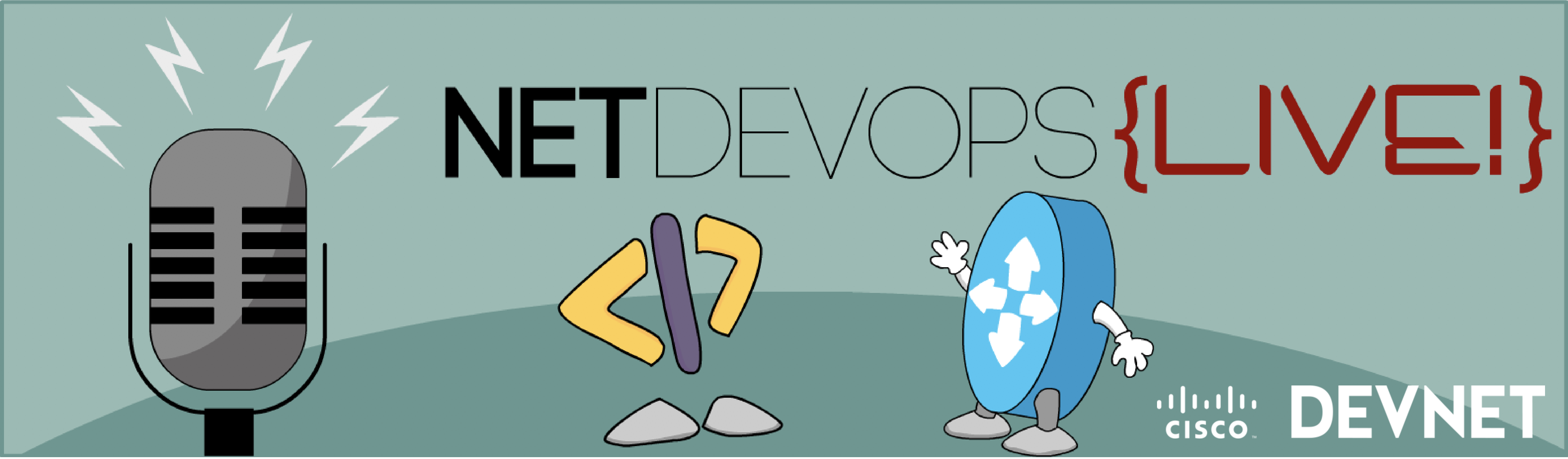


Exploring the DNA Center platform Northbound API

Adam Radford

Distinguished System Engineer

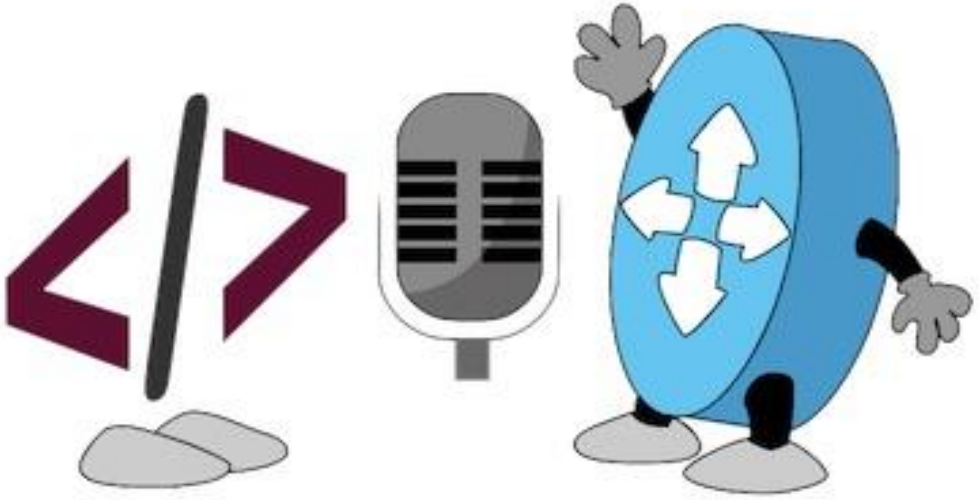
@adamradford123

Season 1, Talk 10

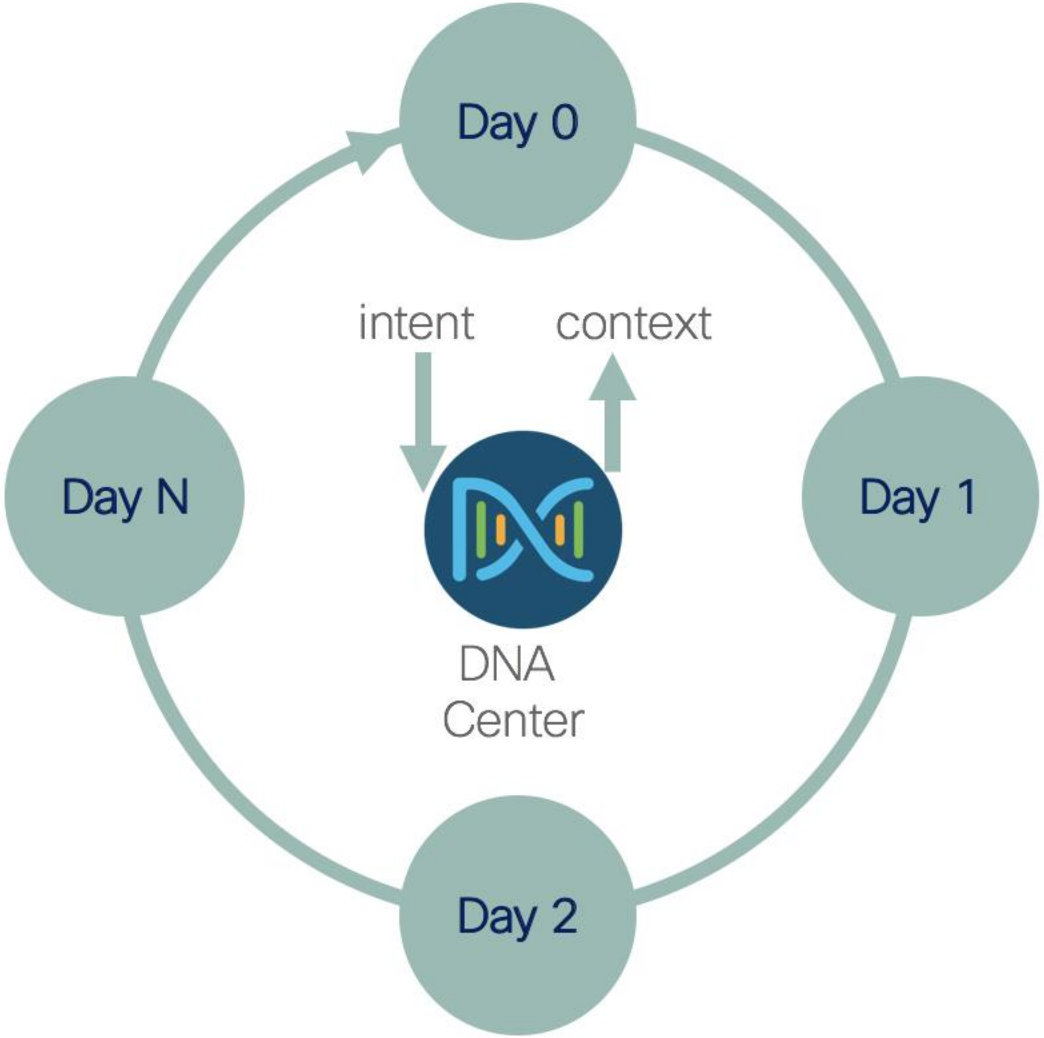
https://developer.cisco.com/netdevops/live

What are we going to talk about?

