The bridge to possible

ACI Troubleshooting

Optimize your APIC User Experience through API Mastery

Gabriel Monroy, CX Principal Engineer

BRKDCN-2635

cisco Live!

Cisco Webex App

https://ciscolive.ciscoevents.com/ciscolivebot/#BRKDCN-2635

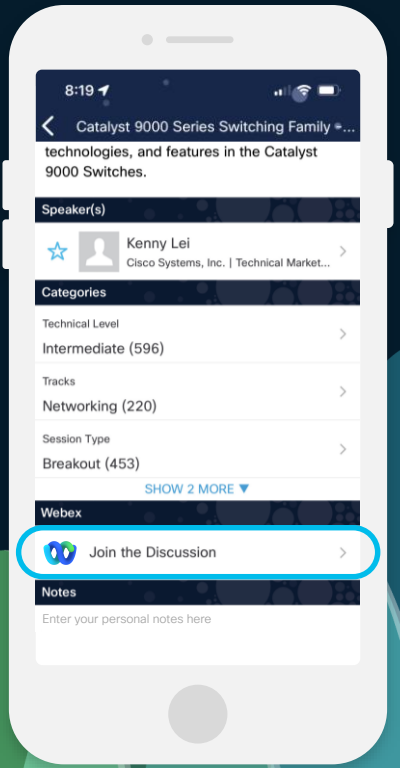
Questions?

Use Cisco Webex App to chat with the speaker after the session

How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 7, 2024.



Agenda

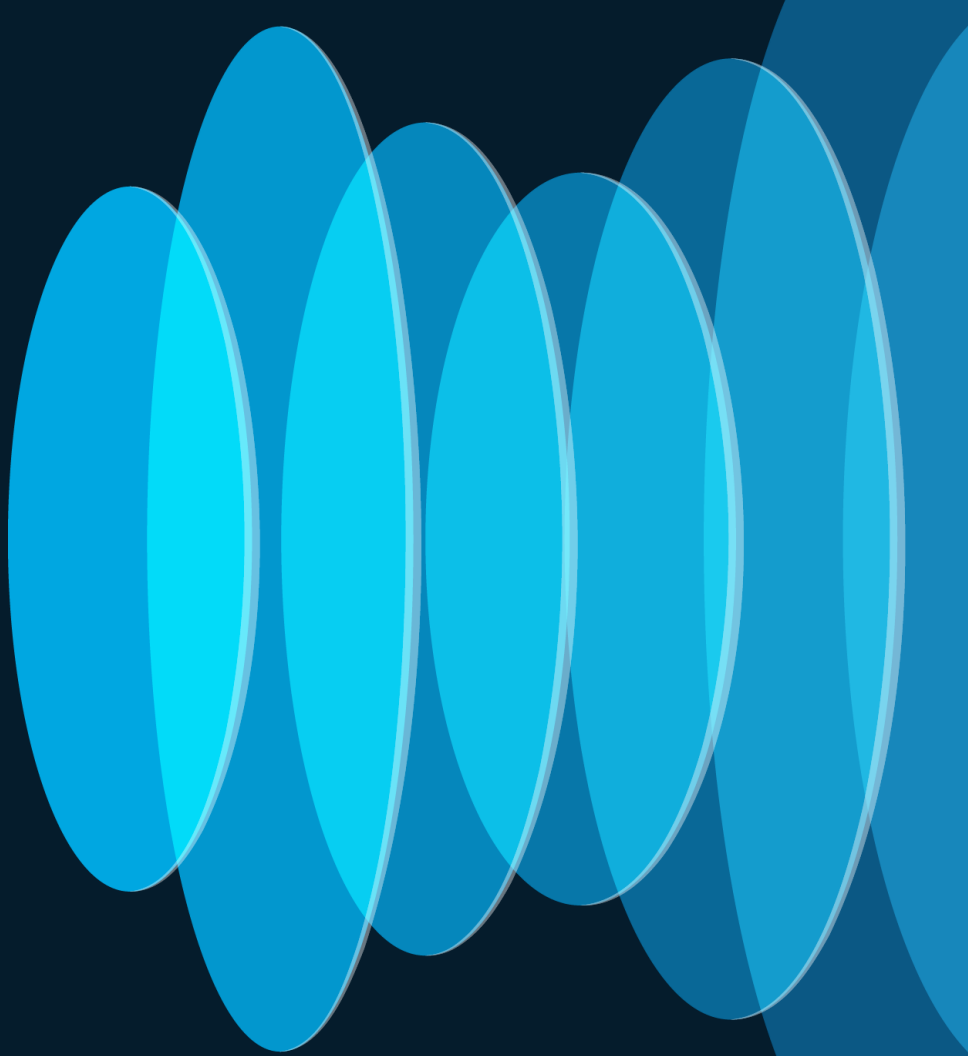
- You can’t spell APIC without API
- Under-the-hood of an APIC Cluster
- API Troubleshooting Tools
- Common API Issues
- Optimize Visibility with Query Subscriptions

Glossary of Acronyms

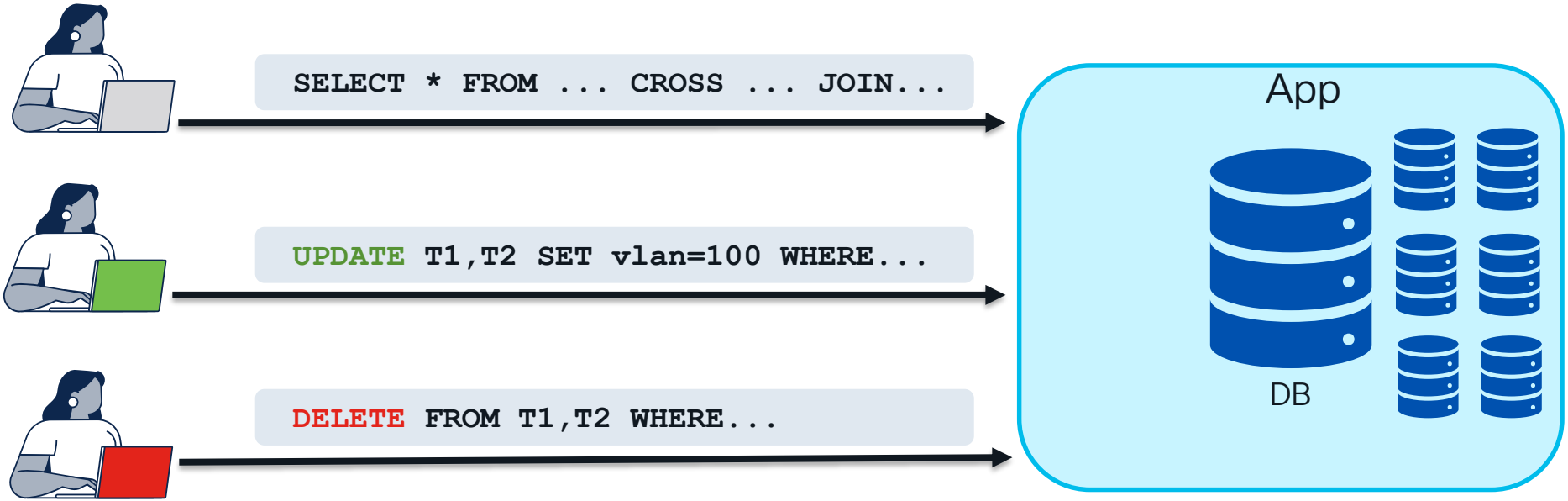
Acronyms	Definitions
ACI	Application Centric Infrastructure
APIC	Application Policy Infrastructure Controller
API	Application Programming Interface
DME	Data Management Engine (ACI Service)
FNV	Fabric Node Vector (ACI Switch registration info)
MIT	Management Information Tree
RV	Replica Vector (Shard/Replica State)
REST	Representational State Transfer – A specific architectural style for web services

Acronyms	Definitions
extXMLApi	DME that works with NGINX
PD	Policy Distributor – Performs Policy Validation
PE	Policy Element – Main Switch Policy DME
PM	Policy Manager – Main APIC Policy DME
MO	Managed Object (an ACI object saved in DME DBs)
DB	Database, can be split into shards and replicated across APICs

You can’t spell APIC without API

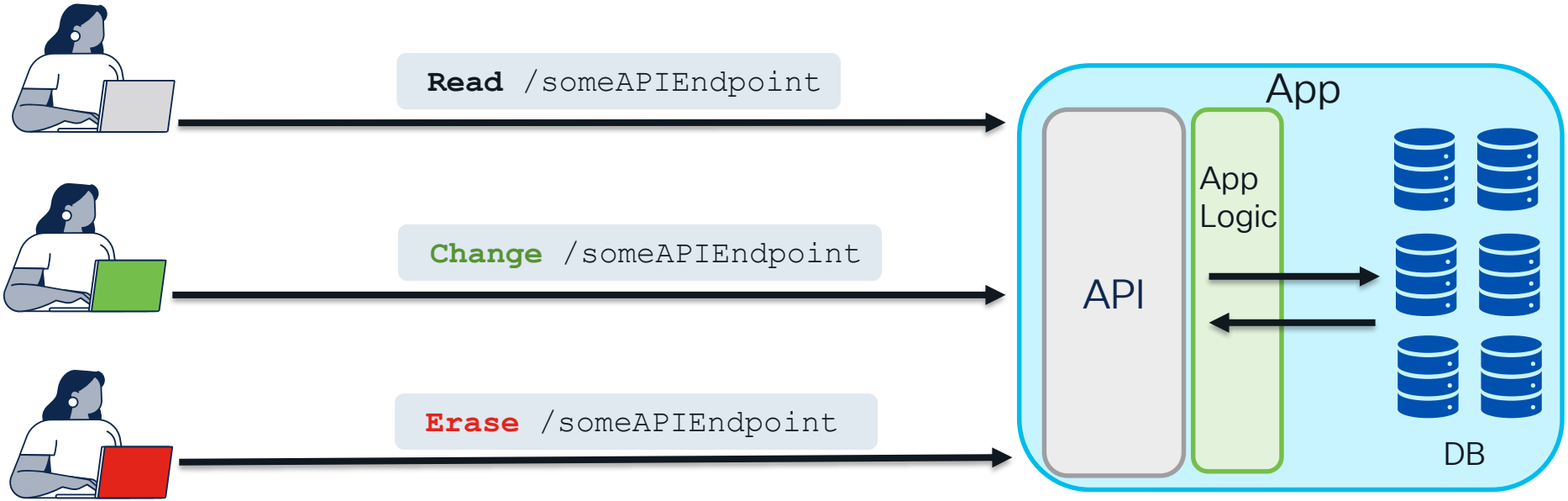


Before APIs



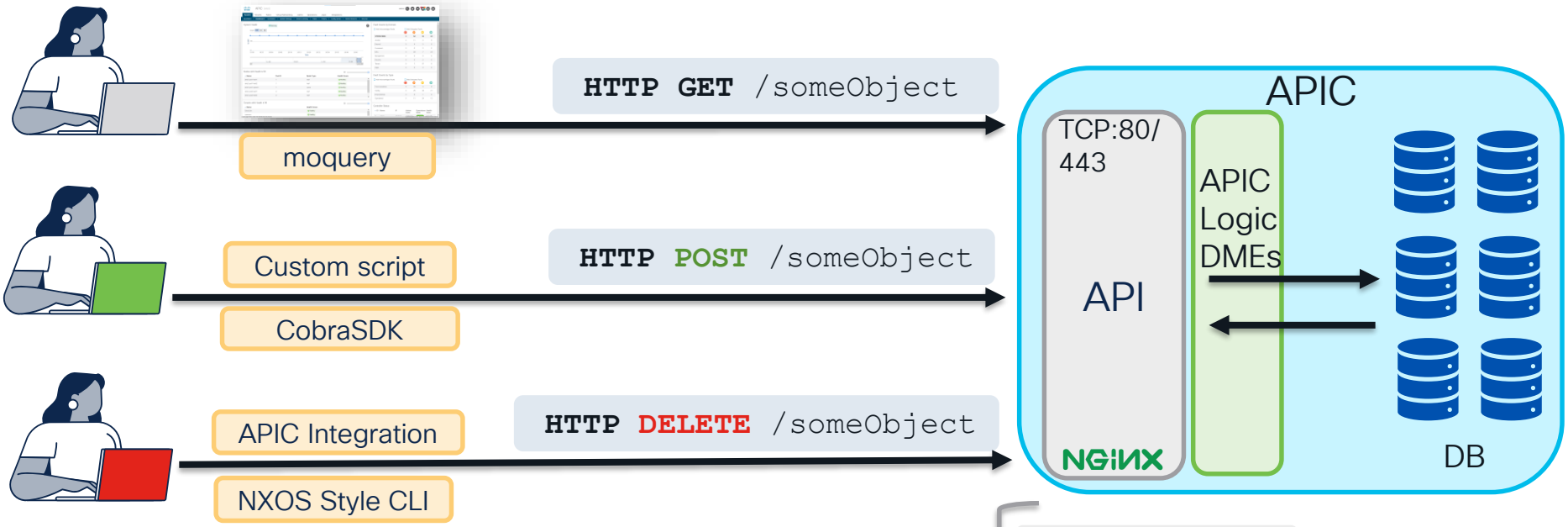
DB Manipulation is not App user-friendly

Why have an API?



App Interface Simplification

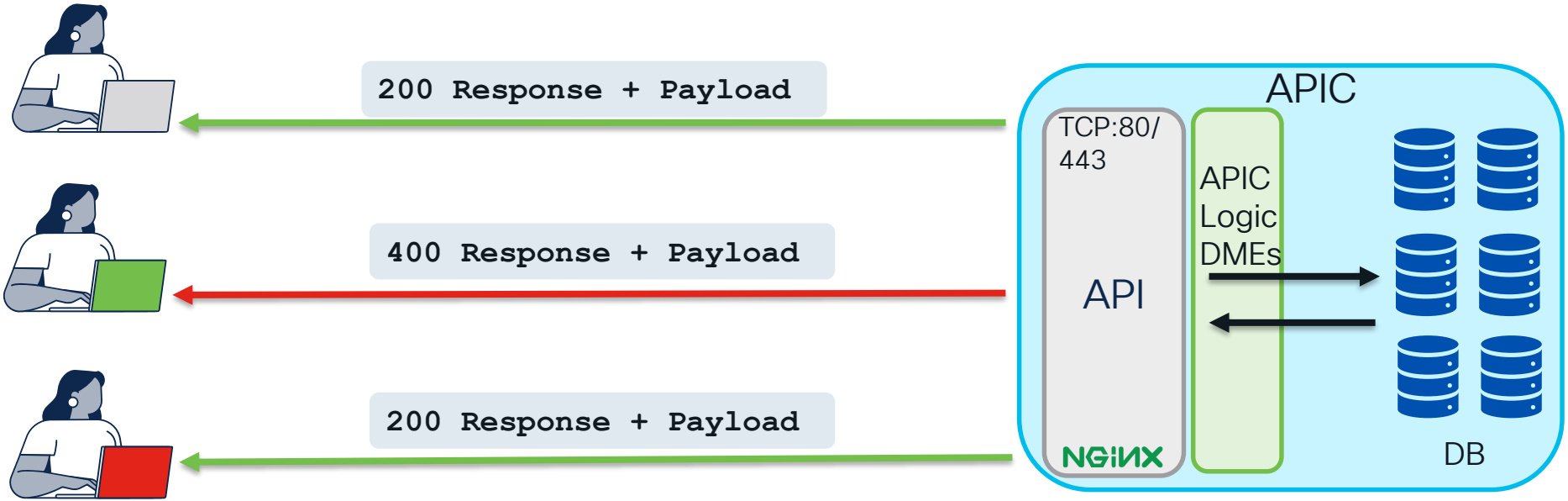
All ACI Interactions are via a REST API



APIC supported REST Methods:

- GET = R/O
- POST = Create/Modify/Delete
- DELETE = Delete

Read the Response code



Respond with HTTP/S Status Codes

2xx = OK

4xx = Client Error

5xx = Server Error

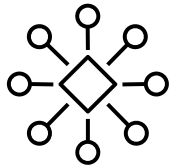
Anatomy of an APIC API Request

`/api/node/mo/uni/tn-CL/ap-CLApp/epg-DB.json`

DN Query

`/api/node/class/fvAEPg.json`

Class Query



`/node{-ID}`

Which Node?*

`/mo /{DN}`

`/class /{className}`

What type of Query?

Distinguished name or Object Class

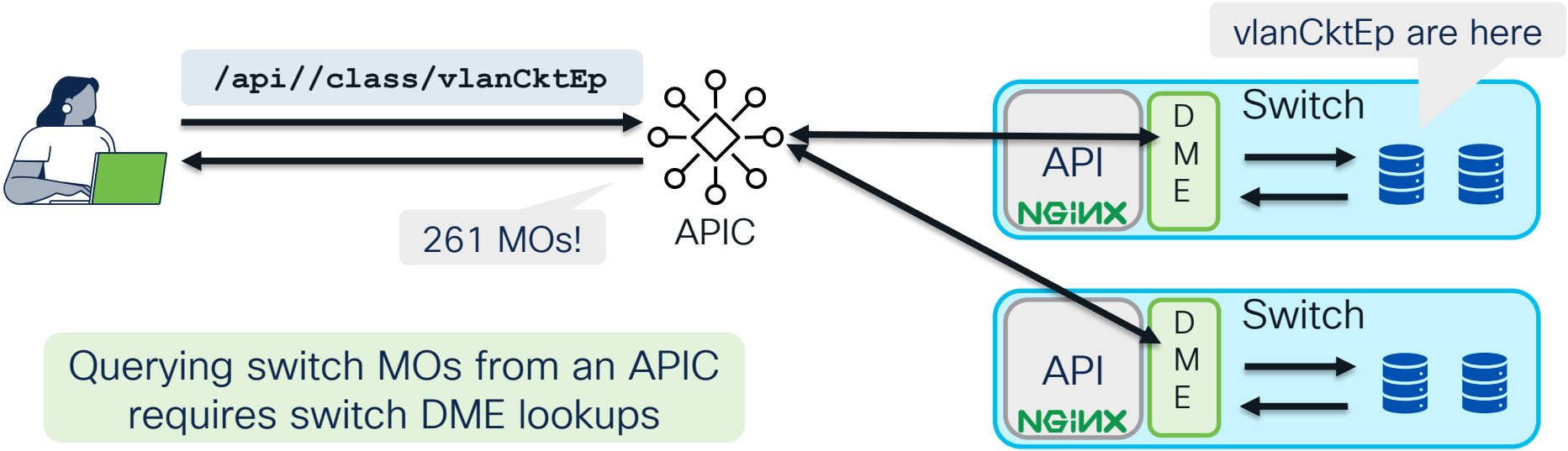
`.{xml|json}`

Response Encoding

`?[options]`

Filters, Selectors, query modifiers

What about Switches?



