includes Unified Communications Manager, Contact Center, Finesse, and Webex. Cisco Unified Communications Manager is also known as Unified CM, CUCM or CallManager.
- AXL is an XML/SOAP-based interface that provides a mechanism for inserting, retrieving, updating, and removing data from the Unified Communication configuration database.
- UDS is a REST-based API that provides a mechanism for inserting, retrieving, updating and removing data from the Unified Communication configuration database.



What Did I Learn in this Module?

- Finesse is Cisco's browser-based contact center agent and supervisor desktop.
- Cisco Webex Teams is an online collaboration solution to connect people and teams through chat, voice, and video.
- Cisco provides a large portfolio of security technologies and product families which are configurable and manageable via APIs.