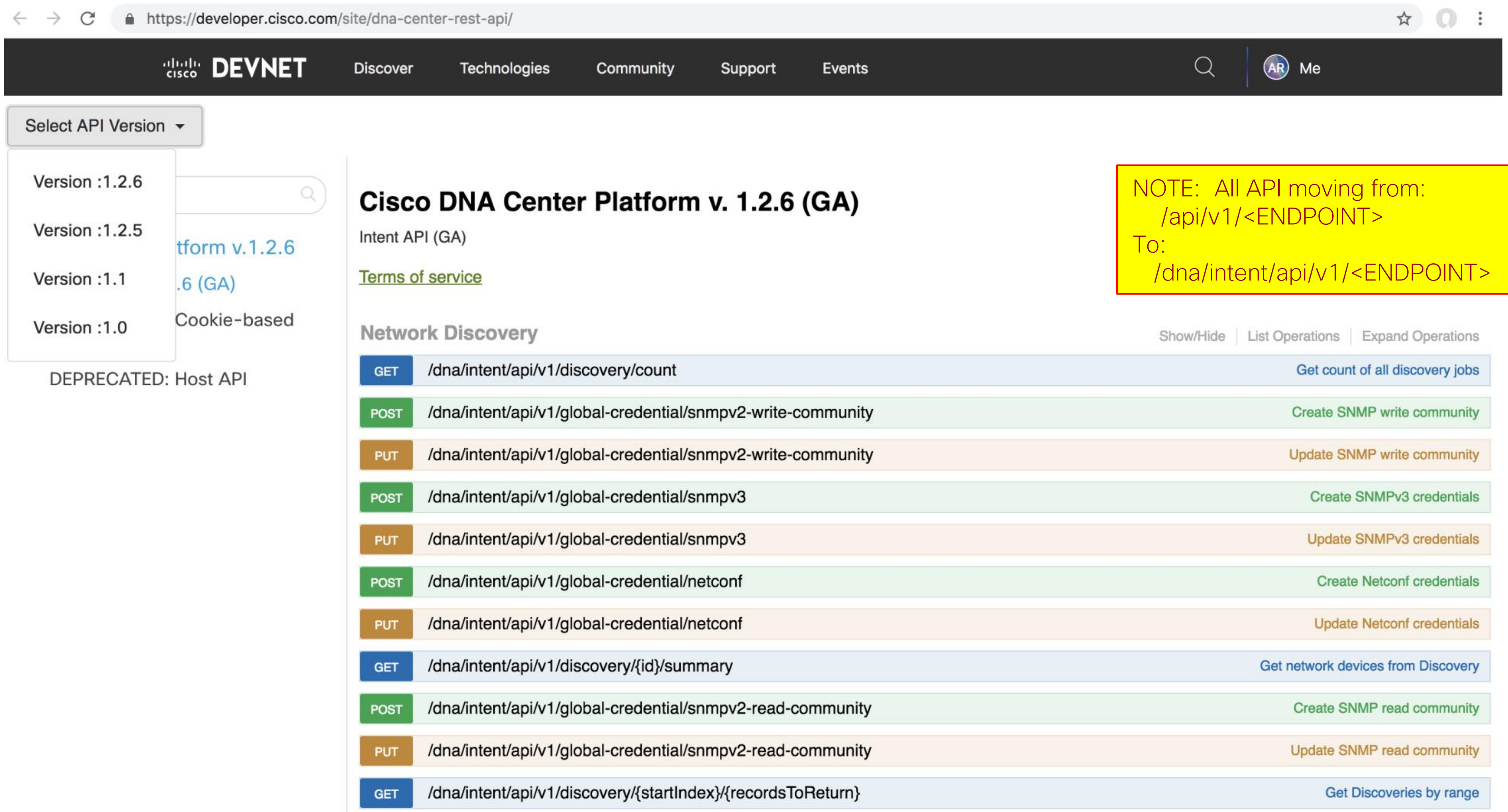
- DNA Center
  - Assuming some level of familiarity
- DNA Center Platform API
- Examples:
  - Lifecycle
  - Webhook



DNA Center API



https://developer.cisco.com/site/dna-center-rest-api/



https://developer.cisco.com/site/dna-center-rest-api/

Cisco DNA Center Platform v. 1.2.6 (GA)

Intent API (GA)

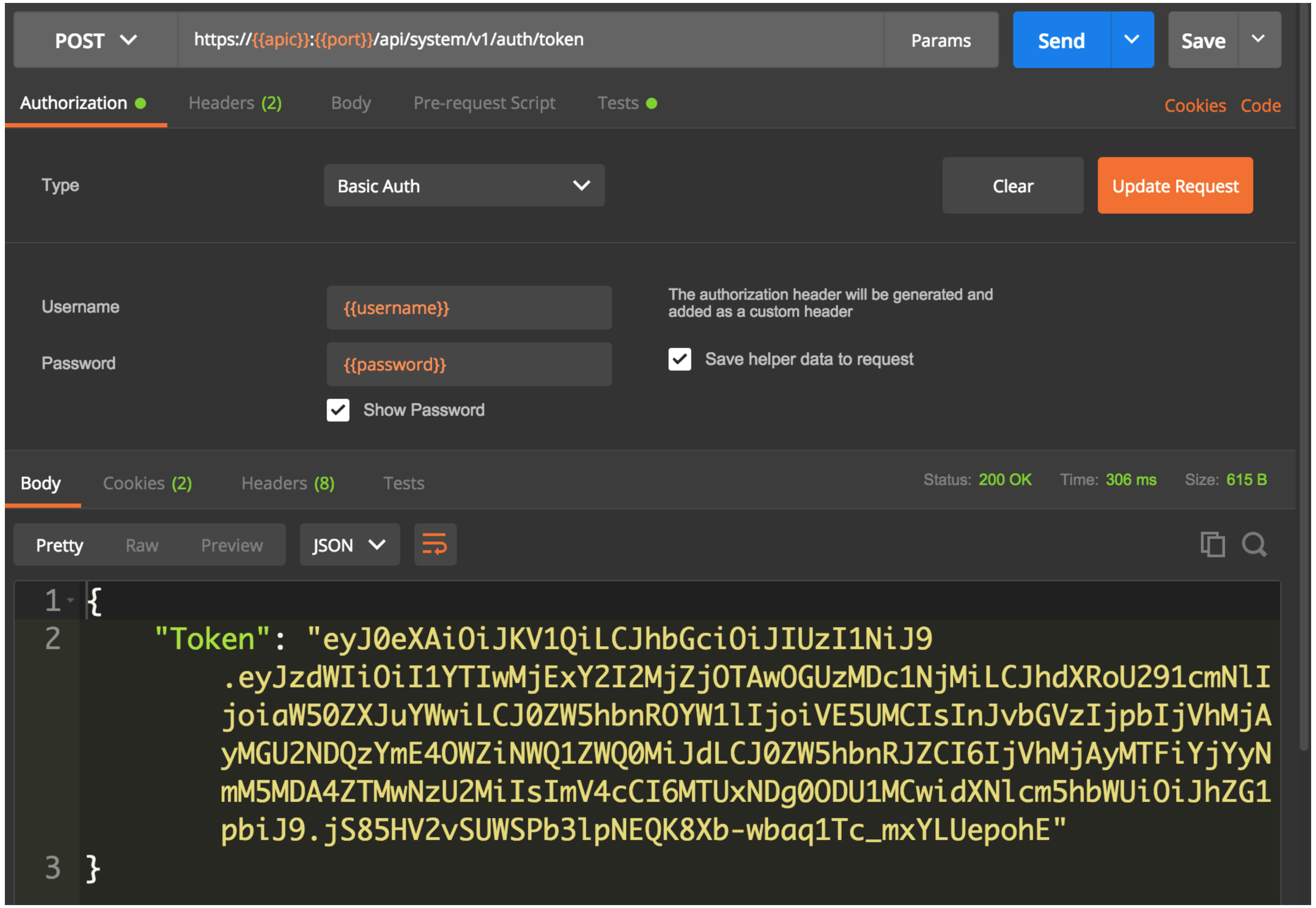
Terms of service

Network Discovery

GET	/dna/intent/api/v1/discovery/count	Get count of all discovery jobs
POST	/dna/intent/api/v1/global-credential/snmpv2-write-community	Create SNMP write community
PUT	/dna/intent/api/v1/global-credential/snmpv2-write-community	Update SNMP write community
POST	/dna/intent/api/v1/global-credential/snmpv3	Create SNMPv3 credentials
PUT	/dna/intent/api/v1/global-credential/snmpv3	Update SNMPv3 credentials
POST	/dna/intent/api/v1/global-credential/netconf	Create Netconf credentials
PUT	/dna/intent/api/v1/global-credential/netconf	Update Netconf credentials
GET	/dna/intent/api/v1/discovery/{id}/summary	Get network devices from Discovery
POST	/dna/intent/api/v1/global-credential/snmpv2-read-community	Create SNMP read community
PUT	/dna/intent/api/v1/global-credential/snmpv2-read-community	Update SNMP read community
GET	/dna/intent/api/v1/discovery/{startIndex}/{recordsToReturn}	Get Discoveries by range