Querying switch MOs from an APIC requires switch DME lookups

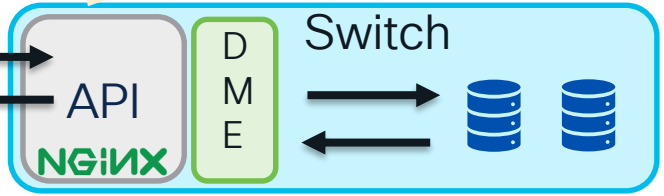
```
apic# moquery -c vlanCktEp
Total Objects shown: 261

# vlan.CktEp
dn : topology/pod-1/node-101/sys/ctx-[vxlan-x]/bd-[vxlan-y]/vlan-[vlan-777]
```

Contains pointer to source node

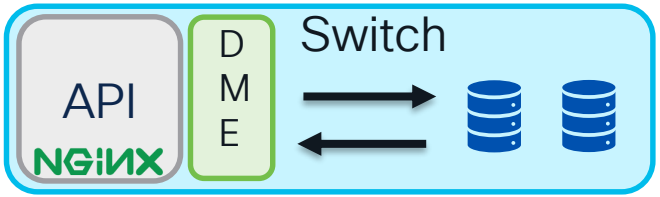
The Switches have an API too!

/api//class/vlanCktEp

Why would we ever use this?



61 MOs!

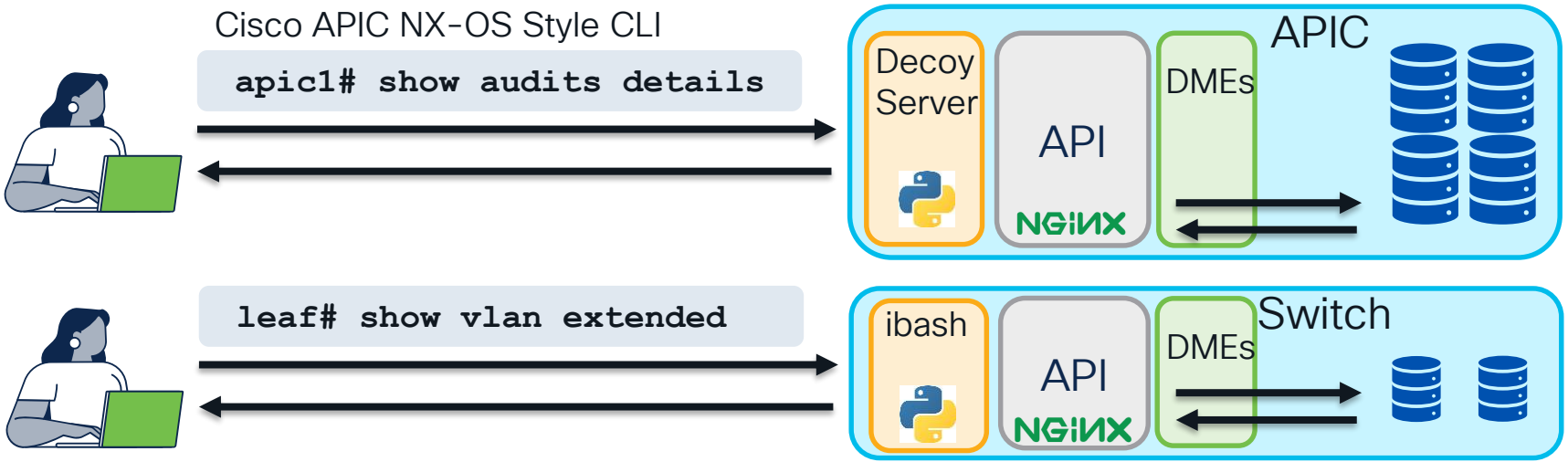


Query a Switch API to return just that Switch's Objects

```
Leaf-101# moquery -c vlanCktEp
Total Objects shown: 61

# vlan.CktEp
dn : sys/ctx-[vxlan-x]/bd-[vxlan-y]/vlan-[vlan-777]
```

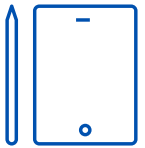
Source node pointer **NOT** tagged from switch perspective

Our first script run; CLI Commands




```
leaf# show vlan extended
 VLAN Name                             Encap            Ports
 ---- -------------------------------- ---------------- -----------------------
 13   CL2024:ap1:epg1                  vlan-3636        Eth1/33
 14   CL2024:ap1:epg1                  vlan-3769        Eth1/34, Eth1/43
```

“show” command filtering via NXOS Style CLI



? for filtering options

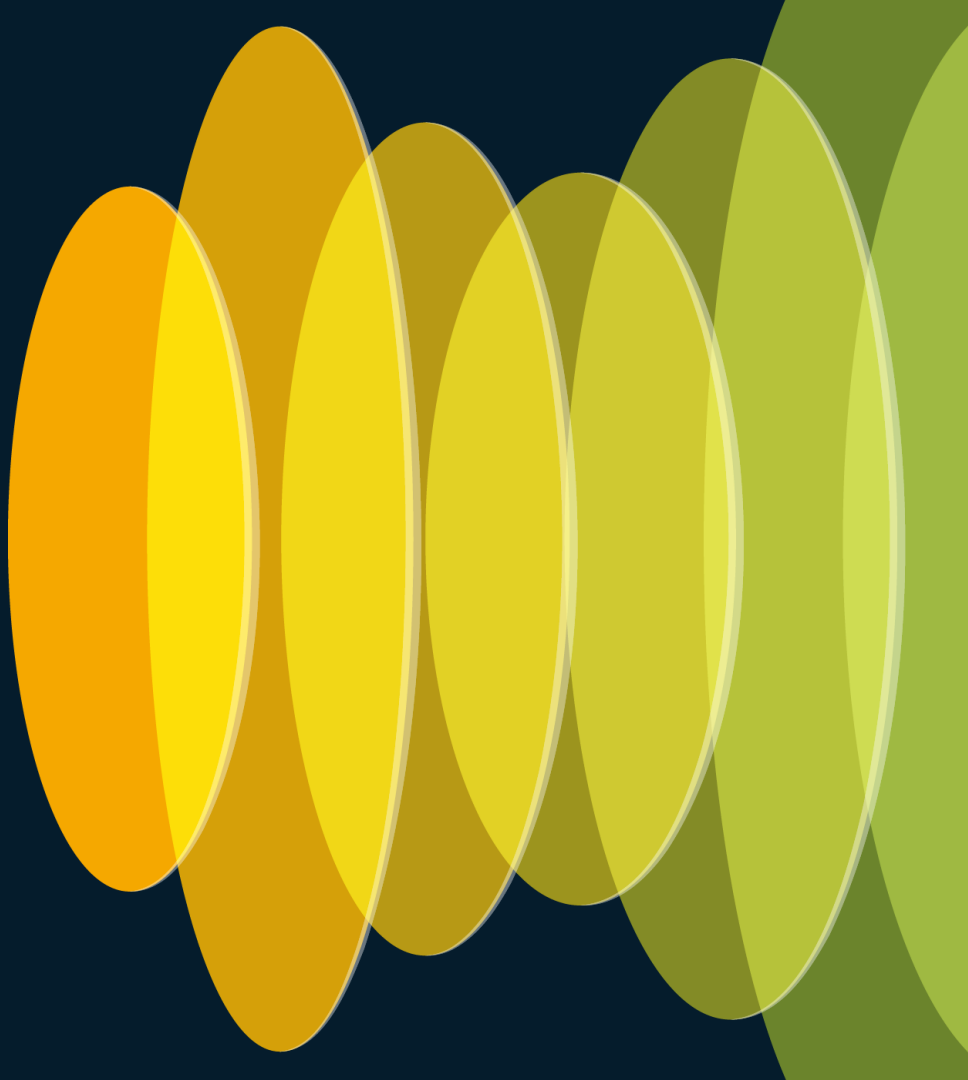
```
apic# show audits detail ?
 <CR>
 action        Object action indicator
 end-time      Logs created in time interval
 id            Log ID
 last-days     Logs created in time interval
 last-hours    Logs created in time interval
 last-minutes  Logs created in time interval
 start-time    Logs created in time interval
 tenant        Show Tenants Information
 user          Name of user
```

```
apic1# show faults history detail ?
 ...
 end-time      Fault activity in time interval
 l4l7-cluster  Show L4 L7 Device information
 l4l7-graph    Show L4 L7 Graph information
 last-days     Fault activity in time interval
 last-hours    Fault activity in time interval
 last-minutes  Fault activity in time interval
 min-severity  Minimum severity
 severity      Severity
 spine         Show command for spine
 start-time    Fault activity in time interval
 tenant        Show Tenants Information
```

Easy time filtering options

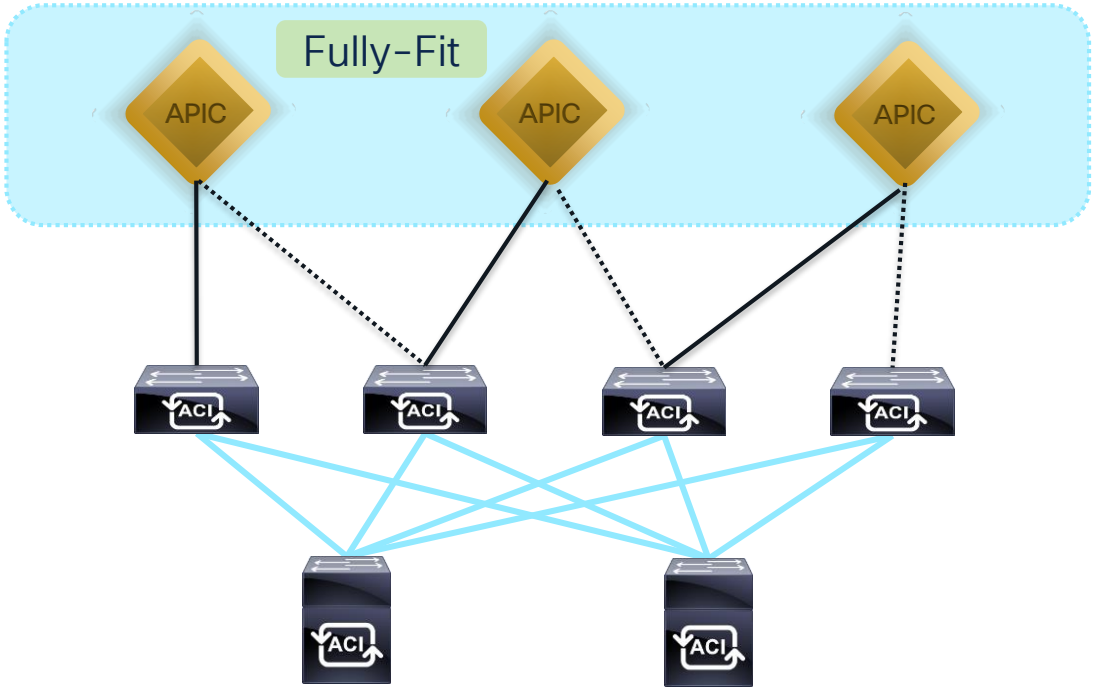
```
apic# show audits detail start-time 2022-03-15T00:00:00 end-time 2022-03-16T00:00:00

apic# show audits detail last-hours 5
```

Under-the-hood of an APIC Cluster

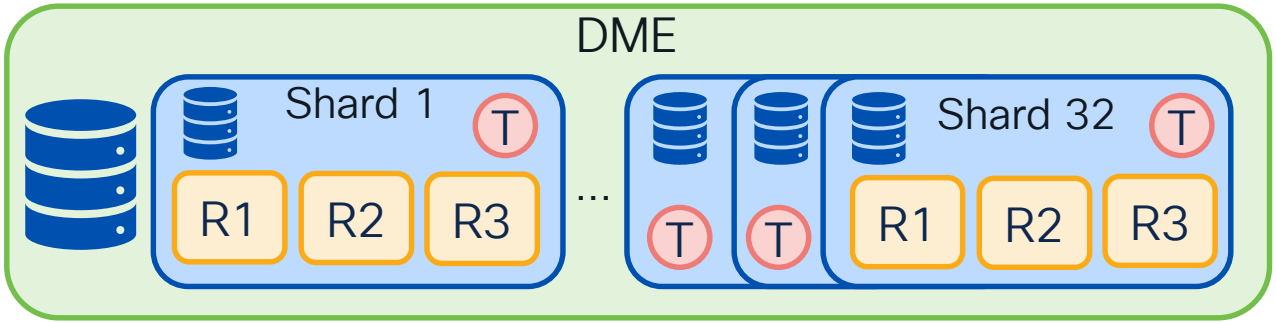


cisco Live!

An APIC Cluster



DMEs, Shards, Replicas and Tokens



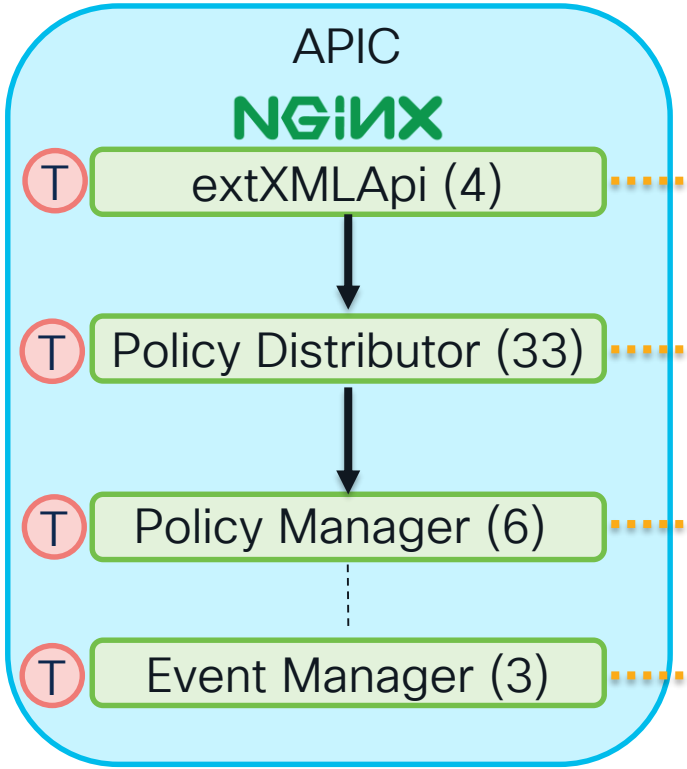
Not every DME is sharded

There are always 32 Shards*

There are always 3 Replicas

Shards use a Token Queuing System

Main APIC DMEs for “general” configuration



Hosts API, manages aaa fsm

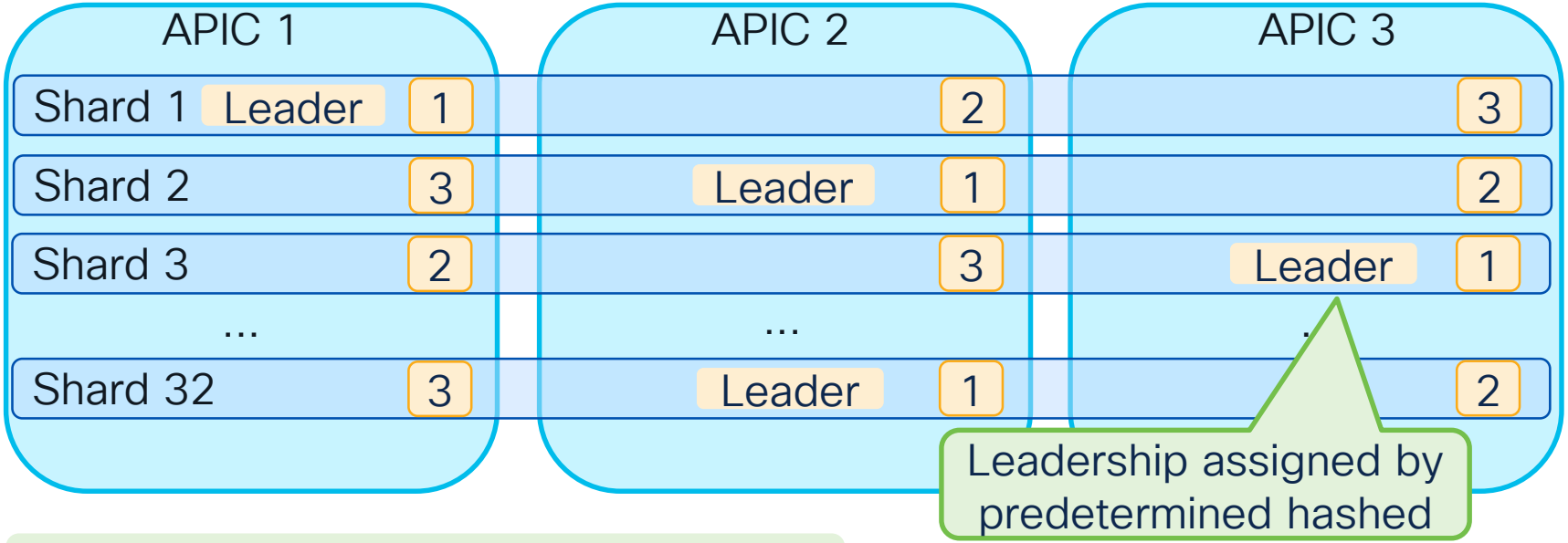
First level policy validation

Main Logical Policy DB, sends policy to other APIC and Switch DMEs

Create and manage MO events

Each handoff and replication costs tokens

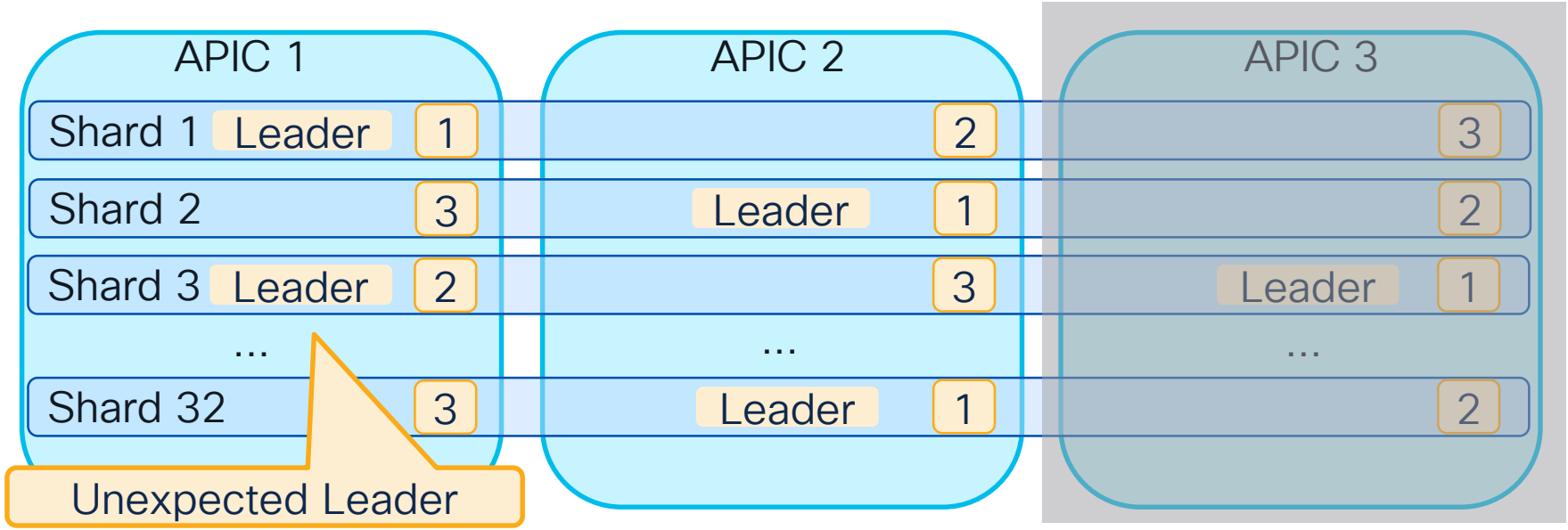
Fully-Fit 3 APIC Cluster – Shard Leadership