NOTE: All API moving from:
/api/v1/<ENDPOINT>
To:
/dna/intent/api/v1/<ENDPOINT>

Authentication request - POSTMAN



https://{{apic}}:{{port}}/api/system/v1/auth/token

Type: Basic Auth

Username: {{username}}

Password: {{password}}

☑ Show Password

The authorization header will be generated and added as a custom header

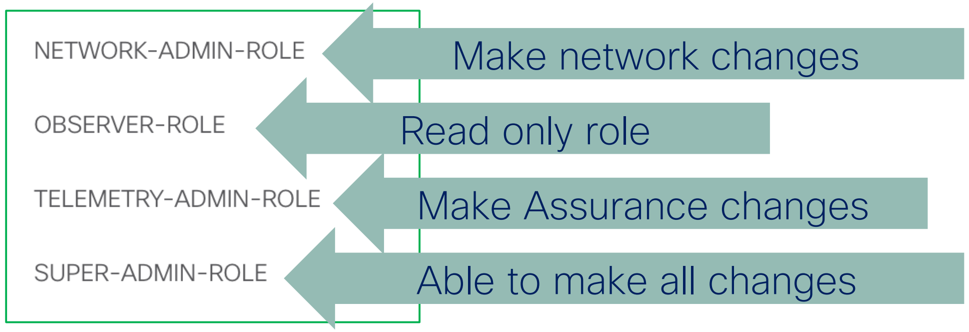
☑ Save helper data to request

Status: 200 OK Time: 306 ms Size: 615 B

```
{
    "Token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9
        .eyJzdWIiOiI1YTIwMjExY2I2MjZjOTAwOGUzMDc1NjMiLCJhdXRoU291cmNlI
        joiaW50ZXJuYWwiLCJ0ZW5hbnROYW1lIjoiVE5UMCIsInJvbGVzIjpbIjVhMjA
        yMGU2NDQzYmE4OWZiNWQ1ZWQ0MiJdLCJ0ZW5hbnRJZCI6IjVhMjAyMTFiYjYyN
        mM5MDA4ZTMwNzU2MiIsImV4cCI6MTUxNDg0ODU1MCwidXNlcm5hbWUiOiJhZG1
        pbiJ9.jS85HV2vSUWSPb3lpNEQK8Xb-wbaq1Tc_mxYLUepohE"
}
```

	APIC-EM	DNAC
Authentication request	POST JSON Body	Basic Auth
Response	["response"]["serviceTicket"]	["Token"]

Roles:

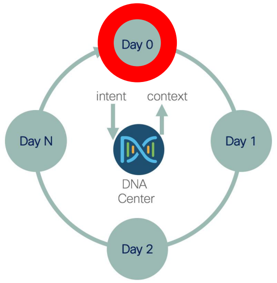


NETWORK-ADMIN-ROLE — Make network changes

OBSERVER-ROLE — Read only role

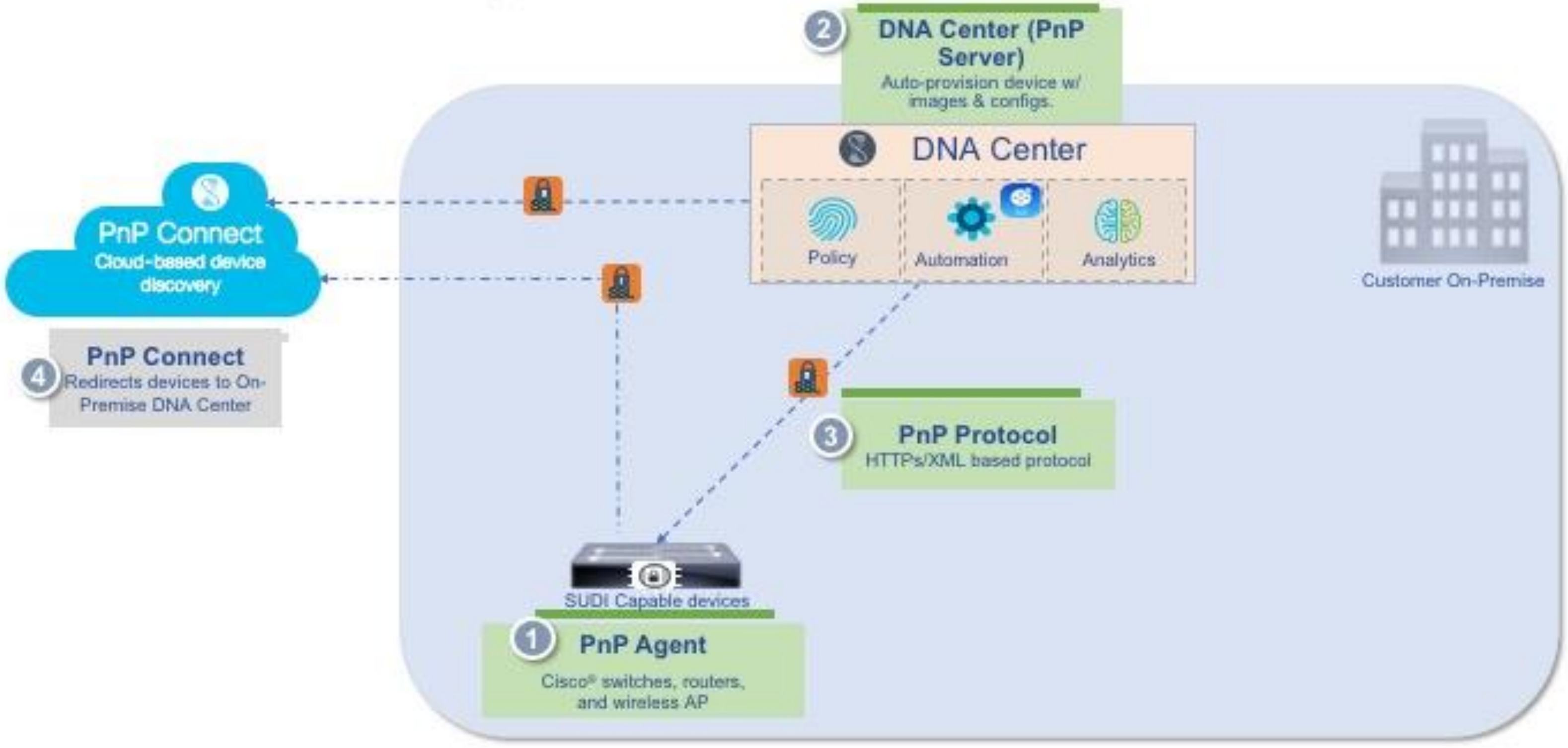
TELEMETRY-ADMIN-ROLE — Make Assurance changes