Shard Leader handles Write Operations

Multiple APICs = Distributed processing

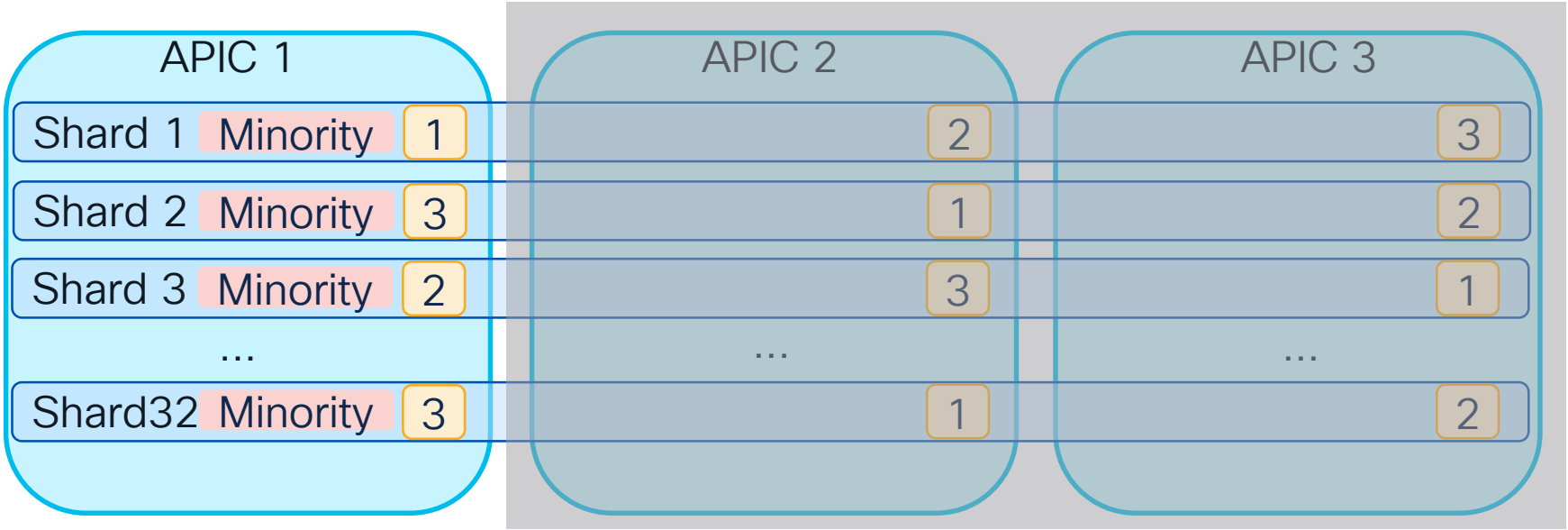
Degraded Leadership



Shards still have the Majority of Replicas; 2 out of 3

Write operations still available

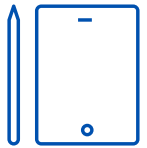
Diverged APIC Cluster – Minority State



Minority state = RO Operations

APIC 1 isolated from APIC 2 and 3

Check acidiag manual for DME ID



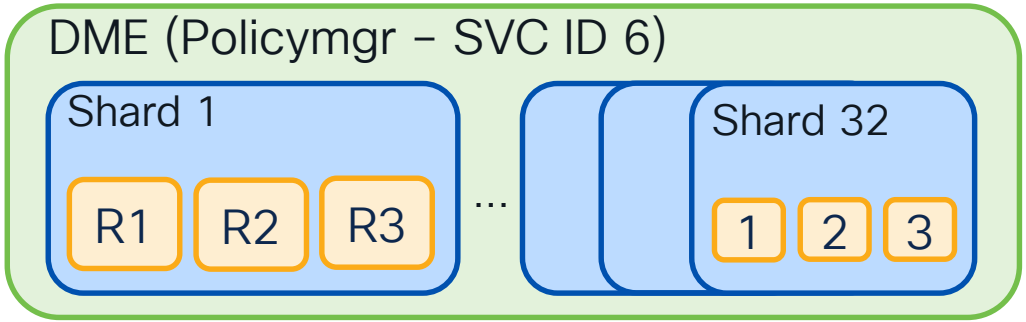
```
apic# man acidiag
Service IDs:
       ...
       3 - eventmgr
       4 - extXMLApi
       5 - policyelem
       6 - policymgr
       7 - reader
       8 - ae
       9 - topomgr
       ...
       31 - opflexp
       32 - analytics
       33 - policydist
       34 - plgnhandler
       35 - domainmgr
       36 - licensemgr
```

acidiag rvread – shards and replicas

```
apic# acidiag rvread
\- unexpected state;    /-unexpected mutator;
s->R  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
r->R123123123123123123123123123123123123123123123123123123123123123123123123123123123123123123123
  1
 ...
  6 ///
 ...
 49
-----------------------------------------------
```

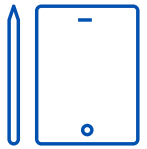
Shard 32

R1, 2, 3



A boring rvread is a healthy rvread

acidiag rvread svcID shardID



avread accepts <svc> <shard> <replica>

```
apic1# acidiag rvread -h
usage: acidiag rvread [-h] [subcommand [subcommand ...]]

positional arguments:
  subcommand  optional svcID [shardID [replicID]]
```

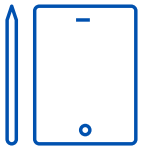
Example: svcId 6 (PM) and shard 1

```
apic1# acidiag rvread 6 1
(6,1,1)    st:6 lm(t):1(2022-03-09T03:56:31.224+00:00) le: reSt:LEADER ...
(6,1,2)    st:6 lm(t):2(2022-03-09T03:48:04.444+00:00) le: reSt:FOLLOWER ...
(6,1,3)    st:6 lm(t):3(2022-03-09T04:21:04.208+00:00) le: reSt:FOLLOWER ...
------------------------------------------
clusterTime=<diff=-1099290 common=2022-04-15T21:33:22.809+00:00 local=2022-04-15T21:51:42.099+00:00
pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2020-01-08T20:53:53.081+00:00)>>
```

APIC 1 is leader of Shard 1

Which APIC has that Replica; APIC 1 happens to have Replica 1

acidiag rvread <svcID> - Find Shard Leaders



```
apic1# acidiag rvread 6 | awk -F 'voGr' '{print $1}'
(6,1,1)    st:6 lm(t):1(2022-03-09T03:56:31.224+00:00) le: reSt:LEADER
(6,1,2)    st:6 lm(t):2(2022-03-09T03:48:04.444+00:00) le: reSt:FOLLOWER
(6,1,3)    st:6 lm(t):3(2022-03-09T04:21:04.208+00:00) le: reSt:FOLLOWER
(6,2,1)    st:6 lm(t):2(2022-03-09T03:56:32.244+00:00) le: reSt:LEADER
(6,2,2)    st:6 lm(t):3(2022-03-09T04:21:02.145+00:00) le: reSt:FOLLOWER
(6,2,3)    st:6 lm(t):1(2022-03-09T03:19:53.559+00:00) le: reSt:FOLLOWER
...
```

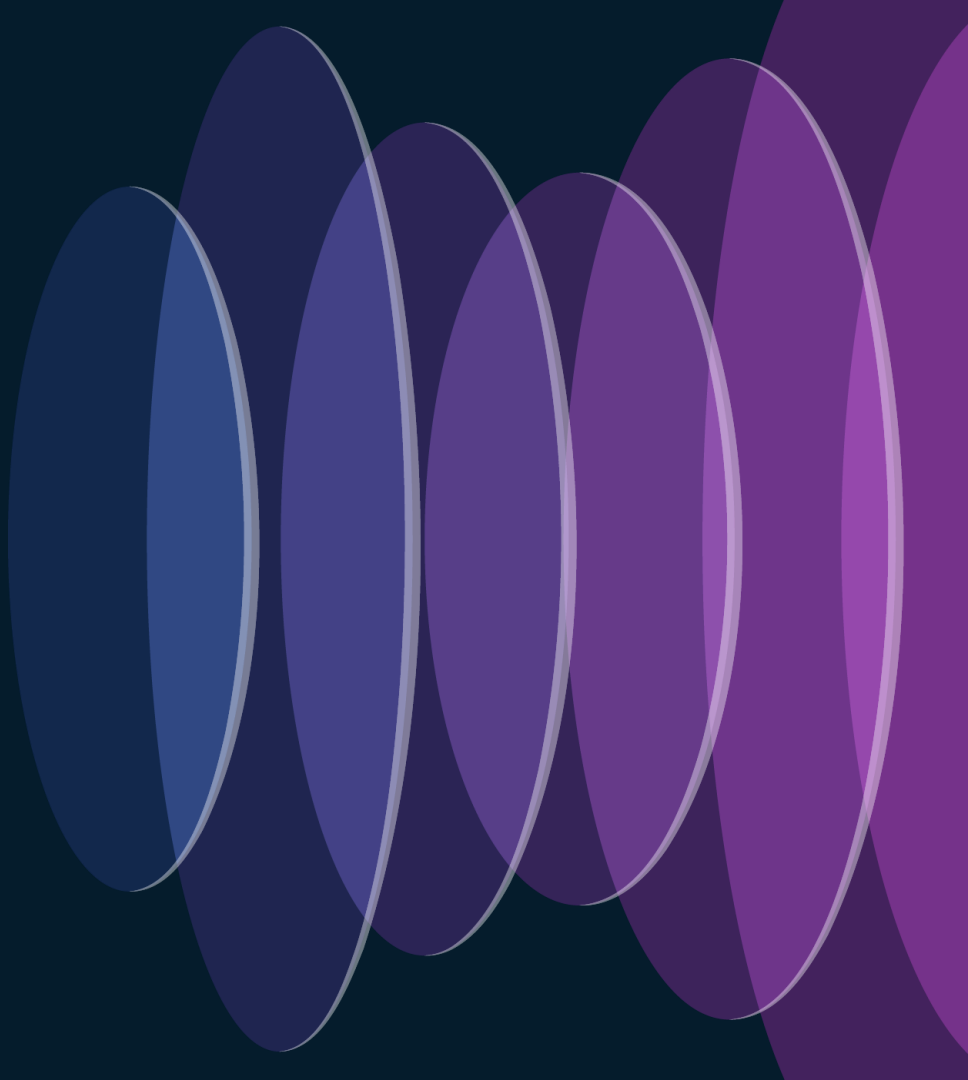
APIC 2 is Shard 2 Leader

APIC 2 has R1 of Shard 2

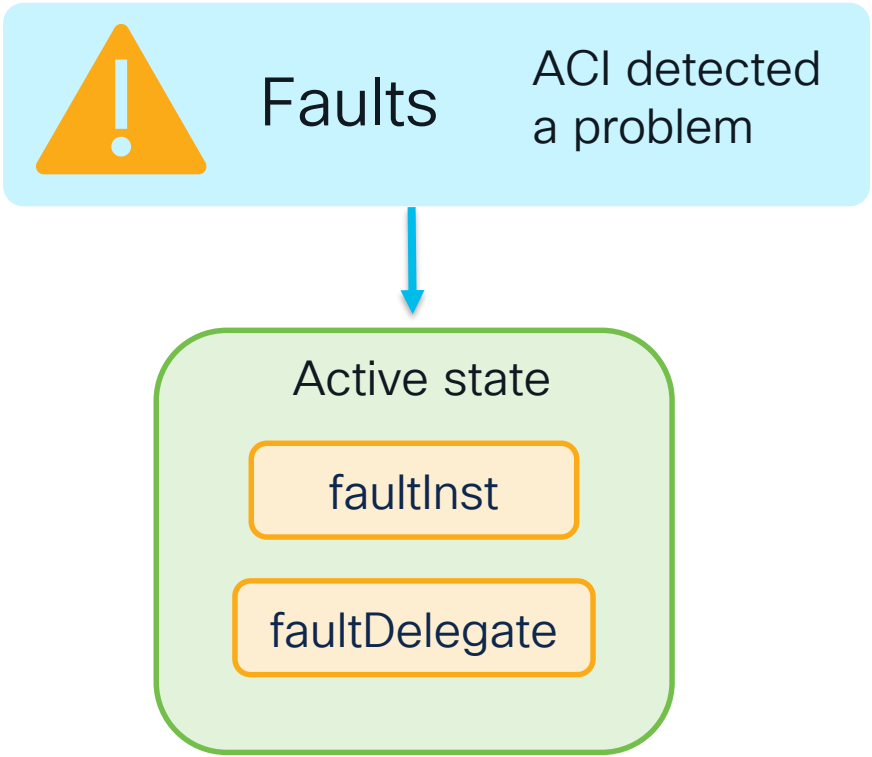
```
...
(6,31,1)    st:6 lm(t):1(2022-03-09T03:56:30.094+00:00) le: reSt:LEADER
(6,31,2)    st:6 lm(t):2(2022-03-09T03:48:28.880+00:00) le: reSt:FOLLOWER
(6,31,3)    st:6 lm(t):3(2022-03-09T04:21:04.214+00:00) le: reSt:FOLLOWER
(6,32,1)    st:6 lm(t):2(2022-03-09T03:56:36.575+00:00) le: reSt:LEADER
(6,32,2)    st:6 lm(t):3(2022-03-09T04:21:05.076+00:00) le: reSt:FOLLOWER
(6,32,3)    st:6 lm(t):1(2022-03-09T03:20:05.416+00:00) le: reSt:FOLLOWER
----------------------------------------------
clusterTime=<diff=-1099289 common=2022-04-15T21:50:54.647+00:00 local=2022-04-15T22:09:13.936+00:00
pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2020-01-08T20:53:53.081+00:00)>>
```

Summary of Cluster States

API Troubleshooting Tools



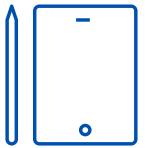
APIC Cluster Health and Verification



*Requires Admin Password

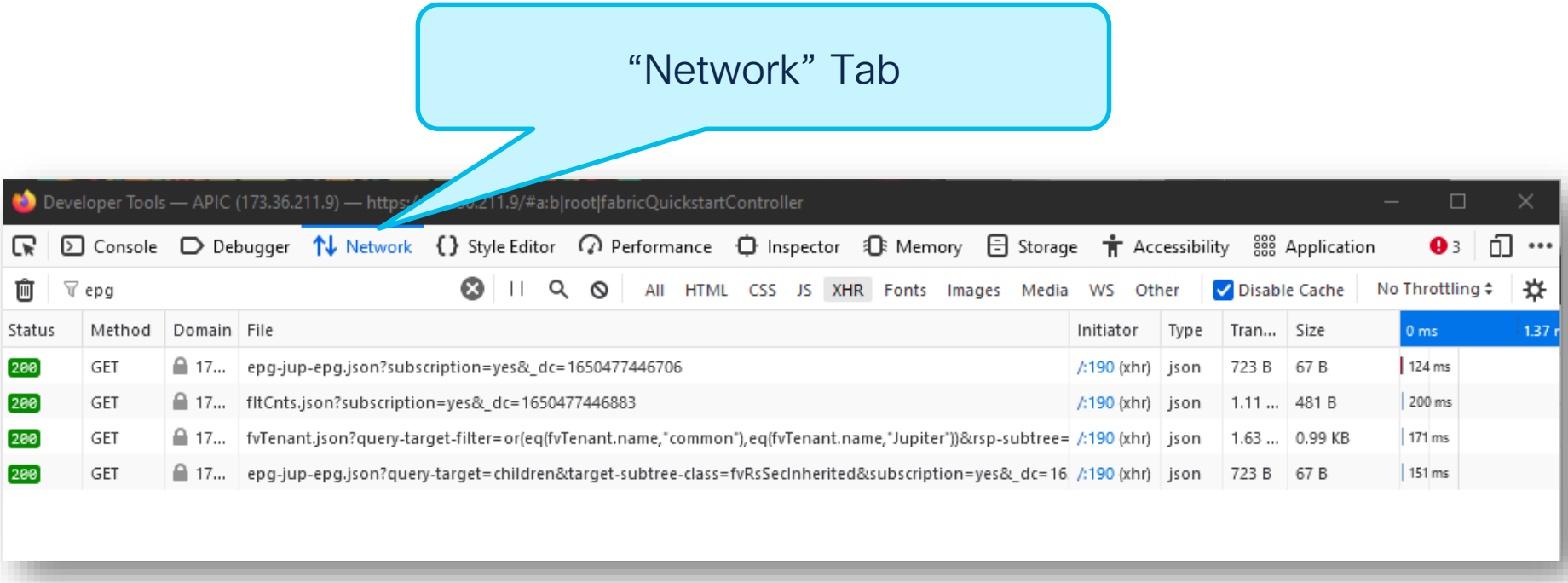
```
apic# acidiag cluster
Admin password:
Checking Wiring and UUID: OK
Checking AD Processes: Running
Checking All Apics in Commission State: OK
Checking All Apics in Active State: OK
Checking Shard Convergence: OK
Checking Leadership: Optimal leaders

Ping OOB IPs:
APIC-1: 192.168.1.1 - OK
APIC-2: 192.168.1.2 - OK
APIC-3: 192.168.1.3 - OK

Ping Infra IPs:
APIC-1: 10.0.0.1 - OK
APIC-2: 10.0.0.2 - OK
APIC-3: 10.0.0.3 - OK

Checking APIC Versions: Same (5.2(4d))
```

acidiag cluster - APICs

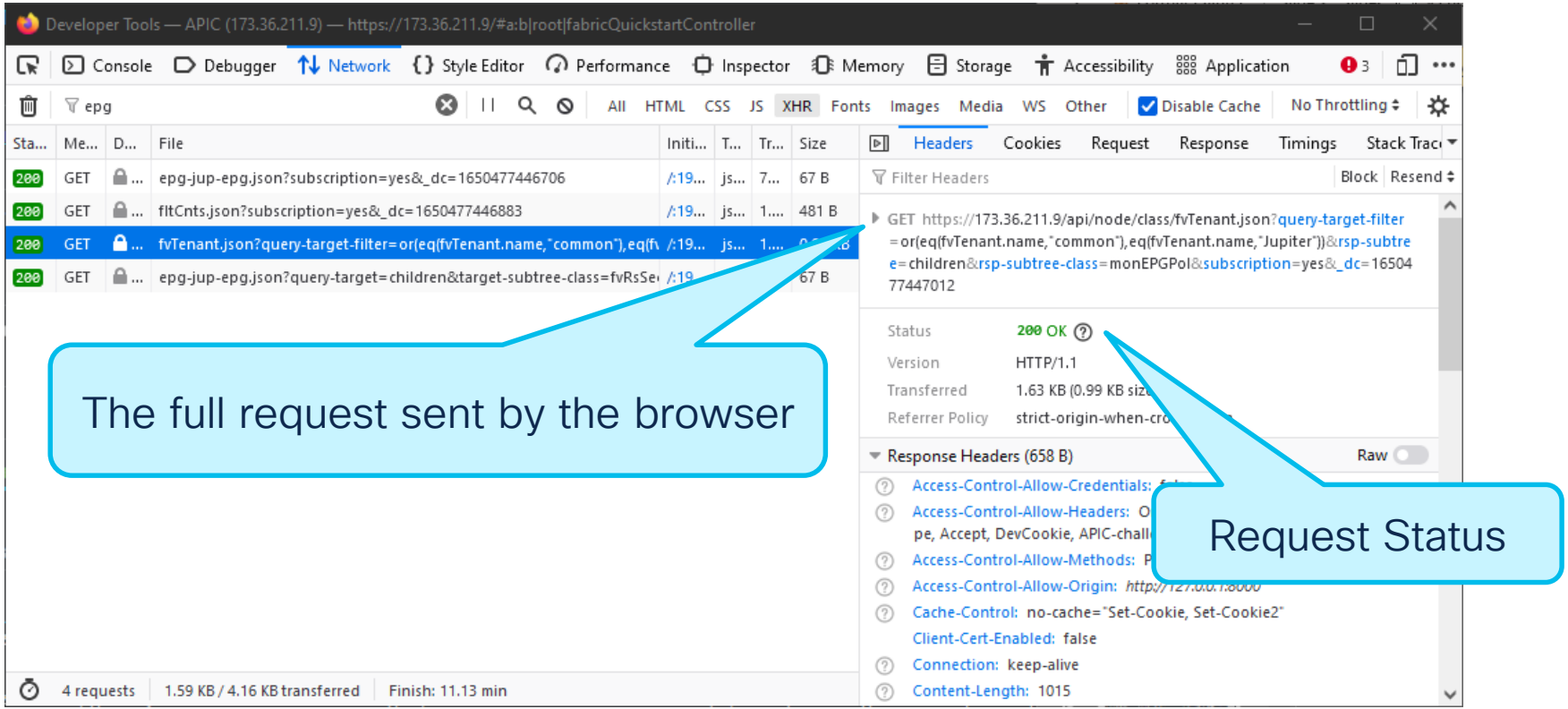


```
apic# acidiag cluster
Admin password:

Checking Wiring and UUID: OK
Checking AD Processes: Running
Checking All Apics in Commission State: OK
Checking All Apics in Active State: OK
Checking Fabric Nodes: OK
Checking Apic Fully-Fit: OK
Checking Shard Convergence: OK
Checking Leadership Degration: Optimal leader for all shards
Ping OOB IPs:
APIC-1: 172.21.208.154 - OK
APIC-2: 172.21.208.155 - OK
APIC-3: 172.21.208.156 - OK
Ping Infra IPs:
APIC-1: 10.0.0.1 - OK
APIC-2: 10.0.0.2 - OK
APIC-3: 10.0.0.3 - OK
Checking APIC Versions: Same (5.2(4d))
Checking SSL: OK
Full file system(s): None
```

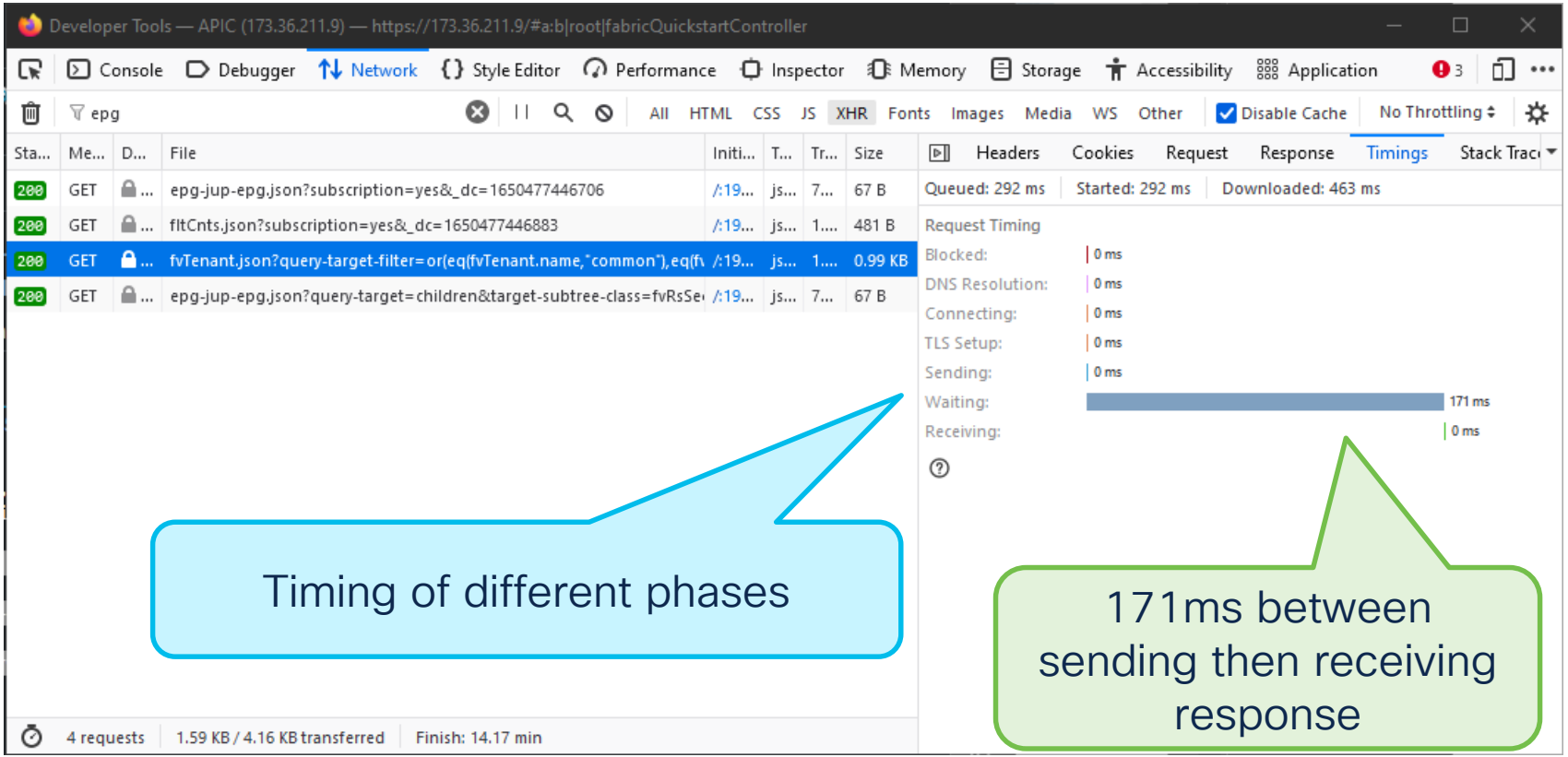
From the Browser – Browser Dev Tools



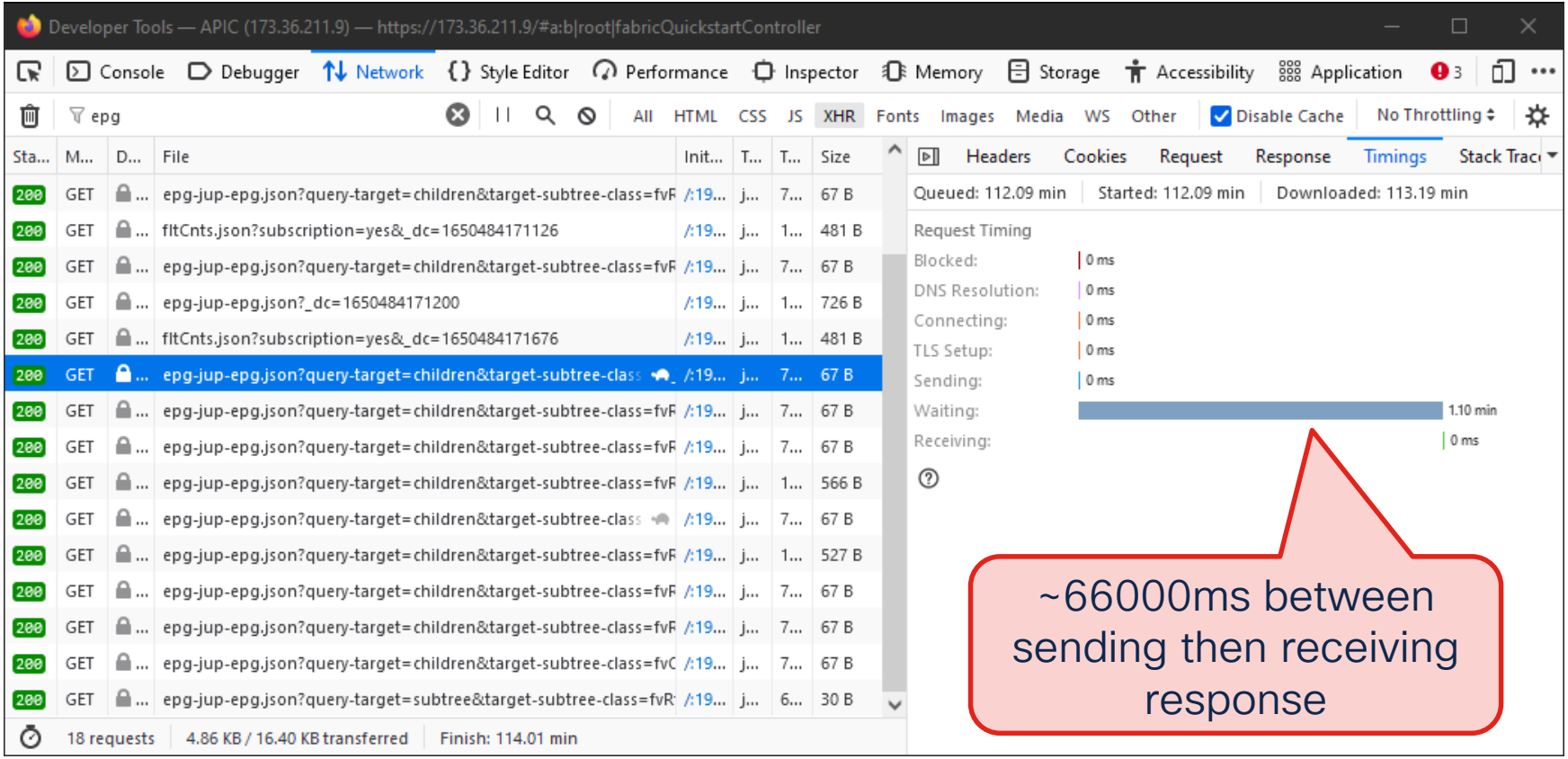
Browser Dev Tools – The Request



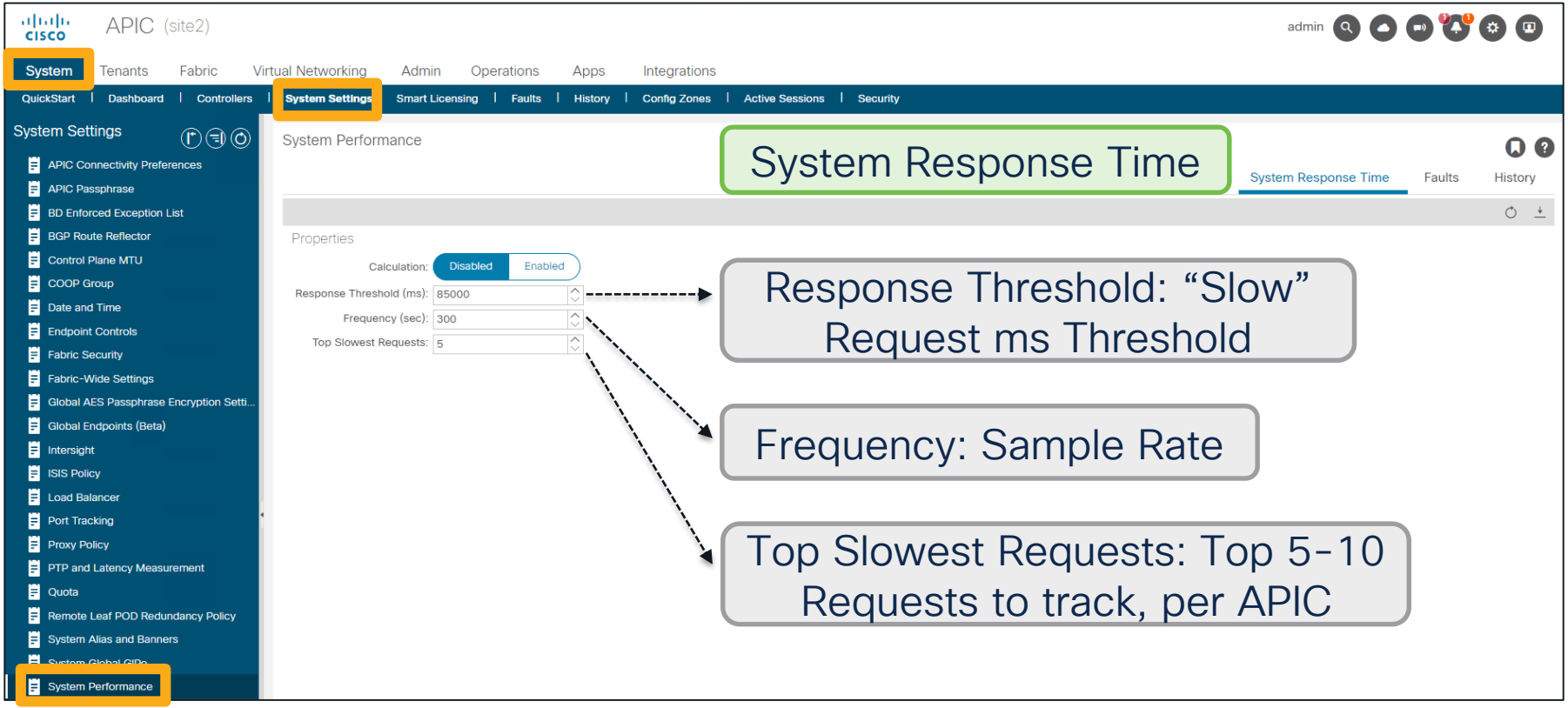
Browser Dev Tools – Timing



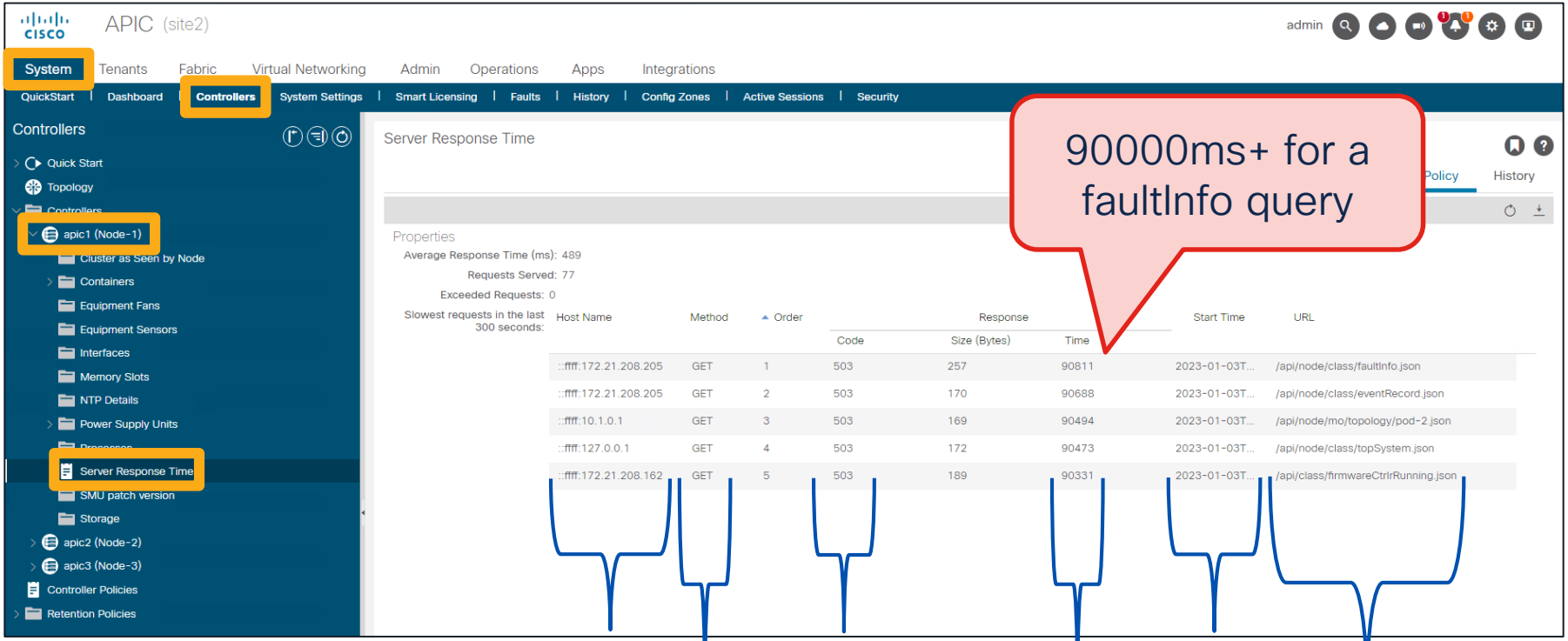
Browser Dev Tools – Slow APIC Response



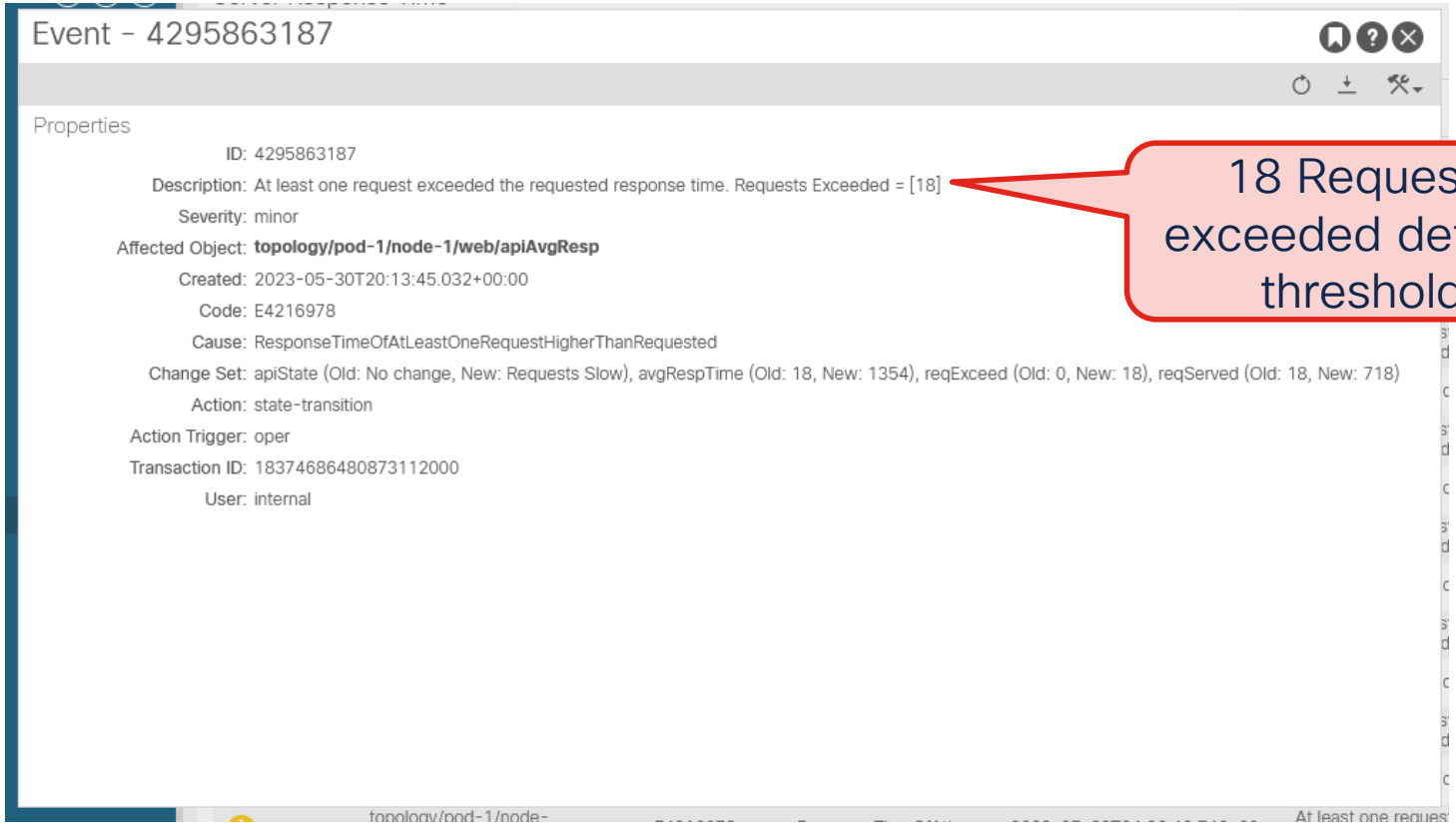
From the APIC – Check System Performance