SUPER-ADMIN-ROLE — Able to make all changes

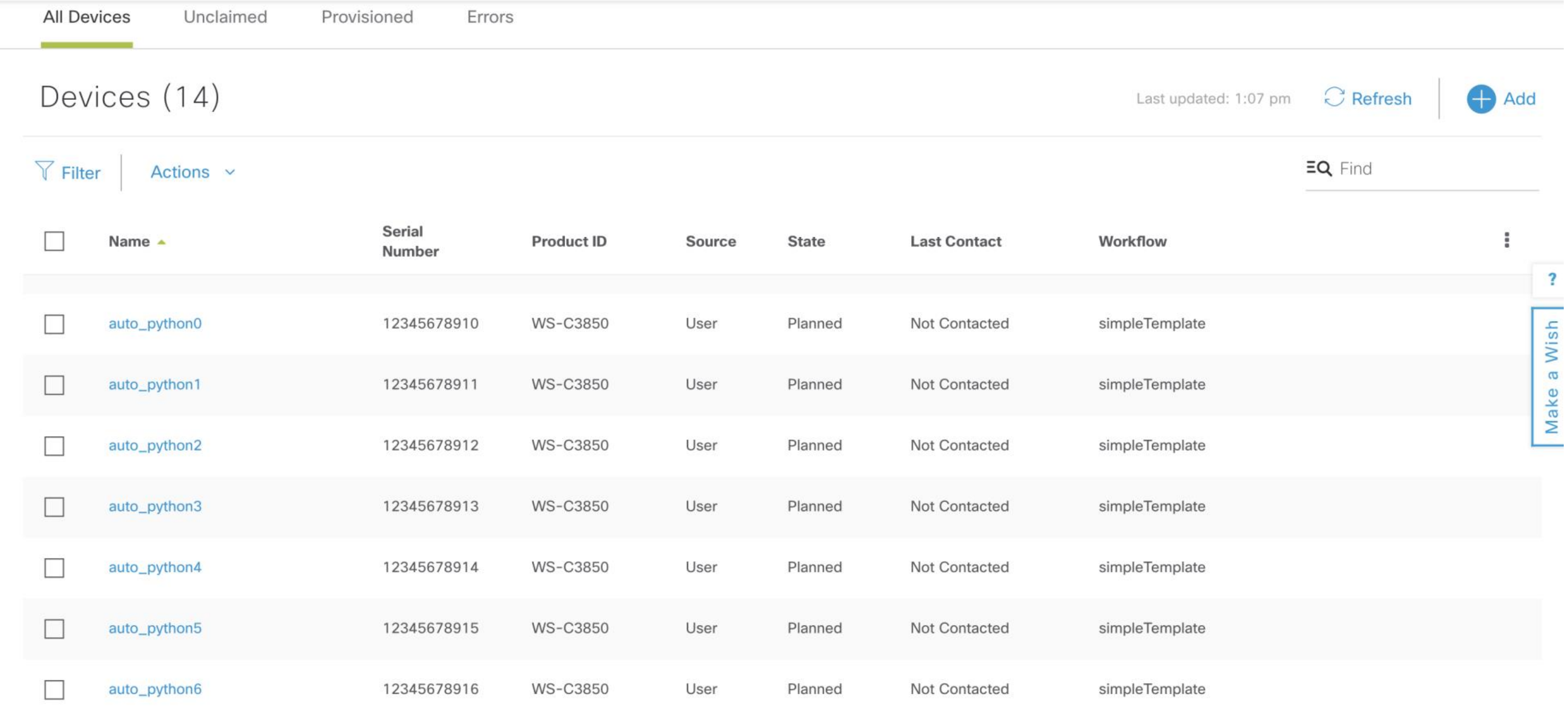


PnP Solution Components

Day 0



Automate Onboarding Rules



Devices (14)

□	Name	Serial Number	Product ID	Source	State	Last Contact	Workflow
□	auto_python0	12345678910	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python1	12345678911	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python2	12345678912	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python3	12345678913	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python4	12345678914	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python5	12345678915	WS-C3850	User	Planned	Not Contacted	simpleTemplate
□	auto_python6	12345678916	WS-C3850	User	Planned	Not Contacted	simpleTemplate

Examples

```
./10_add_and_claim.py work_files/bigtest.csv
./12_delete.py 12345678910

./12_delete.py work_files/bigtest.csv
```

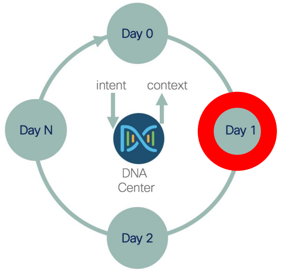
https://github.com/CiscoDevNet/DNAC-onboarding-tools

Notes

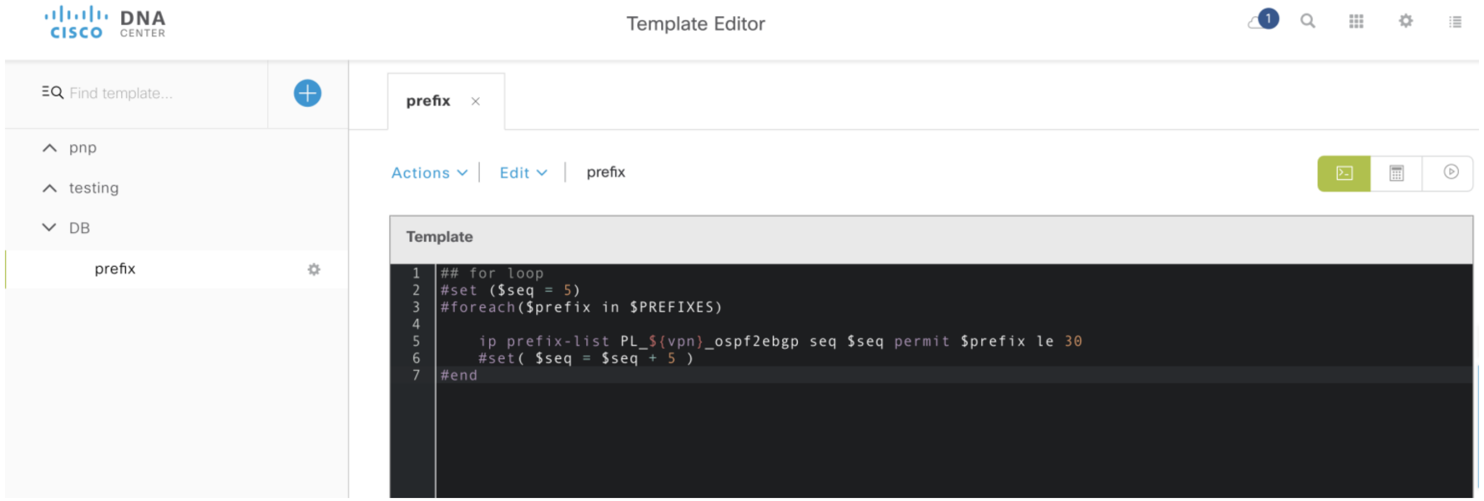
- No current way to upload an .xls of template variables for PnP provisioning rules, hence use API
- Composite templates are not supported for PnP
- Day 0 is typically a bootstrap config, pretty common across all devices in all locations, but can be more specific. Operational choice.

PNP API Summary

Endpoint	Verb	Description
api/v1/onboarding/workflow	GET/POST/ DELETE	CRUD on workflows
api/v1/onboarding/pnp-device/import	POST	Add a device for pre-provisioned workflow. Needs to be claimed.
api/v1/onboarding/pnp-device/import	POST	Claim a device. Attach a workflow to it
api/v1/onboarding/pnp-device	GET/PUT/POST/ DELETE	CRUD on devices
api/v1/file/config	POST	upload an config file, rather than use a template



Day 1



```
## for loop
#set ($seq = 5)
#foreach($prefix in $PREFIXES)

    ip prefix-list PL_${vpn}_ospf2ebgp seq $seq permit $prefix le 30
    #set( $seq = $seq + 5 )
#end
```

Examples

```
./template.py


./template.py --help


./template.py --template DB/prefix


./template.py --template DB/prefix \
  --device 192.168.200.80 \
  --params '{"vpn":"adam","PREFIXES":["1.1.1.1/29", "2.2.2.2/29"]}'
```

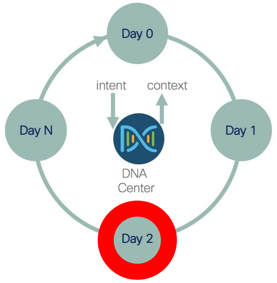
https://github.com/CiscoDevNet/DNAC-TemplateProgrammer

NOTES

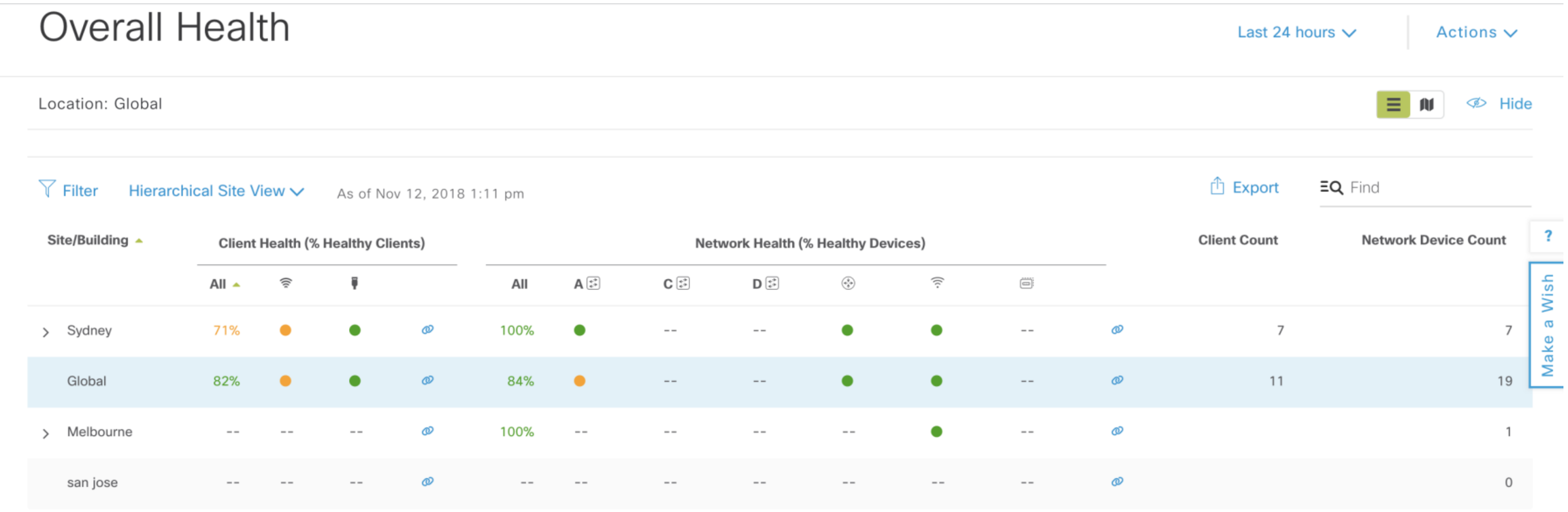
- All templates are versioned. Need to save && Commit. (There is API for this)
- Semantics of "forcePushTemplate". Default is NOT to re-apply a template with same variables.
- Default provisioning behavior is to apply a single (or nested template) to a device. Also includes site specific settings (aaa, ntp etc). API just pushes template to device(s)
- Multiple device requires unique parameters for template (can duplicate)