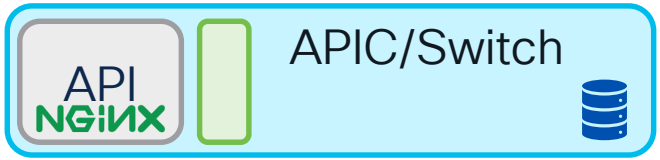
Server Response Time – View Slowest Requests



Server Response Time – Threshold Events



On the Filesystem – NGINX Includes access.log



Every Request through NGINX is logged

```
apic1# tail /var/log/dme/log/access.log
127.0.0.1 (-) - - [07/Apr/2022:20:10:59 +0000]"GET /api/class/topInfo.xml HTTP/1.1"     200 194 "-" "python-requests"
127.0.0.1 (-) - - [07/Apr/2022:20:10:59 +0000]"GET /api/class/topInfo.xml HTTP/1.1"     200 194 "-" "python-requests"
127.0.0.1 (-) - - [07/Apr/2022:20:10:59 +0000]"GET /api/class/topSystem.xml HTTP/1.1"   403 243 "-" "Python-urllib"
127.0.0.1 (-) - - [07/Apr/2022:20:10:59 +0000]"GET /api/class/fvTenant.xml HTTP/1.1"    200 15863 "-" "Python-urllib"
127.0.0.1 (-) - - [07/Apr/2022:20:11:00 +0000]"GET /api/class/topInfo.xml HTTP/1.1"     200 194 "-" "python-requests/"
```

Source IPs

Timestamp

HTTP Request Method

Request API Endpoint

Response Status Code

Client Library

CLI Commands behind-the-curtain

apic1# show audits details

```
apic# tail -f /var/log/dme/log/access.log | egrep decoy -A 20

"POST /decoy/exec/cmd.cli..."                                    ... "python-requests/2.22.0"
"GET /api/class/aaaModLR.xml?page-size=100&page=0&order-by=..."  ... "python-requests/2.22.0"
"GET /api/class/aaaModLR.xml?page-size=100&page=1&order-by=..."  ... "python-requests/2.22.0"
```

Python!

leaf# show vlan extended

```
switch# tail -f /var/log/dme/log/access.log | egrep urllib

"GET /api//class/l2Dom.xml?query-target-filter=..."                  ... "Python-urllib/3.7"
"GET /api/mo//sys/inst-overlay-1/bd-[vxlan-16777209].xml..."         ... "Python-urllib/3.7"
"GET /api/mo//sys/ctx-[vxlan-2621441]/bd-[vxlan-16318378].xml..."    ... "Python-urllib/3.7"

...many more GETs to all your VLANs...
```

Access Log Analyzer

http://cs.co/Access_Log_Analyzer

```
apic1# python /tmp/accLogAnalyzer.py
apic1# cat acclogAnalysis_2022-06-07T17:50:27.output

Access Log Time Analysis Summary:
    Total # of Requests: 46669

    Time Coverage: 109.416666667 Minutes (6565.0 s)
    Avg # of Reqs: 7.11 Queries per second

    Burst Summary: 602 15+ requests-per-second bursts found.
Remote Address Summary:
    Remote addr '127.0.0.1' request count: 10811 (%23.17 of total reqs)
    Remote addr '192.168.2.32' request count: 28431 (%60.92 of total reqs)
    ...

User-Agent Summary:
    User-agent 'Mozilla/5.0 (Wi...' request count: 3398 (%7.28 of total reqs)
    User-agent 'python-requests...' request count: 29052 (%62.25 of total reqs)
    ...

Response Status Summary:
    Response code '200' count: 20008 (%42.87 of total reqs)
    Response code '403' count: 17036 (%36.5 of total reqs)
    ...
```

50k queries over ~2 hours

Who?

What?

Why?

On the OS – Trusty ‘top’ and ‘ps’

top

```
apic# top
...
    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
1310506 root      20   0   10.1g   9.4g 279380 S  10.7   7.5   5575:24 nginx.bin
...
```

What is your APICs NGINX running at Steady State?

ps

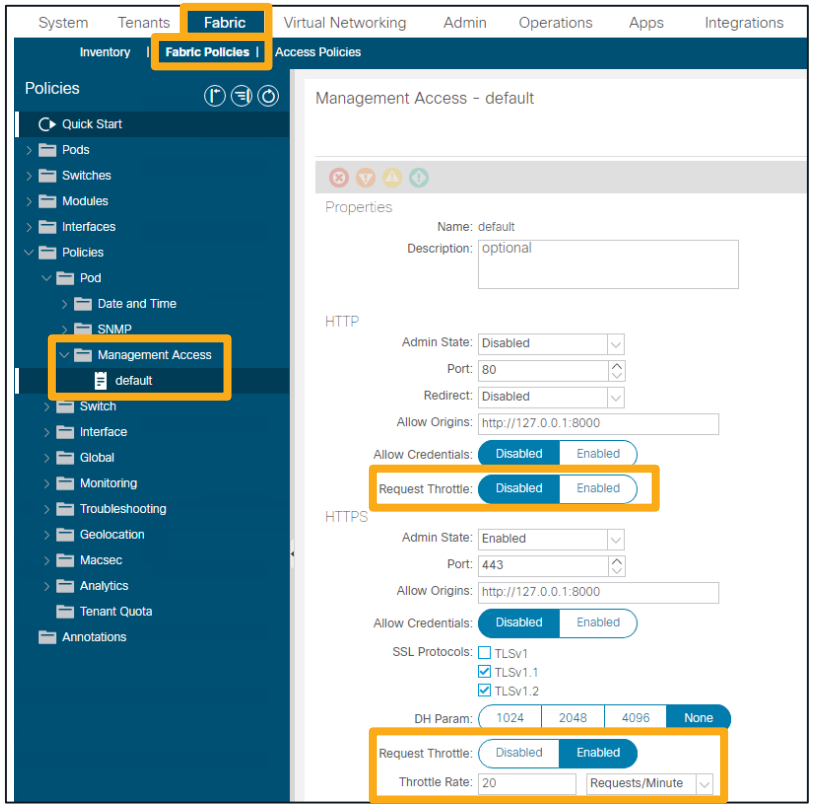
```
apic# ps -ef | grep show
admin ... /.../bin/python3 -m pyclient.remote exec server 173 show audits
```

admin ran ‘show audits’

```
leaf101# ps -ef | egrep show
admin    ... /usr/bin/python3 /controller/bin/show vlan extended
```

admin ran ‘show vlan extended’

Last Resort – NGINX Request Throttle



- Relies on NGINX Rate Limiting
- Set Throttle Rate in R/M or R/S
- Track per client IP address
- Does not affect Self (UI + CLI)
- Burst of Rate x 2 + NoDelay
- Threshold cross = 503 response

Throttled Requests Receive 503 Response

```
apic1# tail /var/log/dme/log/access.log
192.168.1.20 (-) - - [date]"GET /api/class/eventRecord.xml... HTTP/1.1" 200 494 "-" "python..."
192.168.1.20 (-) - - [date]"GET /api/class/eventRecord.xml... HTTP/1.1" 503 494 "-" "python..."
192.168.1.20 (-) - - [date]"GET /api/class/eventRecord.xml... HTTP/1.1" 503 494 "-" "python..."
```

503s follow successful responses

```
apic1# tail /var/log/dme/log/error.log
...limiting requests, excess: 40.292 by zone "httpsClientTagZone", client: 192.168.1.20, request: "GET
/api/class/...“

2023/04/17 20:19:14 [error] ... limiting requests, excess: 40.292 by zone "httpsClientTagZone",
client: h.o.s.t, ... request: "GET /api/node/...", host: "a.p.i.c"
```

Focus on addressing the source of the high rate of requests

Common Response Codes and Reasons

200 — All good

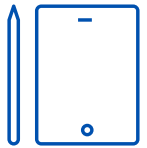
4xx

Response Code	Possible Reasons
400	Policy Distributor rule hit; Duplicate object, new mo requirement.
401	Authentication Issue; session expired, incorrect login Domain
403	API signing issue; incorrect signature calculation. Auth issue; Token timeout and new session required

503

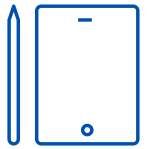
Response Code	Possible Reasons
Unable to deliver the message	Check that all APICs are fully-fit, check for DME token exhaustion
...temporary overload	NGINX throttle enabled, Request Throttled; check nginx error.log

Check acidiag manual for DME Name



```
apic# man acidiag
Service IDs:
       ...
       3 - eventmgr
       4 - extXMLApi
       5 - policyelem
       6 - policymgr
       7 - reader
       8 - ae
       9 - topomgr
       ...
       31 - opflexp
       32 - analytics
       33 - policydist
       34 - plgnhandler
       35 - domainmgr
       36 - licensemgr
```

Checking Tokens of DMEs

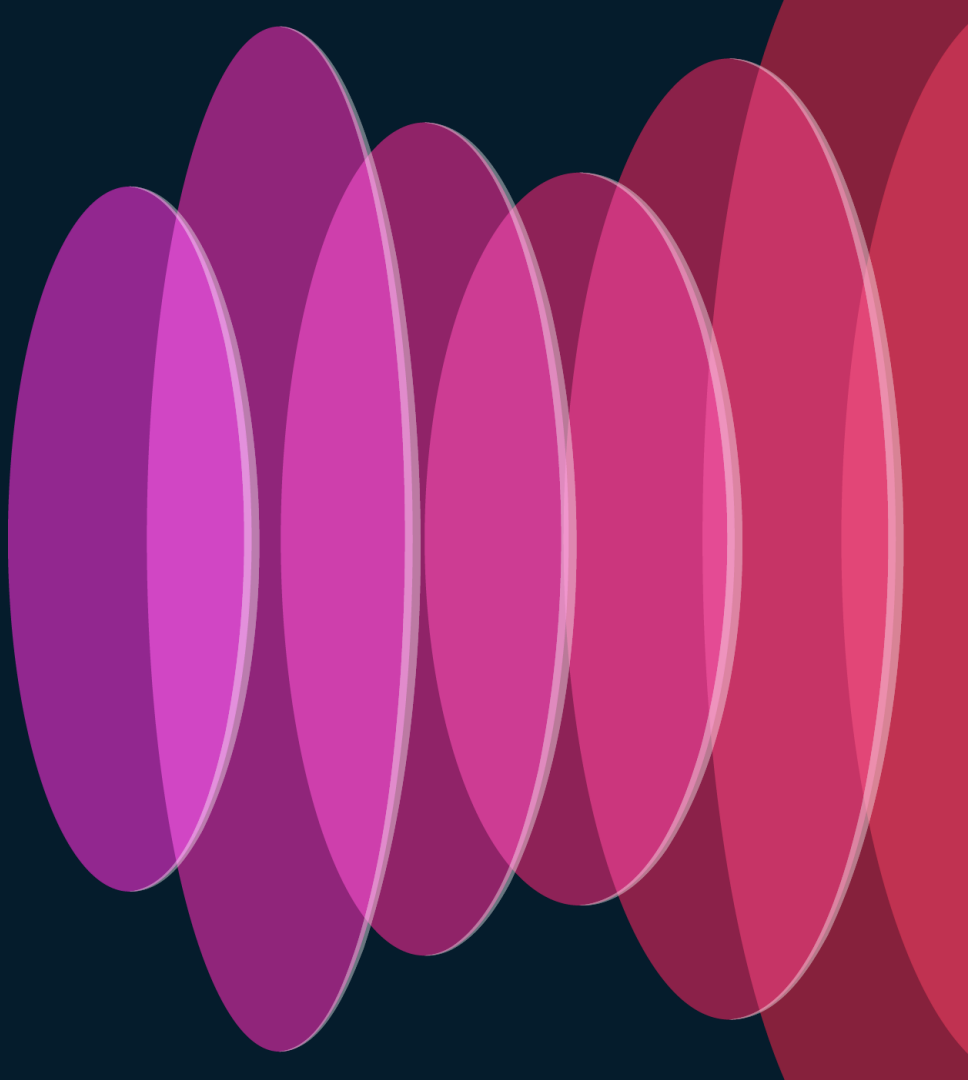


```
cat /debug/<apic-hostname>/<dme-name>/ifm/debug/mo
```

```
apic1# cat /debug/apic1/policymgr/ifm/debug/mo | egrep "id |tokens.available"
          id                : 0
          tokens.available : 1000

          id                : 1
          tokens.available : 902

          id                : 2
          tokens.available : 1000

          id                : 3
          tokens.available : 727

          id                : 4
          tokens.available : 852
          …
```

PM Shard 1 has 902/1000 tokens available

Common API Usage Issues



API Usage Issue – APIC GUI Experience is slow

Check Browser Dev tools

Slow

Check System Response Time

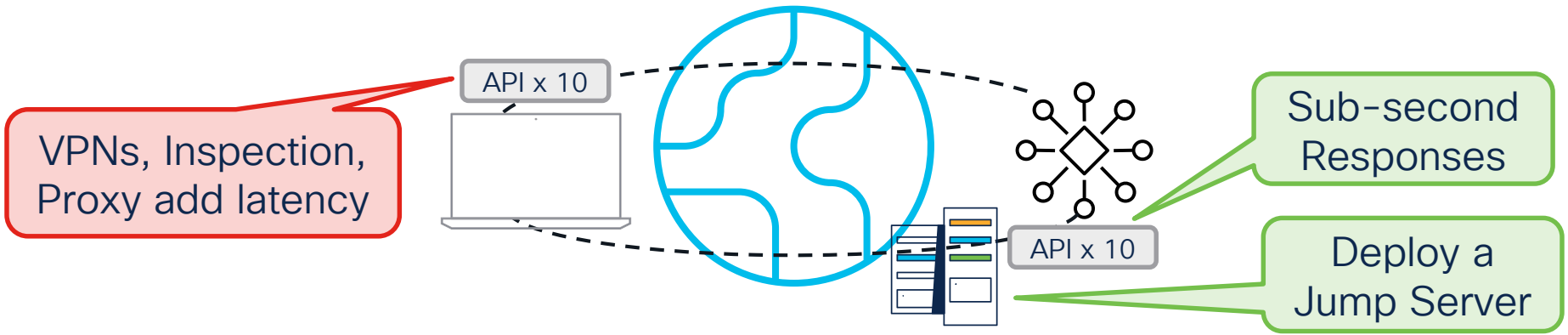
OK

Check 'top' for nginx

OK

Are all users affected?

Some



API Usage Issue – APIC API Experience is slow

Check Browser Dev tools	Check System Response Time	Check 'top' for nginx	Are all users affected?
Slow	Slow	High	All

Time to check AccessLog

```
apic1# top
...
    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
1310506 root      20   0   10.1g   9.4g 279380 S  170.7  7.5   5575:24 nginx.bin
...
```

...find the actor (IP/User/Script), then talk about APIC API usage optimization

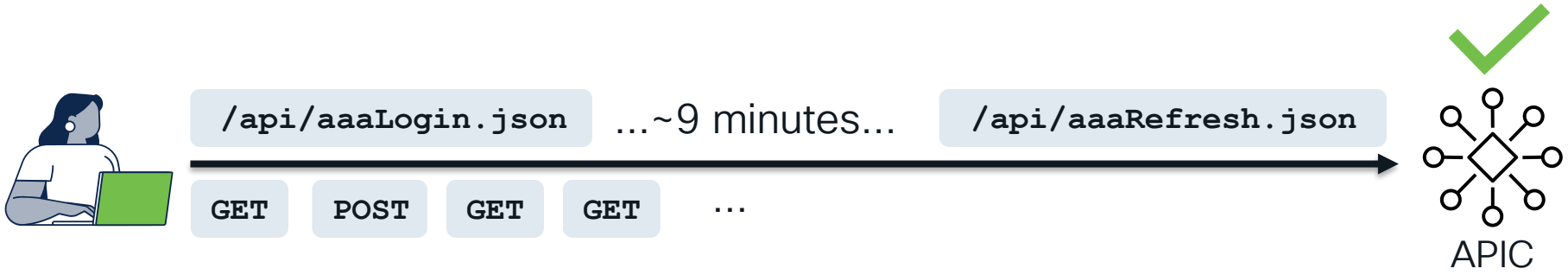
API Usage Issue – Logins per Request

Doubles rate of query

Each login spawns a new session

Login Rate of 2 Requests/second is the default non-configurable throttle

Optimization 1 – Login Session Refresh



A successful Login returns cookie:
imdata.aaaLogin.attributes.token

GET aaaRefresh with cookie:
Extends session by configured timeout

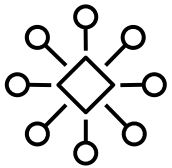
Optimization 2 – Signature-Based Transactions

`.key`

`cookies: ...`

`GET https://a.p.i.c/api/class/fvTenant.json`

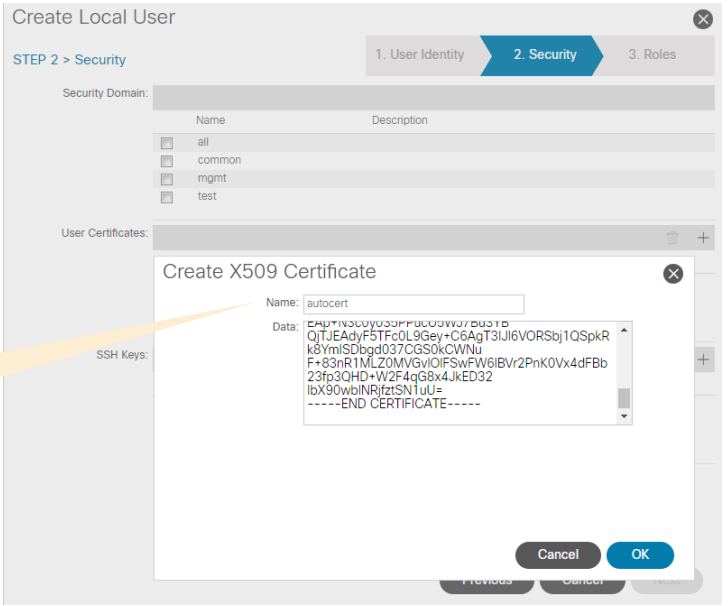
`.crt`



APIC

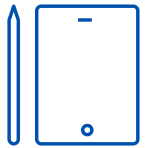
APIC-Request-Signature=**{signature}**;
APIC-Certificate-Algorithm=v1.0;
APIC-Certificate-Fingerprint=fingerprint;
APIC-Certificate-DN=user_cert_dn;