Template API Summary

Endpoint	Verb	Description
api/v1/template-programmer/project	GET/POST/ DELETE	CRUD on template projects
api/v1/template-programmer/template	GET/POST/ DELETE	CRUD on templates. A template is part of a project
api/v1/template-programmer/template/deploy	POST	Deploy a template to a device
/api/v1/template-programmer/template/deploy/status/{{deploymentId}}	GET	Status of the deployment job



Day 2



Overall Health

Location: Global

As of Nov 12, 2018 1:11 pm

Site/Building	Client Health (% Healthy Clients)				Network Health (% Healthy Devices)								Client Count	Network Device Count
	All				All	A	C	D						
Sydney	71%	●	●	🔗	100%	●	--	--	●	●	--	🔗	7	7
Global	82%	●	●	🔗	84%	●	--	--	●	●	--	🔗	11	19
Melbourne	--	--	--	🔗	100%	--	--	--	--	●	--	🔗		1
san jose	--	--	--	🔗	--	--	--	--	--	--	--	🔗		0

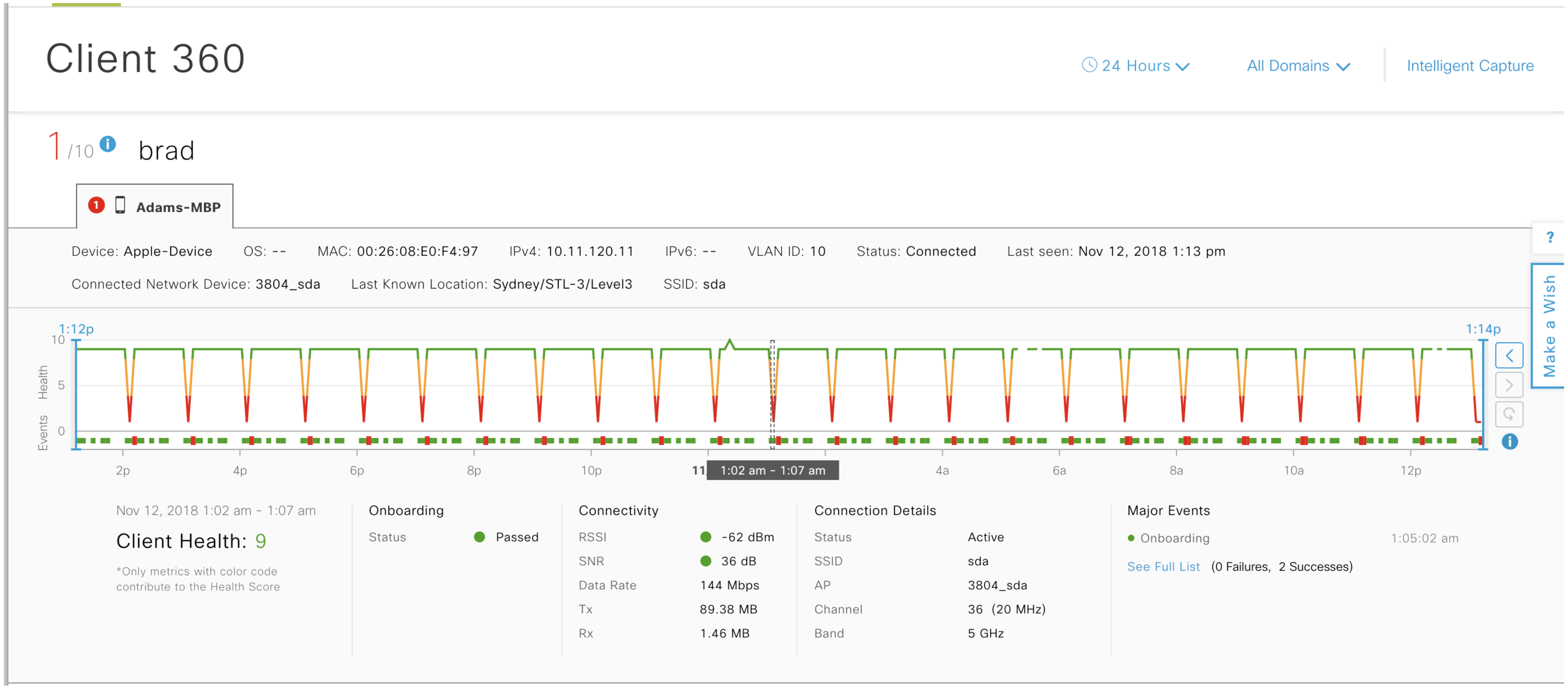
Examples

```
./assurance.py
./assurance.py --raw
./assurnace.py --timestamp <epoc>

#  these can also be run with --raw and --timestamp
./assurance.py --mac 00:26:08:E0:F4:97
./assurance.py --hostName 3504
```

https://github.com/CiscoDevNet/DNAC-Assurance

Client Enrichment



Client 360

24 Hours | All Domains | Intelligent Capture

1 /10 brad

❶ Adams-MBP

Device: Apple-Device   OS: --   MAC: 00:26:08:E0:F4:97   IPv4: 10.11.120.11   IPv6: --   VLAN ID: 10   Status: Connected   Last seen: Nov 12, 2018 1:13 pm

Connected Network Device: 3804_sda   Last Known Location: Sydney/STL-3/Level3   SSID: sda

Nov 12, 2018 1:02 am - 1:07 am

Client Health: 9

*Only metrics with color code contribute to the Health Score

Onboarding

Status	●	Passed

Connectivity

RSSI	●	-62 dBm
SNR	●	36 dB
Data Rate		144 Mbps
Tx		89.38 MB
Rx		1.46 MB

Connection Details

Status	Active
SSID	sda
AP	3804_sda
Channel	36 (20 MHz)
Band	5 GHz

Major Events

● Onboarding 1:05:02 am

See Full List (0 Failures, 2 Successes)

User Enrichment

```
https://dnac/dna/intent/api/v1/user-enrichment  GET
```

```
Headers:
__runsync: True
entity_type: mac_address
entity_value: c8:4c:75:68:b2:c0
```

```
[ {
  "userDetails" : {
    "id" : "C8:4C:75:68:B2:C0",
    "connectionStatus" : "CONNECTED",
    "hostType" : "WIRED",
    "userId" : null,
    "hostName" : null,
    "hostOs" : null,
    "hostVersion" : null,
    "subType" : "UNKNOWN",
    "lastUpdated" : 1528484918341,
    "healthScore" : [ {
      "healthType" : "OVERALL",
      "reason" : "",
      "score" : 10
    }, {
      "healthType" : "ONBOARDED",
      "reason" : "",
      "score" : 4
    }, {
      "healthType" : "CONNECTED",
      "reason" : "",
      "score" : 6
    } ],
    "hostMac" : "C8:4C:75:68:B2:C0",
    "hostIpV4" : "10.10.22.98",
    "hostIpV6" : [ ],
    "authType" : null,
<cont>
```

```
    "vlanId" : "1",
    "ssid" : null,
    "frequency" : null,
    "channel" : null,
    "apGroup" : null,
    "location" : "Sydney/NSD5",
    "clientConnection" :
"cat_9k_1.abc.inc",
    "connectedDevice" : [ ],
    "issueCount" : 0,
    "rssi" : null,
    "avgRssi" : null,
    "snr" : null,
    "avgSnr" : null,
    "dataRate" : null,
    "txBytes" : null,
    "rxBytes" : null,
    "dnsSuccess" : null,
    "dnsFailure" : null,
    "onboarding" : {
      "averageRunDuration" : null,
      "maxRunDuration" : null,
      "averageAssocDuration" : null,
      "maxAssocDuration" : null,
      "averageAuthDuration" : null,
      "maxAuthDuration" : null,
      "averageDhcpDuration" : null,
      "maxDhcpDuration" : null,
      "aaaServerIp" : null,
      "dhcpServerIp" : null,
```

```
    "port" : null
  },
  "connectedDevice" : [ {
    "deviceDetails" : {
      "family" : "Switches and Hubs",
      "type" : "Cisco Catalyst 9300 Switch",
      "location" : null,
      "errorCode" : null,
      "macAddress" : "f8:7b:20:67:62:80",
      "role" : "ACCESS",
      "apManagerInterfaceIp" : "",
      "associatedWlcIp" : "",
      "bootDateTime" : "2018-01-11 14:42:26",
      "collectionStatus" : "Managed",
      "interfaceCount" : "41",
      "lineCardCount" : "2",
      "lineCardId" : "1cd043ef-aaf7-4b2e-b720-
7af782b98b1c, a2b2467b-1692-46d4-8c64-e1765945efc1",
      "managementIpAddress" : "10.10.22.66",
      "memorySize" : "889226872",
      "platformId" : "C9300-24UX",
      "reachabilityFailureReason" : "",
      "reachabilit
      "snmpContact
      "snmpLocatio
      <cont>
```

NOTES:
Synchronous execution with header __runsync
Entity_type can be mac_address or network_user_id

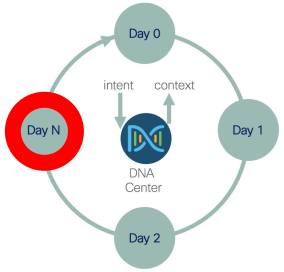
Notes

- Be aware of __runsync == "true", otherwise need to GET result
  - Header, not query param.
- Timestamp is in milli-epoch (divide by 1000 to get epoch seconds)
- Finite history for data (default 7 days)
- User Enrichment API use headers, not query param for mac_address etc

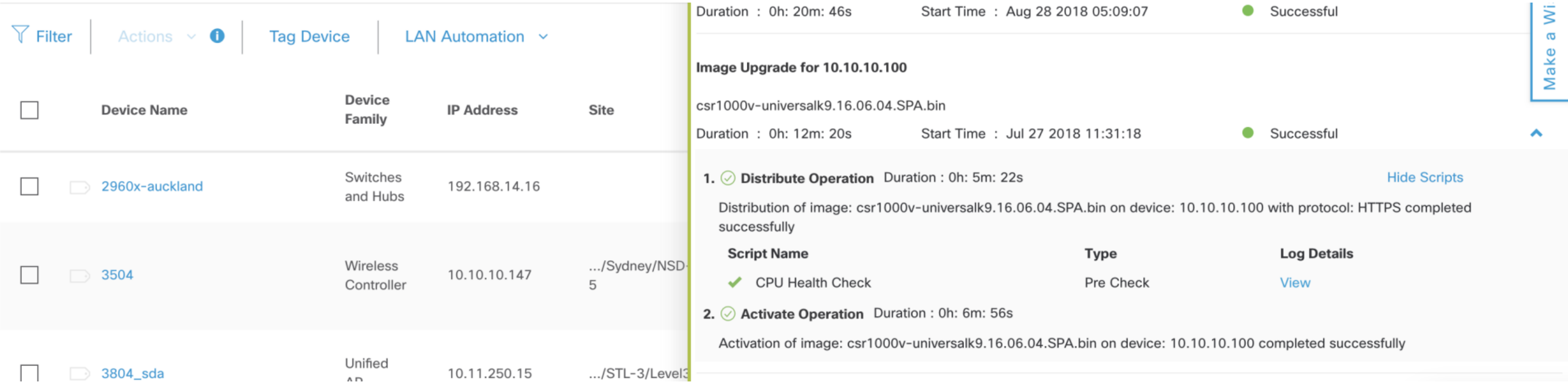
Assurance API Summary

Endpoint	Verb	Description
api/v1/site-hierarchy?timestamp=	GET	GET health of all sites
api/v1/network-health?startTime=&endTime=	GET	GET Health of Network Devices
api/v1/client-health?startTime=&endTime=	GET	GET Health of Clients
api/v1/client-detail?timestamp={{time}}&macAddress={{mac}}	GET	GET Client health Detail
api/v1/device-detail? Timestamp={}&searchBy={}&identifier=nwDeviceName	GET	GET network device detail

api/v1/user-enrichment	GET	User/device enrichment.
		Passed as header
		`entity_type: mac_address`
		`entity_value:c8:4c:75:68:b2:c0`



Day N



□	Device Name	Device Family	IP Address	Site
□	2960x-auckland	Switches and Hubs	192.168.14.16	
□	3504	Wireless Controller	10.10.10.147	.../Sydney/NSD-5
□	3804_sda	Unified AP	10.11.250.15	.../STL-3/Level3

Duration : 0h: 20m: 46s   Start Time : Aug 28 2018 05:09:07   ● Successful

Image Upgrade for 10.10.10.100

csr1000v-universalk9.16.06.04.SPA.bin

Duration : 0h: 12m: 20s   Start Time : Jul 27 2018 11:31:18   ● Successful

1. ✓ **Distribute Operation** Duration : 0h: 5m: 22s   Hide Scripts

Distribution of image: csr1000v-universalk9.16.06.04.SPA.bin on device: 10.10.10.100 with protocol: HTTPS completed successfully

Script Name	Type	Log Details
✓ CPU Health Check	Pre Check	View

2. ✓ **Activate Operation** Duration : 0h: 6m: 56s

Activation of image: csr1000v-universalk9.16.06.04.SPA.bin on device: 10.10.10.100 completed successfully

Examples

```
./list_images.py


./distribute.py --tag upgrade9k \
  --image cat9k_iosxe.16.06.02s.SPA.bin
```