DN of user cert object:
uni/userext/user-automation/usercert-autocert



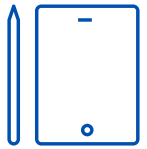
Optimization 2 cont – Calculating the Signature

Signed Request – CURL Example



```
# PREREQS: CRT/KEY generated. CRT added to APIC Local User.
# REQUEST: GET

# Prep Request Payload
echo -n 'GET/api/class/fvTenant.json' > payload.txt

# Use private key to generate signature
openssl dgst -sha256 -sign automation.key payload.txt > payload_sig.bin

# Convert signature to base64
openssl base64 -A -in payload_sig.bin -out payload_sig.base64

# Send the CURL with specific cookies including base64 signature and DN of APIC Local User CRT.
curl -k --cookie \
"APIC-Request-Signature=...base64_signature...\
APIC-Certificate-Algorithm=v1.0; \
APIC-Certificate-Fingerprint=fingerprint; \
APIC-Certificate-DN=uni/userext/user-automation/usercert-autocert" \
"https://a.p.i.c/api/class/fvTenant.json"
```

Signed Request – Python Example

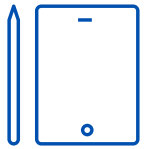


```
openssl req -new -newkey rsa:1024 -days 36500 -nodes -x509 -keyout automation.key -out automation.crt
-subj '/CN=Auto User/O=Cisco Live/C=US'
```

```python
from OpenSSL.crypto import FILETYPE_PEM, load_privatekey,
sign
import requests
import base64

APIC_ADDR = "https://a.p.i.c"
USERNAME = "automation"
CERT_MO = "autocert"

key_file_path = f"{USERNAME}.key"
cert_dn = f"uni/userext/user-{USERNAME}/usercert-
{CERT_MO}"

method = "GET"
api = "/api/class/fvTenant.json"
url = APIC_ADDR + api
```

```python
with open(key_file_path, "r") as f:
    key = f.read()

x509Key = load_privatekey(FILETYPE_PEM, key)
payload = method.encode("utf-8") + api.encode("utf-8")

signedDigest = sign(x509Key, payload, "sha256")
signature = base64.b64encode(signedDigest).decode("utf-
8")

cookies = {
    "APIC-Request-Signature": signature,
    "APIC-Certificate-Algorithm": "v1.0",
    "APIC-Certificate-Fingerprint": "fingerprint",
    "APIC-Certificate-DN": cert_dn,
}
r = requests.get(url, cookies=cookies, verify=False)
```

Track a Signed Request – nginx logs



```
apic1# egrep Signature /var/log/dme/log/nginx.bin.log

...snip...

5649||
2023-04-25T10:18:30.451349393-04:00||
nginx||
DBG4||
co=doer:255:127:0xff0000000a1c8c18:1||
Requested UserCert uni/userext/user-automation/usercert-autocert Fingerprint fingerprint
Signature VMtWCs3MU...Algorithm Version V1.0...
```

Signature from cookies of sent Request

Username for cert lookup + audit logging

Certificate Object to verify signature against

API Usage Issue – Requests by Use-Case

Use Case: I want to monitor Node 101 Interface Stats

topology/**pod-1**/**node-101**/sys/**phys-[eth1/1]**/**dbgEtherStats**

DN root is **topology**, not **uni**

Pod ID and **Node ID** are both parent objects

Interface ID is a parent parameter

rmonEtherStats is the className of the MO in question

Optimization – The Flexibility of the API

5.2(6e)+

*300K

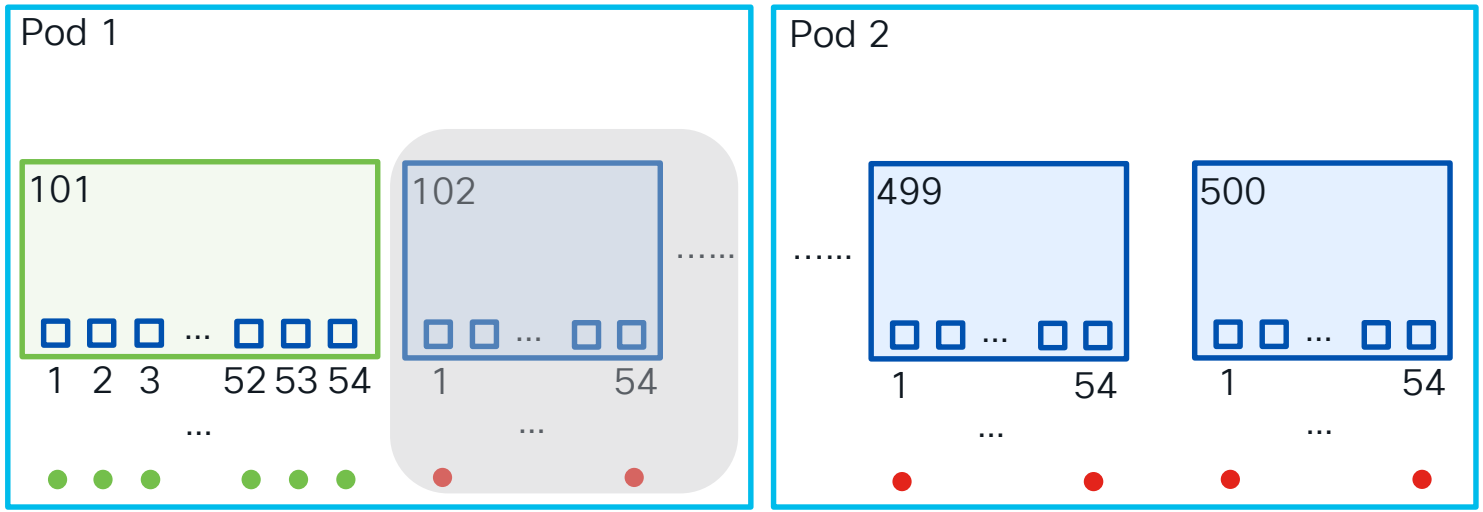
Use Case: I want to monitor Node 101 Interface Stats

Pod
Heavy Volume

Node
Just Right

Interface
High Count

rmonEtherStats
Heavy Volume



Goal: Use Query Options to Filter based on use case

100K* Response Object limit

Optimization – The Flexibility of the API

5.2(6e)+

*300K

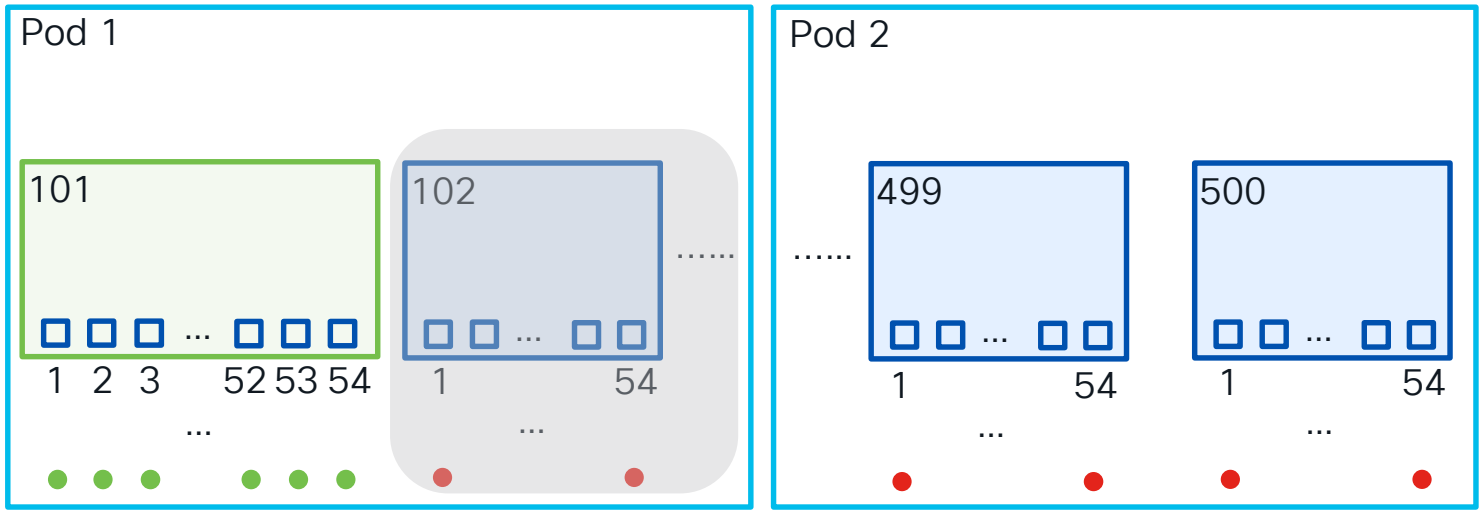
Use Case: I want to monitor Node 101 Interface Stats

Pod
Heavy Volume

Node
Just Right

Interface
High Count

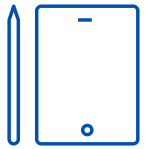
rmonEtherStats
Heavy Volume



Goal: Use Query Options to Filter based on use case

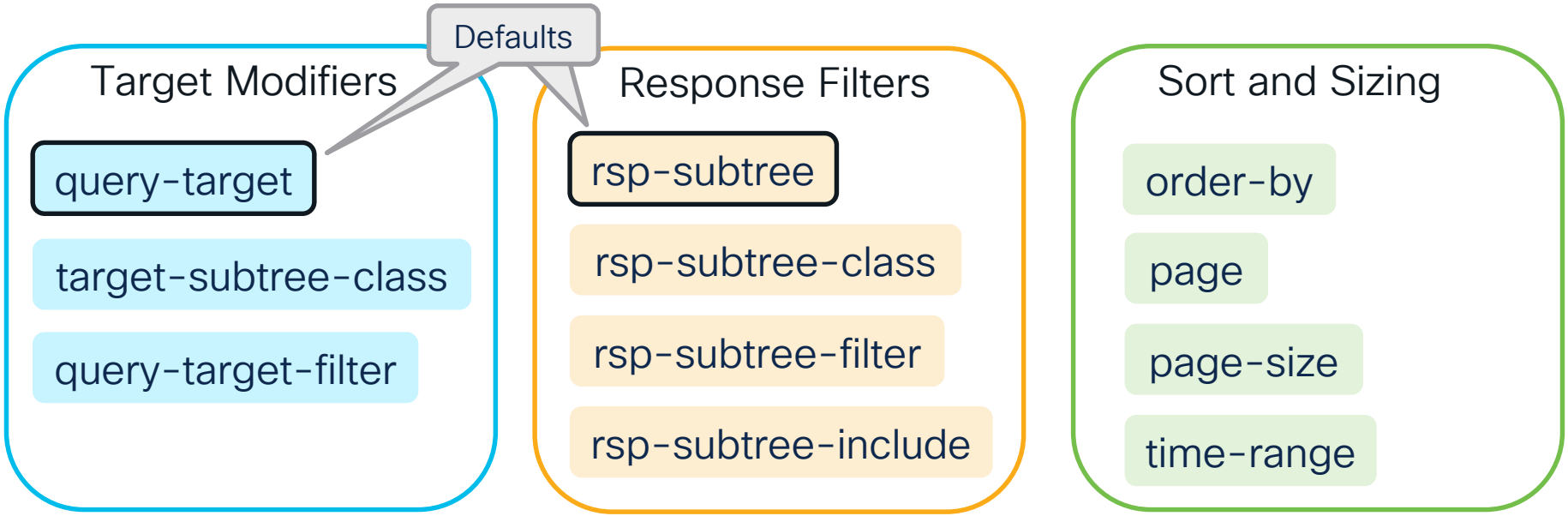
100K* Response Object limit

Interface Stats Object



```
apic# moquery -d 'topology/pod-1/node-101/sys/phys-[eth1/1]/dbgEtherStats'
{
    "totalCount": "1",
    "imdata": [
        {
            "rmonEtherStats": {
                "attributes": {
                    "broadcastPkts": "15964",
                    "cRCAlignErrors": "0",
                    "dn": "topology/pod-1/node-101/sys/phys-[eth1/1]/dbgEtherStats",
                    ...
                    "rXNoErrors": "46046",
                    "tXNoErrors": "73140",
                    ...
                }
            }
        }
    ]
}
```