https://github.com/CiscoDevNet/DNAC-Assurance

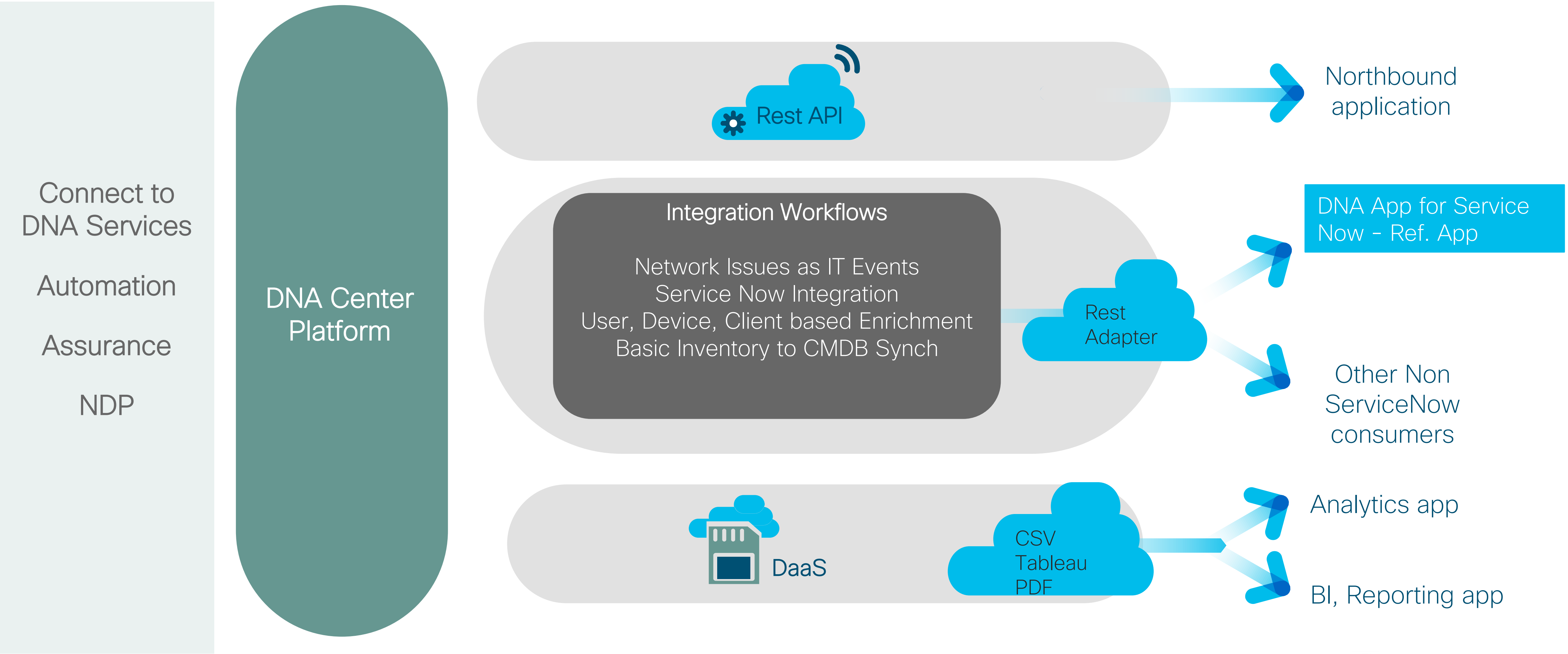
Notes

- SWIM application can only upgrade to "golden image" by default. Can specify golden images for site/role.
- SWIM events (Golden Image compliance) available as "event"
- Need to re-sync for DNA-Center to show device upgraded
  - Default resync interval is 25mins

SWIM API Summary

Endpoint	Verb	Description
api/v1/image/importation?name=<str>	GET	GET images that have <str> in their name
api/v1/image/distribution	POST	Distribute images to devices
api/v1/image/activation/device	POST	Activate images on the devices
api/v1/mage/importation/source/url	POST	Upload image to DNAC
api/v1/device-image/device?id=<idList>	POST	Gets the status of golden images for the list of devices. Comma separated list of deviceId
api/v1/device-image/device<deviceId>file/<filename>	DELETE	Removes the file from the device.

DNA Center Platform ITSM Integration Overview



Connect to DNA Services

Automation

Assurance

NDP

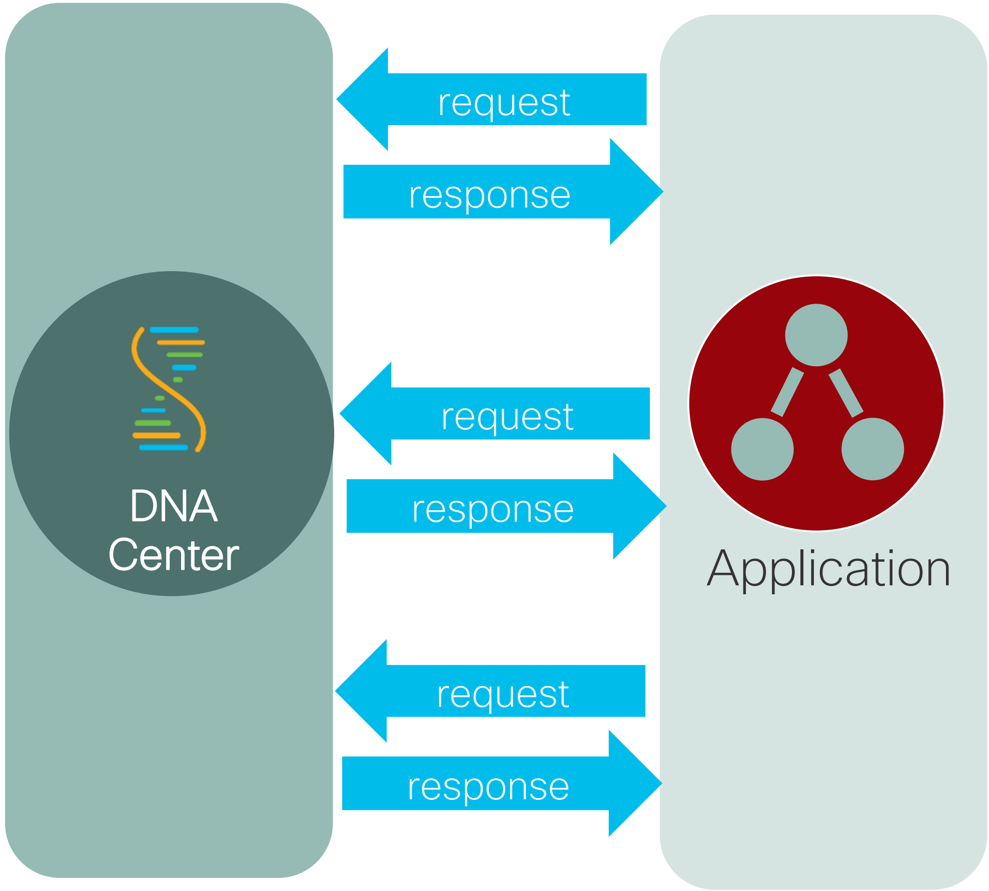
DNA Center Platform

Integration Workflows

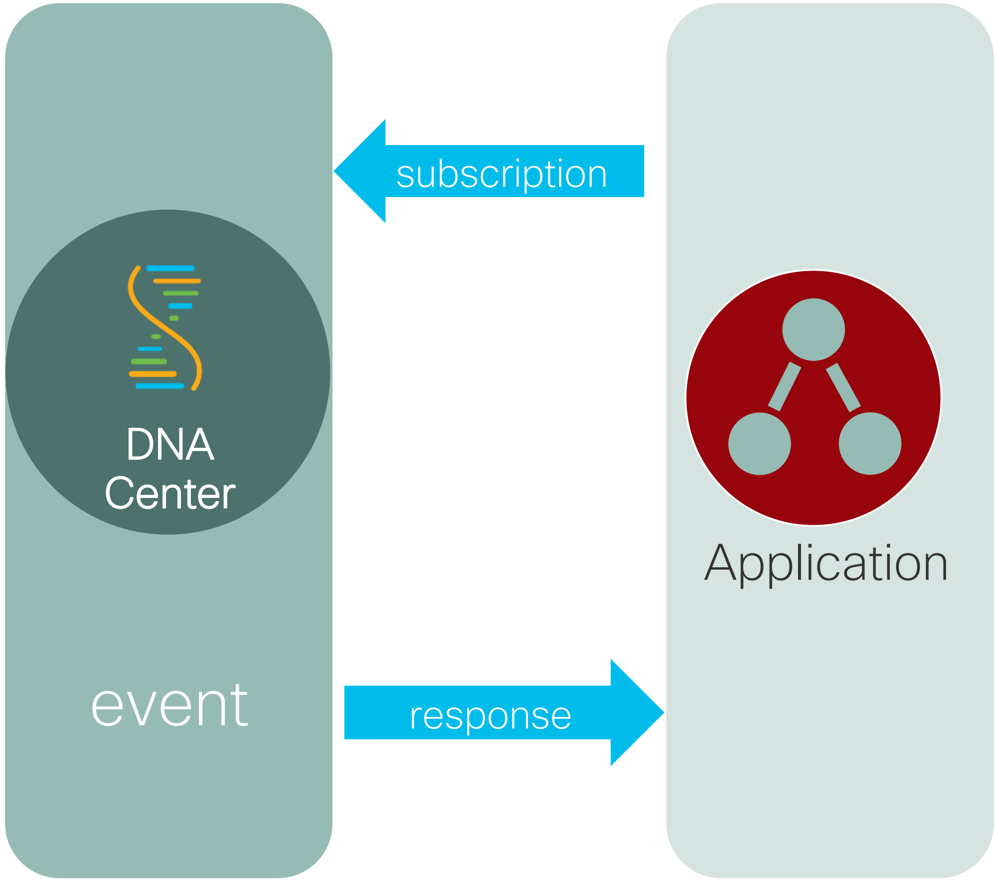
Network Issues as IT Events
Service Now Integration
User, Device, Client based Enrichment
Basic Inventory to CMDB Synch

Polling vs Notification

Polling (Pull)



Notification (Push)



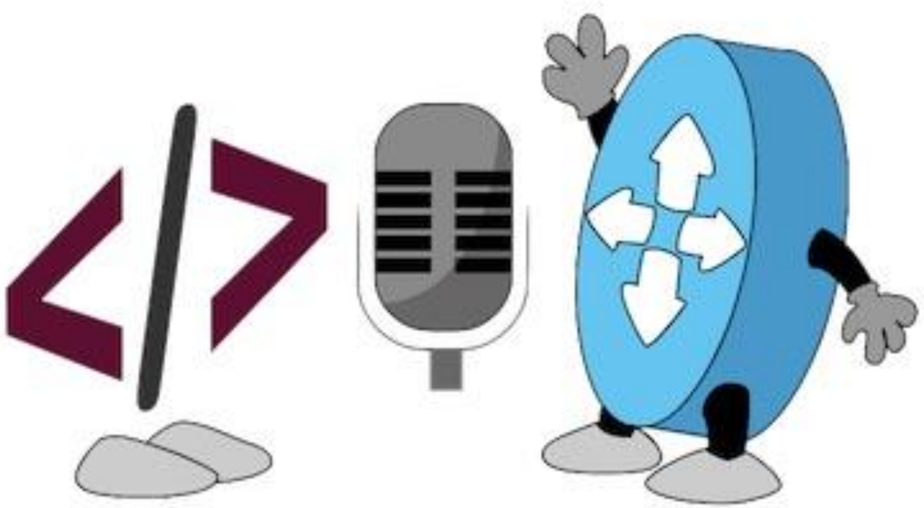
Available via Webhook

- SWIM compliance - generic
- Issues - generic
- SNOW Integration
  - CMBD
  - SWIM
  - Issues

Current internal implementation is minimum15 + 30min polling and push.

What did we Talk about?

- DNA Center
- Device Lifecycle Augmented via API



Webinar Resource List

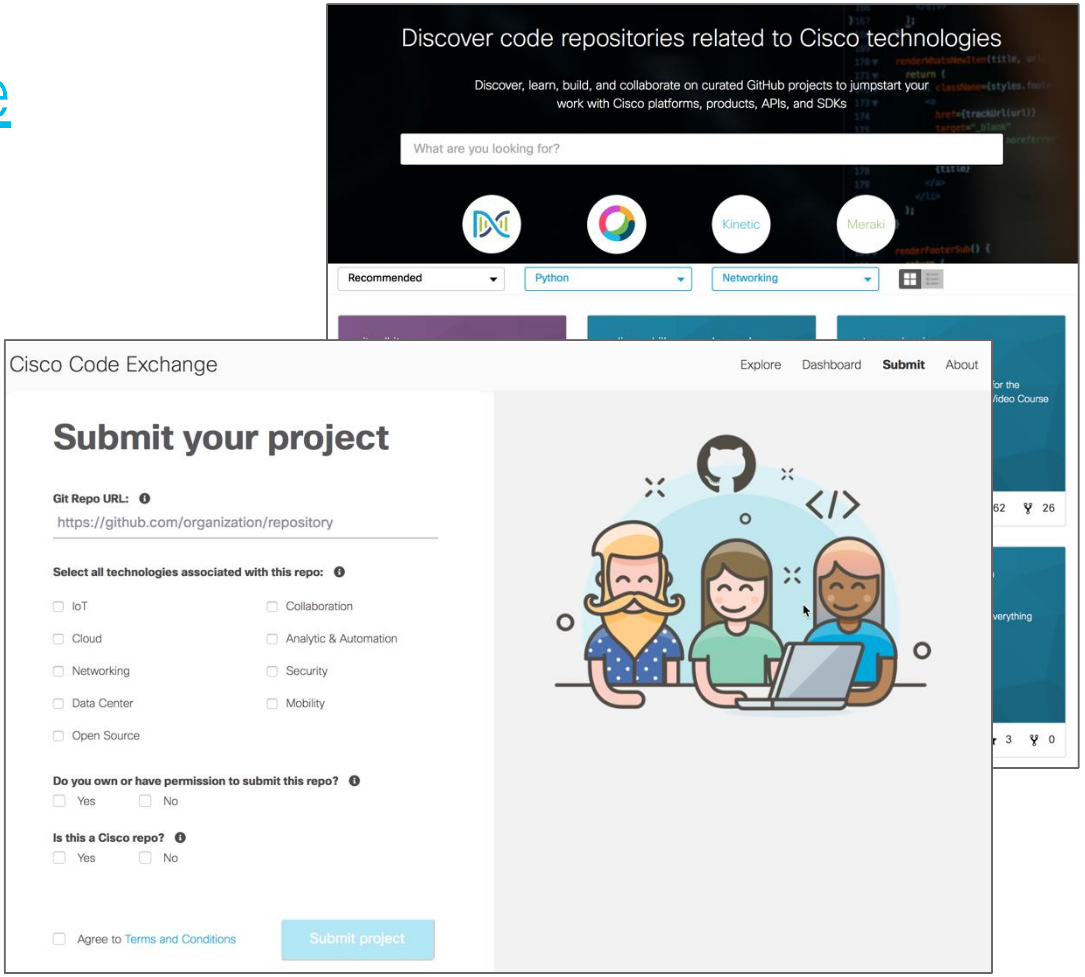
- Docs and Links
  - https://developer.cisco.com/site/dna-center-rest-api/
  - https://blogs.cisco.com/tag/dna-center
- Learning Labs
  - Laptop Setup http://cs.co/lab-dev-setup
  - DNA Center Learning Labs http://cs.co/lab-dnacenter
- DevNet Sandboxes
  - DNA Center Always On http://cs.co/sbx-dnac-ao & https://sandboxdnac.cisco.com
- Code Samples
  - http://cs.co/code-dnacenter

NetDevOps Live! Code Exchange Challenge

developer.cisco.com/codeexchange

Build a Python script that leverages one of the DNA Center APIs.

Example: Report back on the health of a client.



Discover code repositories related to Cisco technologies

Discover, learn, build, and collaborate on curated GitHub projects to jumpstart your work with Cisco platforms, products, APIs, and SDKs

What are you looking for?

Recommended ▾ Python ▾ Networking ▾

Cisco Code Exchange

Submit your project

Git Repo URL: https://github.com/organization/repository

Select all technologies associated with this repo:

□ IoT
□ Cloud
□ Networking
□ Data Center
□ Open Source
□ Collaboration
□ Analytic & Automation
□ Security
□ Mobility

Do you own or have permission to submit this repo?
□ Yes □ No

Is this a Cisco repo?
□ Yes □ No

□ Agree to Terms and Conditions

Submit project

Looking for more about NetDevOps?

- NetDevOps on DevNet
  developer.cisco.com/netdevops
- NetDevOps Live!
  developer.cisco.com/netdevops/live
- NetDevOps Blogs
  blogs.cisco.com/tag/netdevops
- Network Programmability Basics Video Course
  developer.cisco.com/video/net-prog-basics/

Got more questions? Stay in touch!

CISCO DEVNET

LEARN CODE INSPIRE CONNECT

Adam Radford

developer.cisco.com

aradford@cisco.com

@adamradford123

http://github.com/aradford123

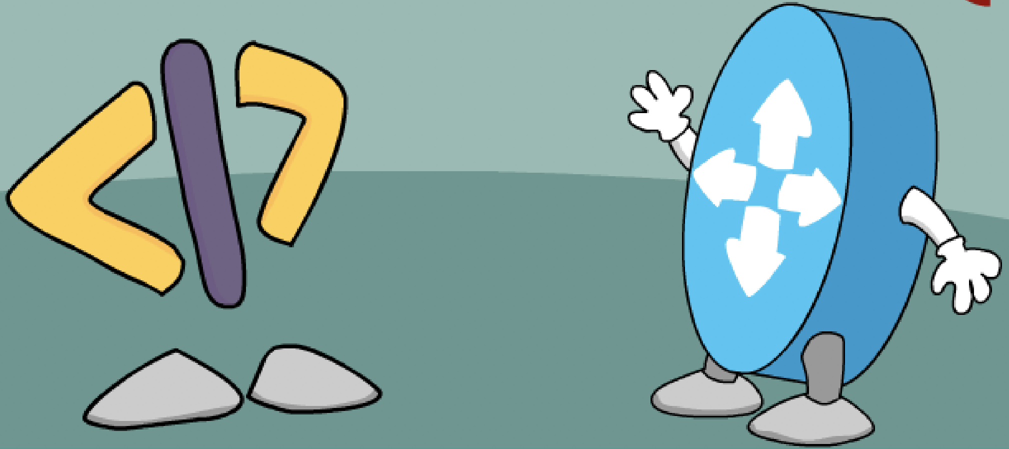
@CiscoDevNet

facebook.com/ciscodevnet/

http://github.com/CiscoDevNet

NETDEVOPS{LIVE!}

CISCO DEVNET



https://developer.cisco.com/netdevops/live

@netdevopslive