Query Options Usage



`/api/mo/uni/tn-CL.xml`

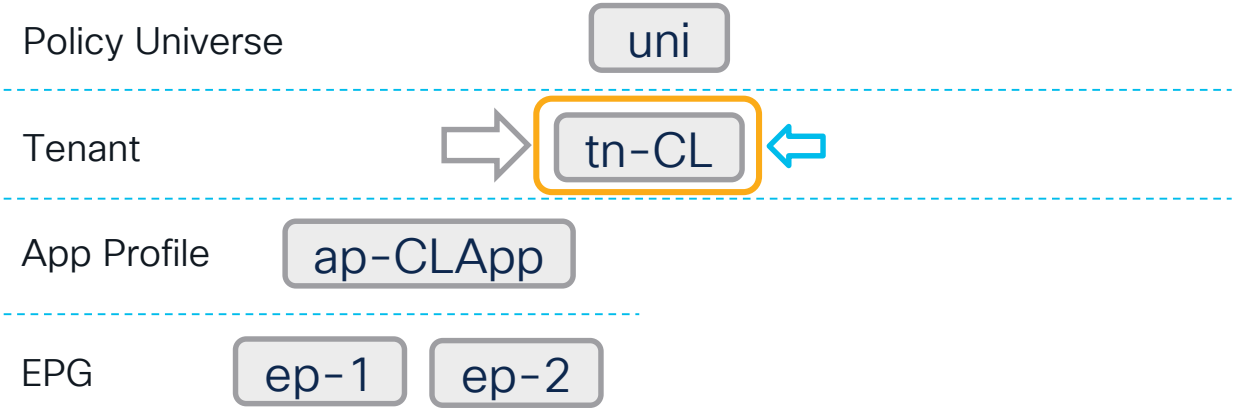
`/api/class/fvTenant.xml`

Defaults

`?query-target=self` `&rsp-subtree=no`

Options – Defaults

uni/tn-CL.xml ?query-target=**self** &rsp-subtree=**no**



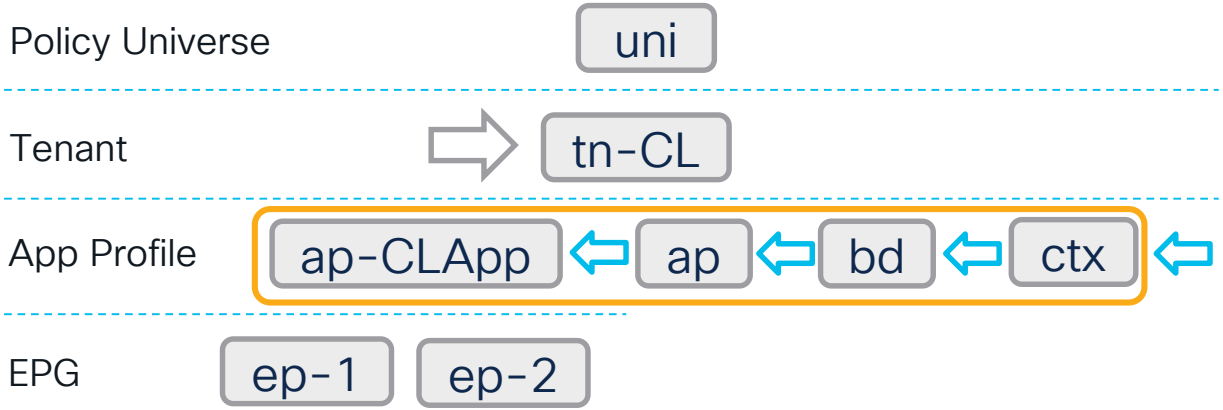
Response

- fvTenant

MOs returned: 1

Options – Query Target Children

uni/tn-CL.xml ?query-target=**children**



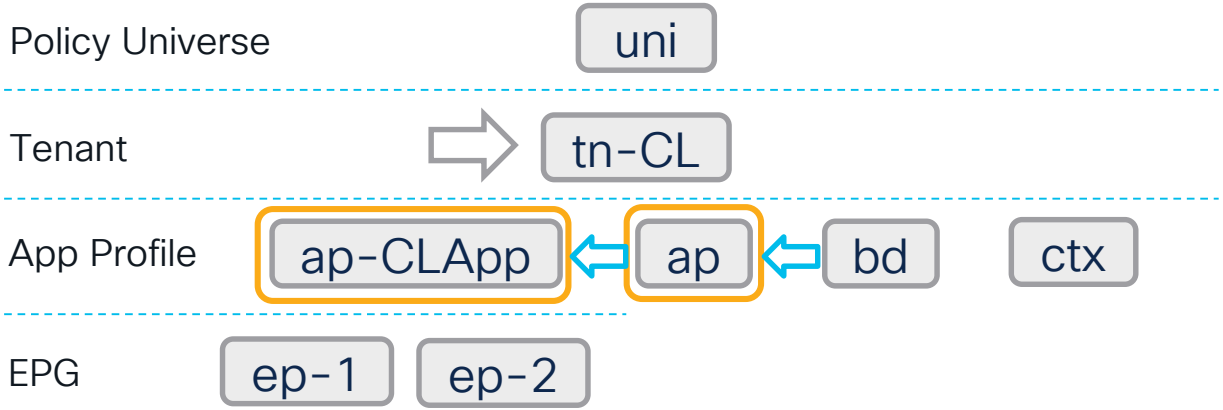
Response

- fvAp
- fvAp
- ...every other child
- **fvBd**
- **fvCtx**
- ...and so on

MOs returned: 2+

Options – Target Subtree Class

uni/tn-CL.xml ?query-target=**children** &target-subtree-class=**fvAp**



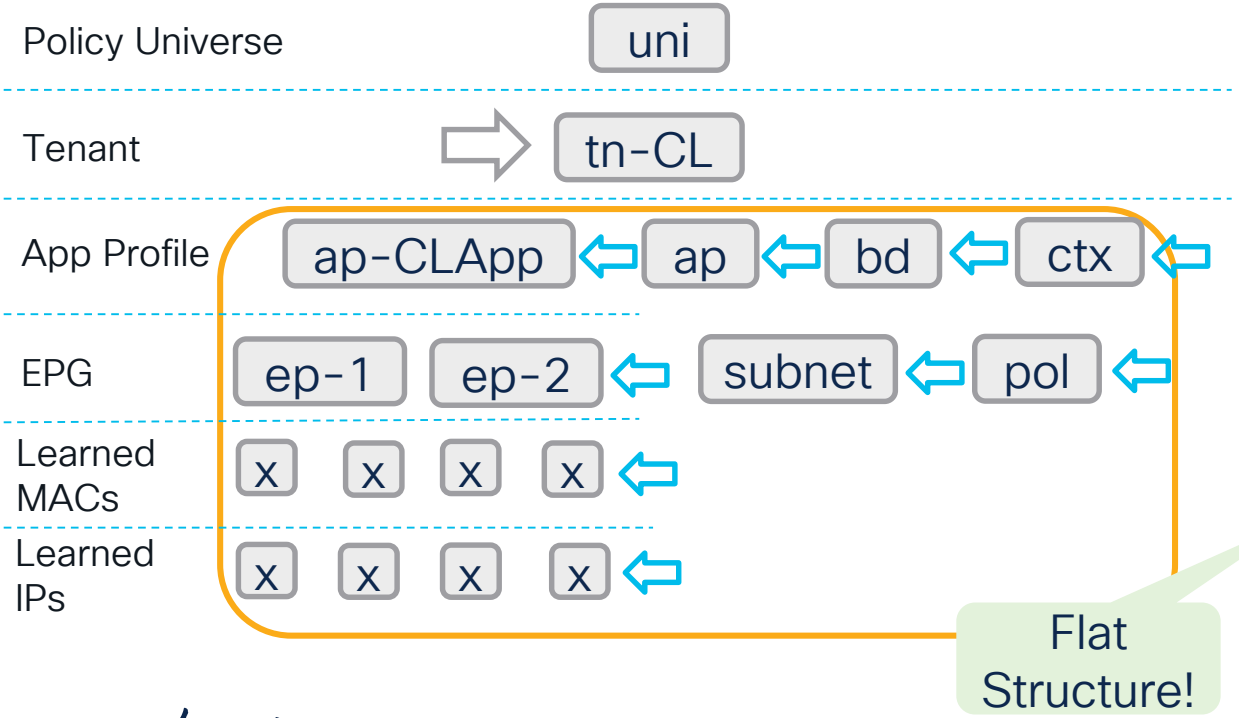
Response

- fvAp
- fvAp

MOs returned:
only child fvAp

Options – Query Target Subtree

uni/tn-CL.xml ?query-target=**subtree**



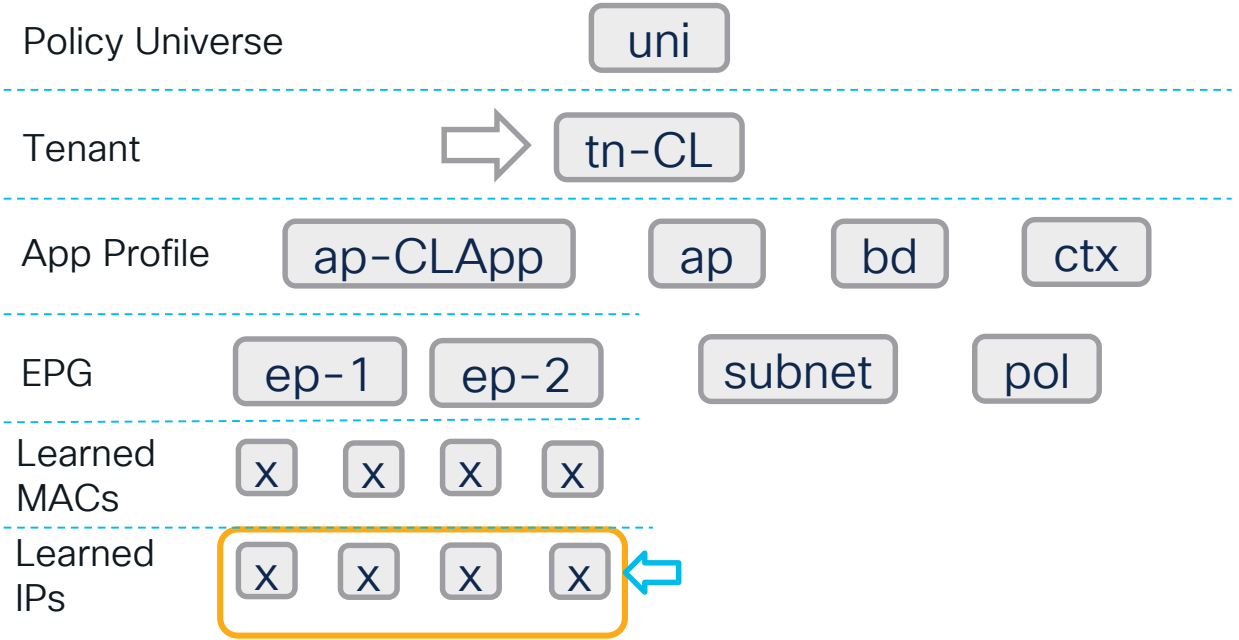
Response

- fvAp
- fvAp
- fvAEPg
- fvStPathAtt
- fvCEp
- fvIp
- ...and so on

MOs returned: x00+

Options – Query Target Subtree

uni/tn-CL.xml ?query-target=**subtree** &target-subtree-class=**fvIp**



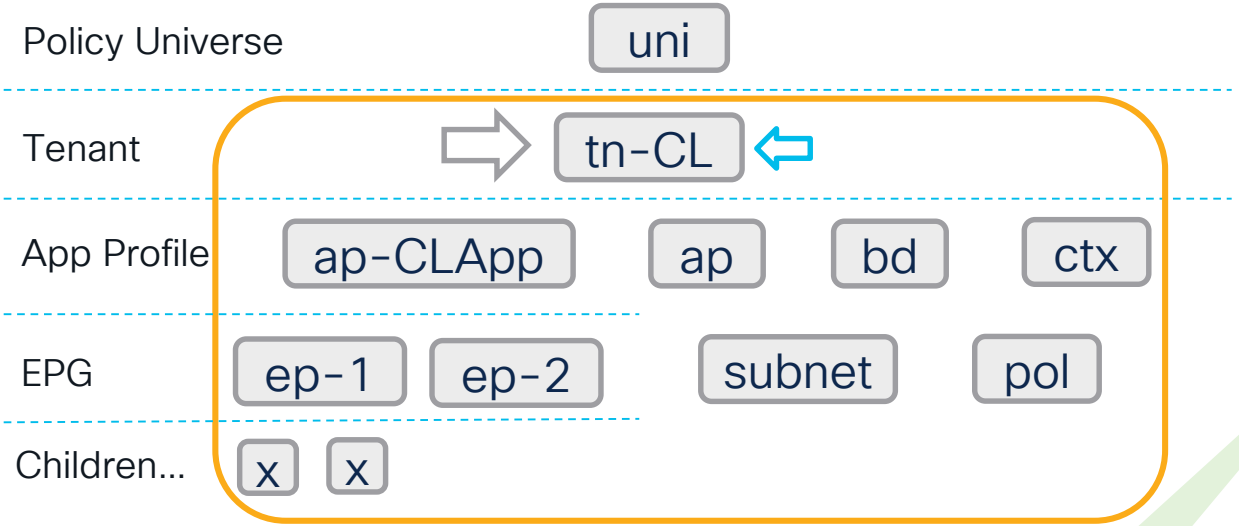
Response

- fvIp
- fvIp
- fvIp
- fvIp

MOs returned: 4

Options – Response Subtree

uni/tn-CL.xml ?query-target=**self** &rsp-subtree=**full**



Response

- fvTenant
  - **fvBD**
    - **fvSubnet**
  - **fvCtx**
  - fvAp
    - fvAEPg
      - fvCEp
  - ...

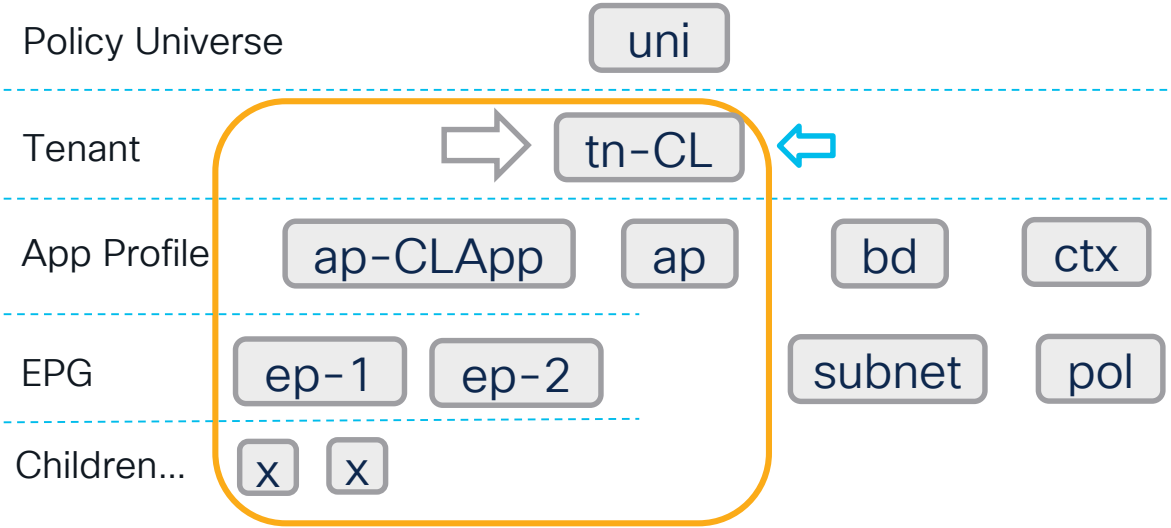
MOs returned:
1 parent, all branches

Nested structure!

Powerful when combined with a class query

Options – Response Subtree Class

uni/tn-CL.xml ?rsp-subtree=**full** &rsp-subtree-class=**fvAp**



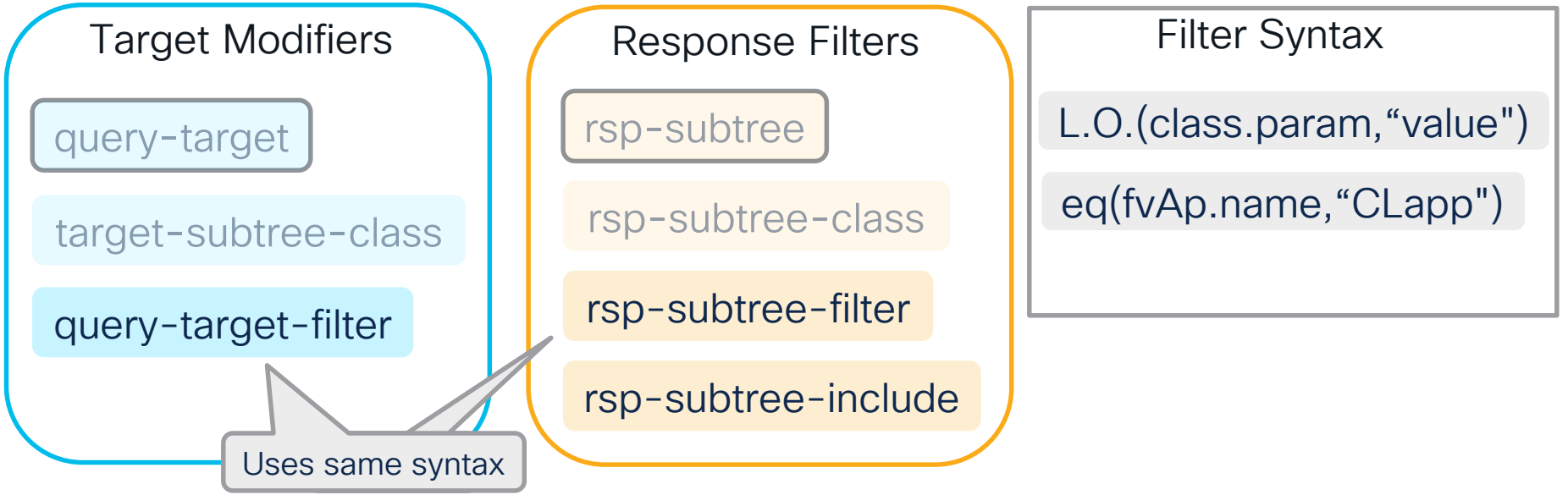
Focus on branches of the class(es) specified

Response

- fvTenant
  - fvAp
  - fvAp
    - fvAEPg
    - fvAEPg
      - fvCEp
  - ...and so on

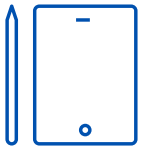
MOs returned:
1 parent, fvAp branch

Advanced Options – The Parameter Filters




```
class/fvTenant.xml ?query-target-filter=eq(fvTenant.name,"CL")
class/fvTenant.xml ?rsp-subtree-filter=eq(fvTenant.name,"CL")
```

Advanced Options – All Logical Operators



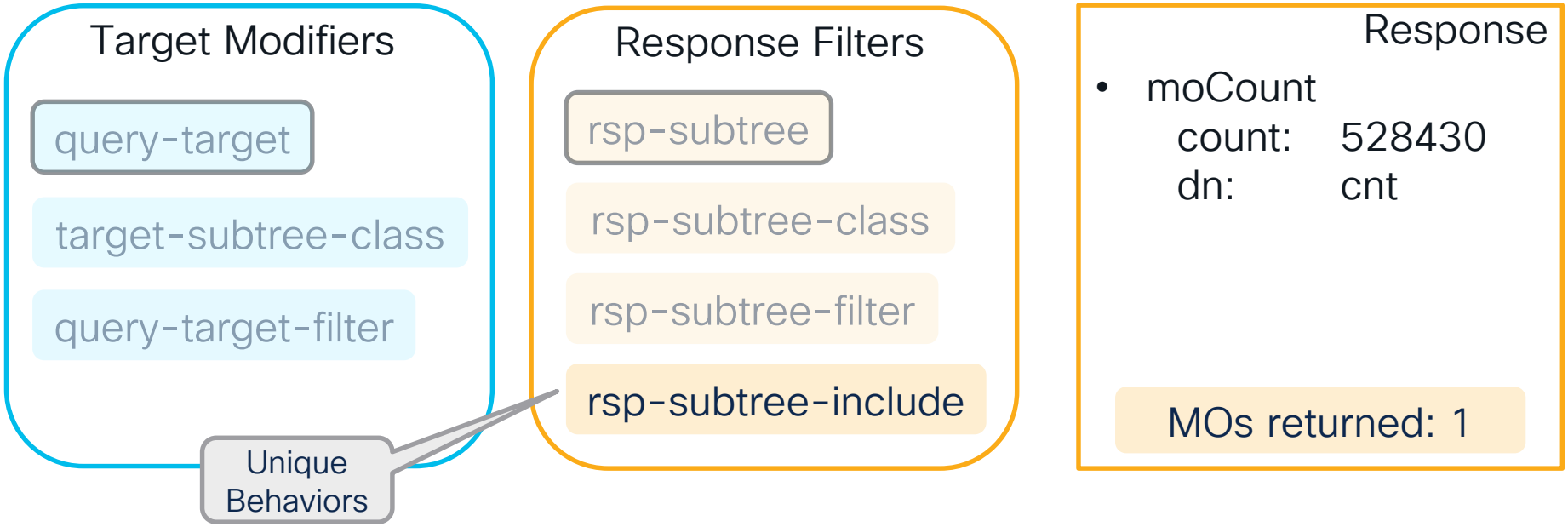
Logical Operator	Description
eq	Equal to
ne	Not equal to
lt	Less than
gt	Greater than
le	Less than or equal to
ge	Greater than or equal to
bw	Between
not	Logical inverse
and	Logical AND
or	Logical OR
xor	Logical exclusive OR
true	Boolean TRUE
false	Boolean FALSE
anybit	TRUE if at least one bit is set
allbits	TRUE if all bits are set
wcard	Wildcard
pholder	Property holder
passive	Passive holder

Multiple Filter Syntax

```
L.O.(class.param,“value")
```

```
and(
  ne(fvAp.name,“CLapp"),
  ne(fvAP.name,“TRapp")
)
```

Advanced Options – Response Subtree Include



class/eventRecord.xml ?rsp-subtree-include=**count**

Advanced Options – Response Subtree Include

Related MOs

Operator	Class
audit-logs	aaaModLR
event-logs	eventRecord
fault-records	faultRecord
health-records	healthRecord
count	moCount of top level

Child MOs

Operator	Class
faults	faultInst, faultDelegate
health	healthInst
relations	**many
stats	**many
tasks	**very low level

Response Options

Operator	Returns
subtree	Must be used with no-scope, pulls subtree into logic
no-scoped	Only return above ‘included’ objects, not the parents. Flattens the result.
required	Only return parent MO if ‘included’ child object exists.

Example Response Subtree Include Queries

Get Count of all MOs under all Tenants

```
class/fvTenant.xml
    ?query-target=subtree &rsp-subtree-include=count
```

Get ONLY audits for subtree of tn-CL

```
uni/tn-CL.xml ?rsp-subtree-include=audit-logs,subtree,no-scoped
```

Get BDs and children ONLY if it has a subnet

```
class/fvBD.xml ?rsp-subtree=children
    &rsp-subtree-class=fvSubnet&rsp-subtree-include=required
```

Advanced Options – Sort and Paginate

Sort and Sizing

- order-by
- page
- page-size
- time-range

Sort – ascending or descending

class/aaaModLR.xml

?order-by=**aaaUser.firstName|desc**

Paginate through chunks of response

class/aaaModLR.xml

?page=**0** &page-size=**1000**

High Churn Record objects may result in duplicates

Advanced Options – time-range Record Caching

Sort and Sizing

order-by

page

page-size

time-range 5.2(3)+

Freeze Rolling Records

class/eventRecord.xml

?page=**0** &page-size=**2000** &time-range=**24h**

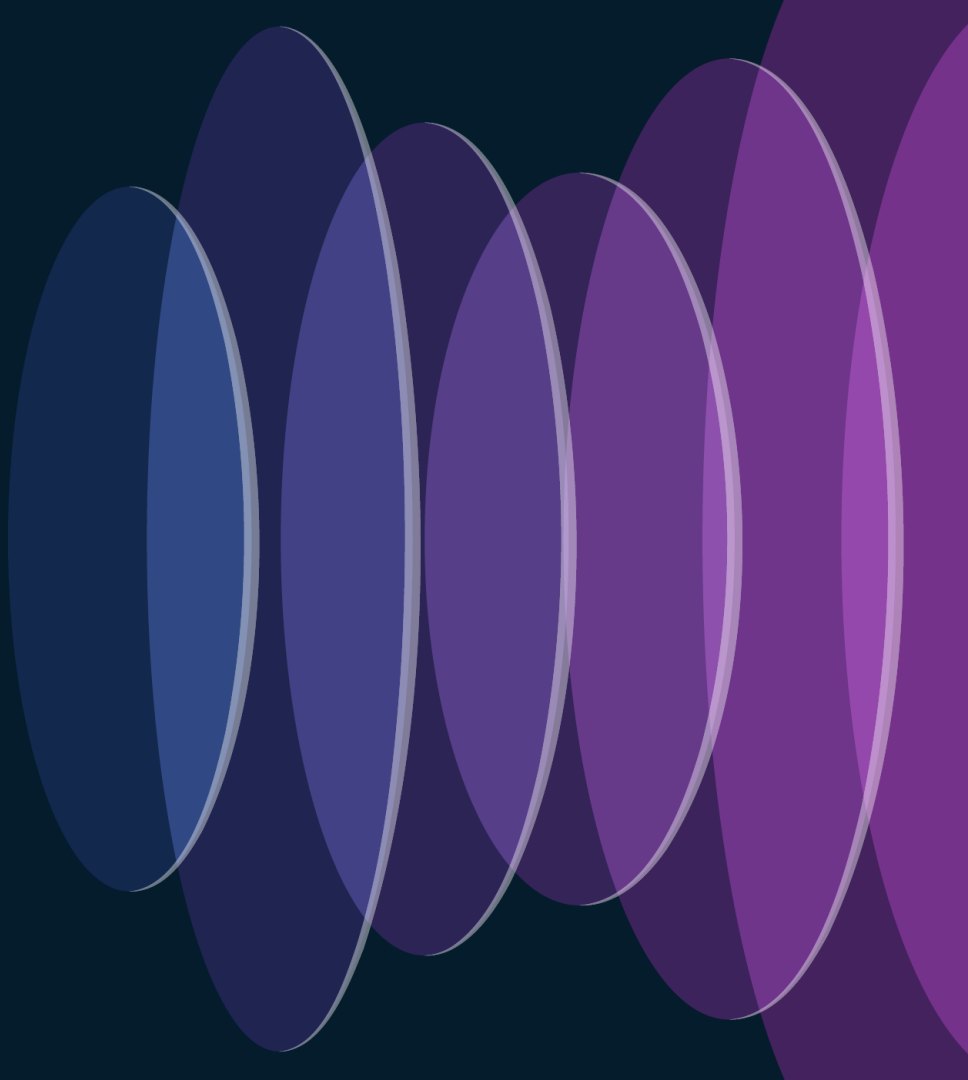
time-range units	syntax
hours	xx**h**
weeks	xx**week**
months	xx**month**
Range	yyyy-mm-dd|yyyy-mm-dd

Must start with page=0, then leaf through page+=1 to maintain cache

page-size limit of 2000

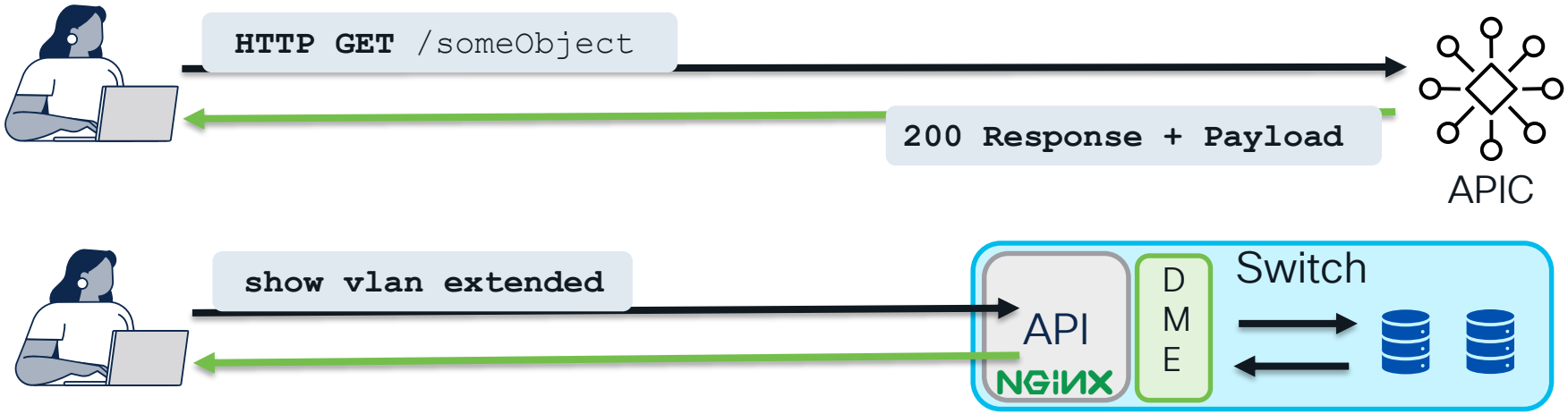
Auto-sorts on time

Optimize Visibility with Query Subscriptions



cisco Live!

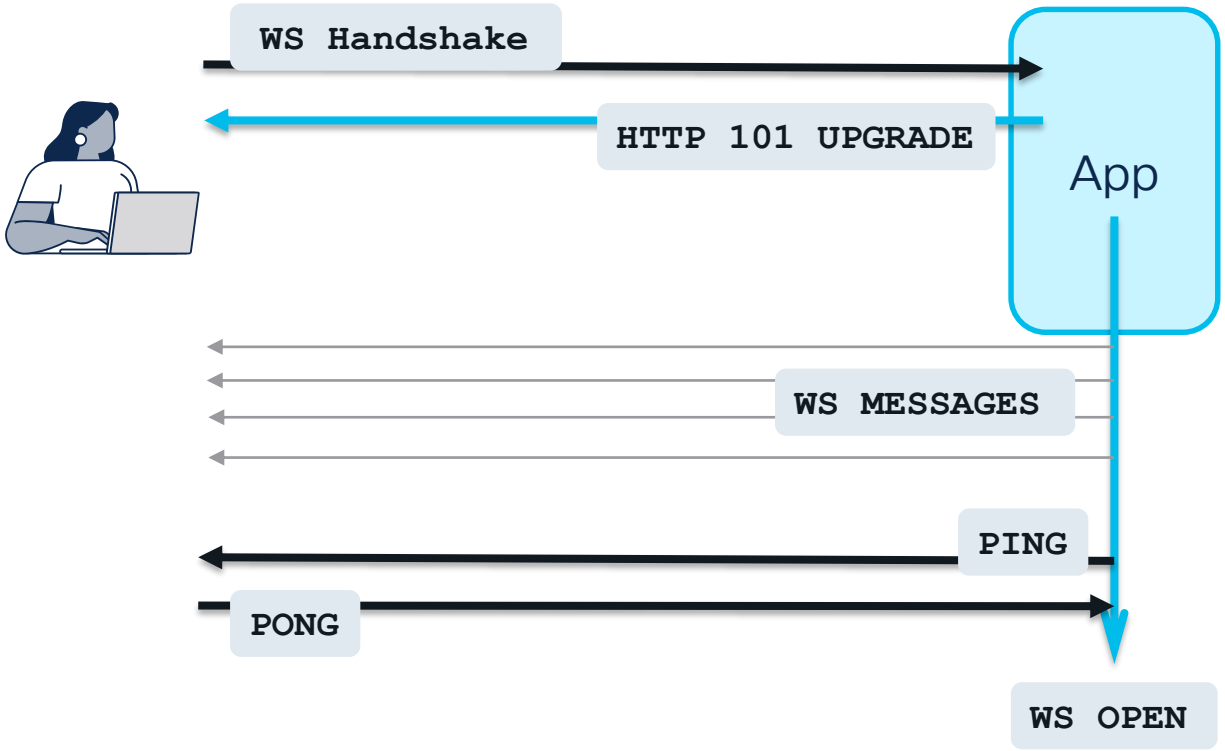
We are now the champions of Polling



Single transactional request, with single response

How do I track live MO changes without a timer?

Let’s talk WebSockets

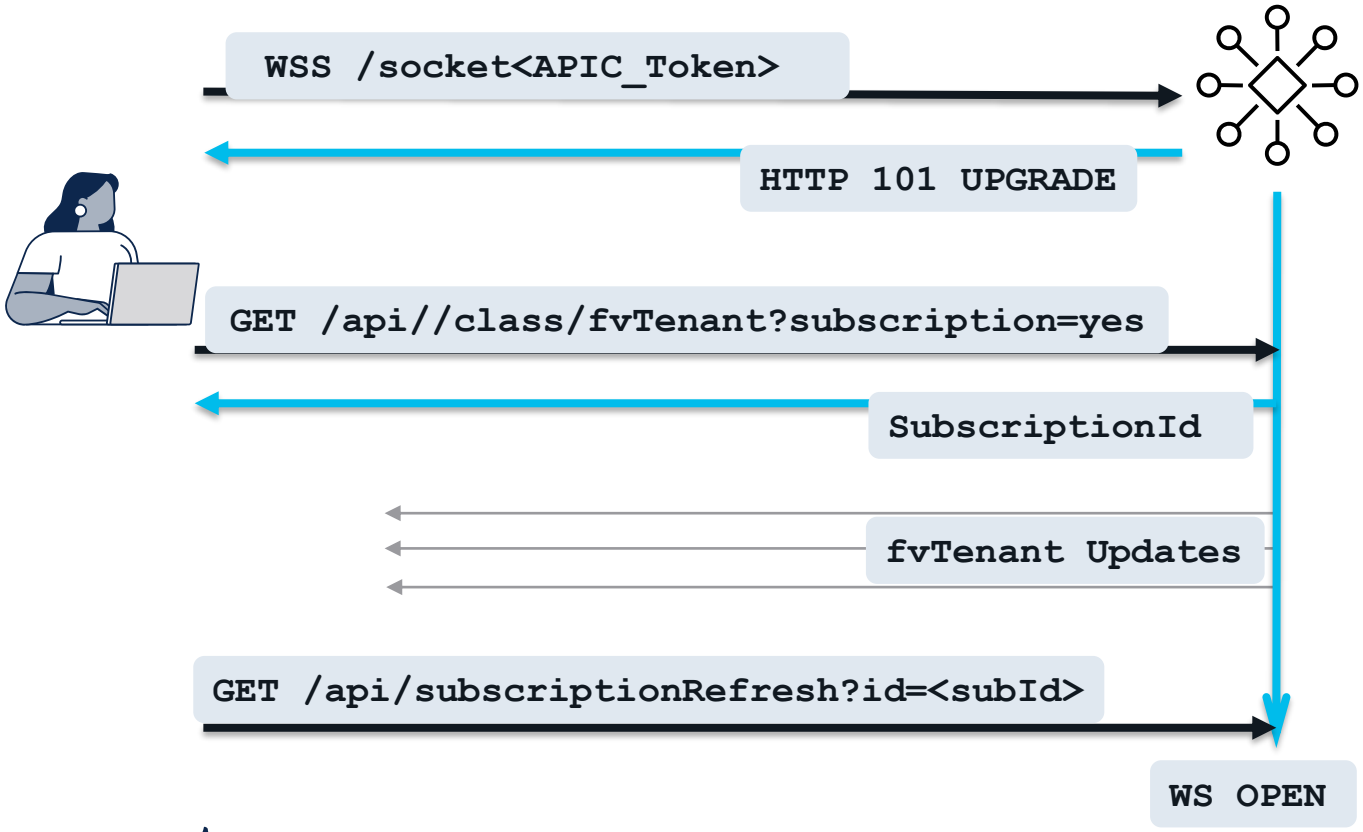


Handshake to OPEN channel

As App State changes, User receives Messages

Keepalives used to maintain OPEN

Subscribing to Query Results via WebSockets



Subscription is tied to Login Session

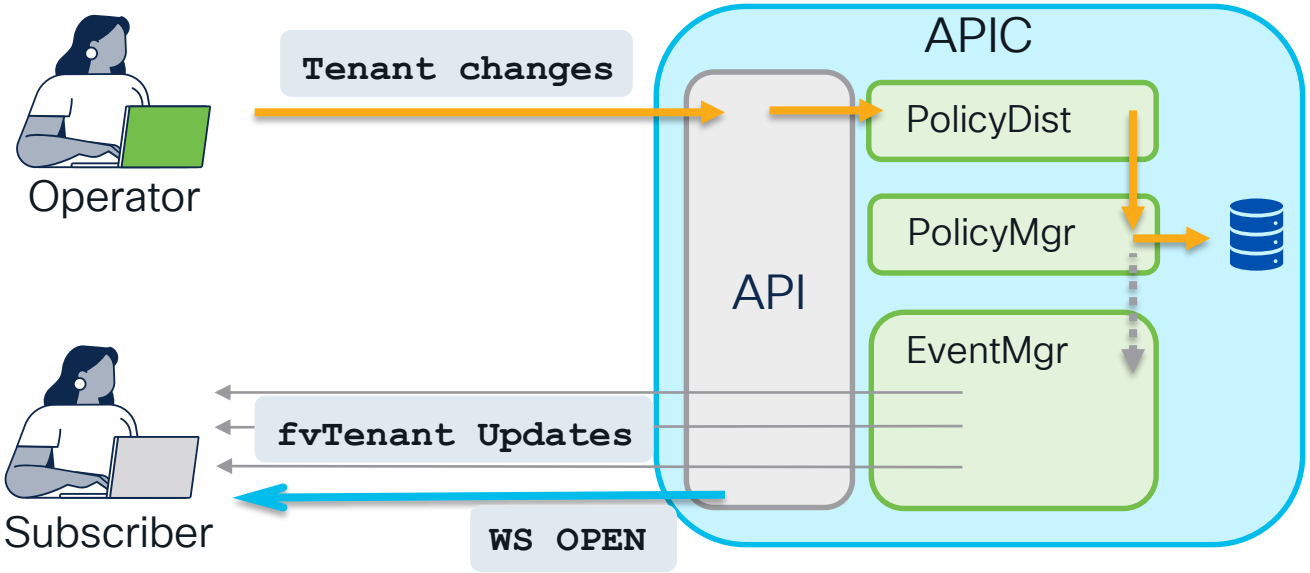
Open Socket with APIC Token

Send query with **subscription=yes**

Receive MO updates

Refresh Subscription with subId

Subscription - Under the hood



Receive MO events that go through eventMgr

Logical MO changes - explicit config changes

Record MO changes – audits, faults, events

No Stats MO changes – frequency intensive

Subscribable MOs subject to change: epRecord

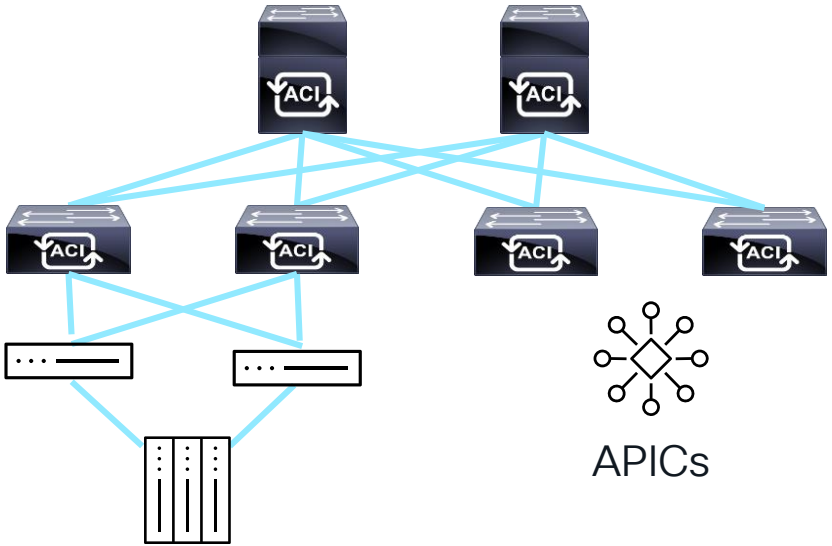
Query Subscription Notification Message

http://cs.co/APIC_Websocket_Starter

```
mypc$ python apic_query_subscription.py -a a.p.i.c -u gmonroy -x
gmonroy password:
********** WebSocket Subscription Status & Messages **********

https://a.p.i.c/api/class/fvTenant.xml
     ?subscription=yes&query-target=subtree&target-subtree-class=fvAEPg,fvBD
- Subscription ID: 72339464170766337

********** WebSocket Subscription Messages **********
...
<imdata subscriptionId="72339464170766337">
<fvAEPg ... dn="uni/tn-CiscoLive/ap-cl-ap/epg-CL_EPG"          tus="created" .../>

...
<imdata subscriptionId="72339464170766337">
<fvBD ... dn="uni/tn-CiscoLive/BD-cl-bd"... status="modified"/>
```

SubId in each message

Evolve from Optimization to Isolation



Consider MOs of importance going into a Maintenance Window

Ex: Subscribe to commRequestData

Build a baseline and track those MOs during maintenance

Complete Your Session Evaluations

Complete a minimum of 4 session surveys and the Overall Event Survey to be entered in a drawing to **win 1 of 5 full conference passes** to Cisco Live 2025.

Earn 100 points per survey completed and compete on the Cisco Live Challenge leaderboard.

Level up and earn **exclusive prizes!**

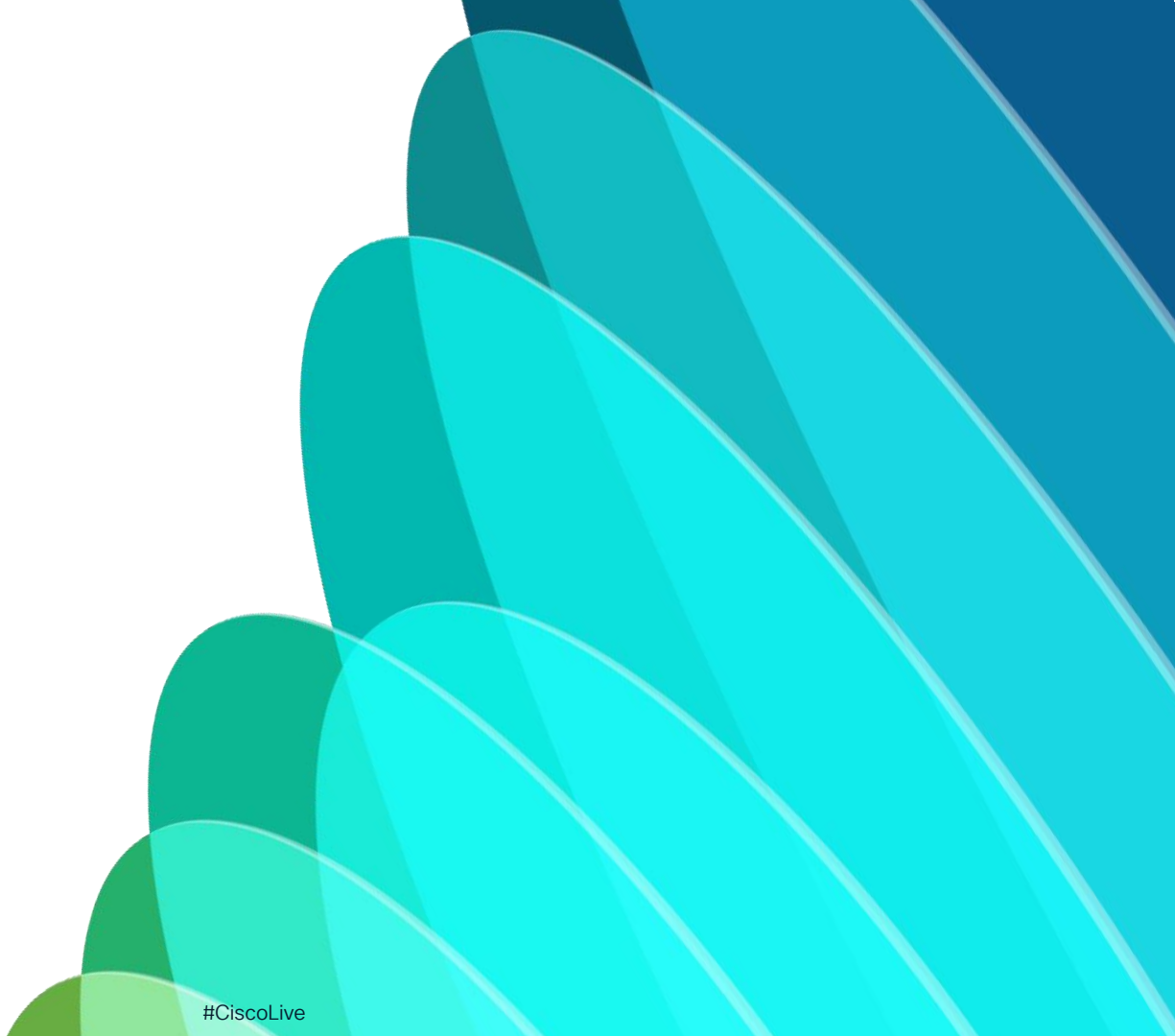
Complete your surveys in the **Cisco Live mobile app.**

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

The bridge to possible

Thank you



cisco Live!