
cisco
You make possible

Programmability with IOS-XR Platforms

Shambhu Mishra
Technical Consulting Engineer

BRKOPS-2285

cisco Live!
Barcelona | January 27-31, 2020

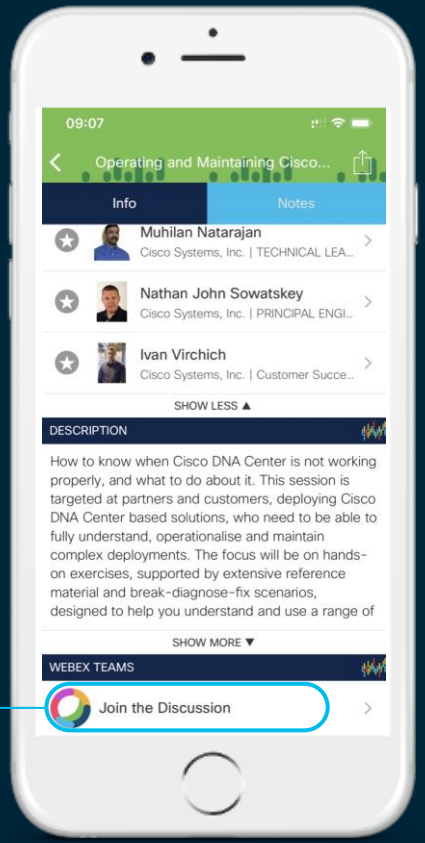
Cisco Webex Teams

Questions?

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

How

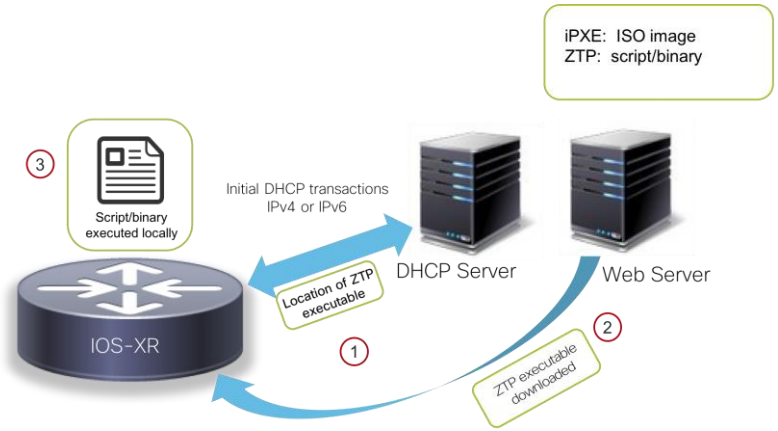
1. Find this session in the Cisco Events Mobile App
2. Click “Join the Discussion”
3. Install Webex Teams or go directly to the team space
4. Enter messages/questions in the team space

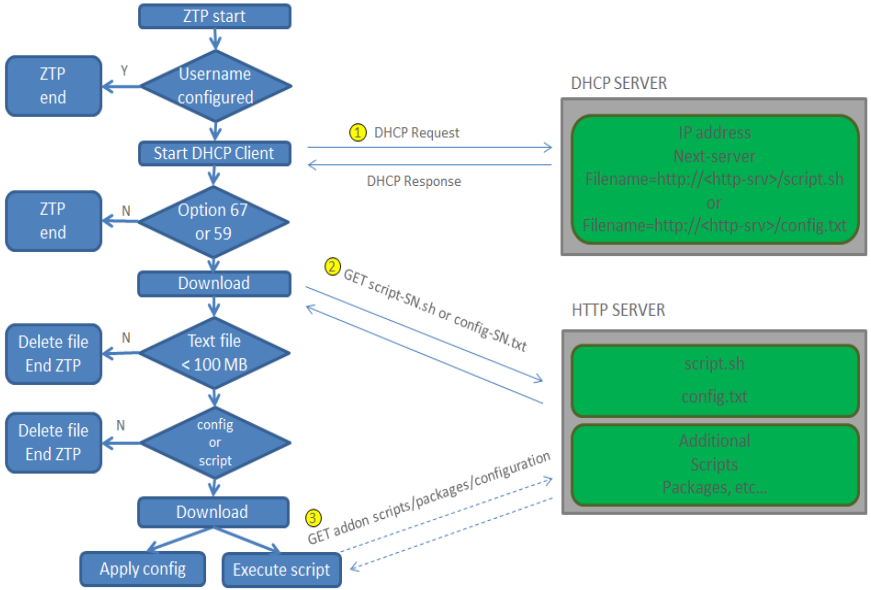


Agenda

- Cisco ZTP (Bringing up automation)
- Model Driven Manageability
- Streaming Telemetry
- Service layer API
- App Hosting
- Demo

ZTP





Ref : https://developer.cisco.com/learning/lab/01-iosxr-02-cli-automation-python/step/1

ZTP helper Libraries

xrapply(), xrapply_string(), xrreplace()

```
RP/0/RP0/CPU0:r1#bash Sat Sep 8 18:53:22.730 UTC
[r1:~]$ python

Python 2.7.3 (default, Dec 12 2017, 08:22:03)
[GCC 4.9.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
```

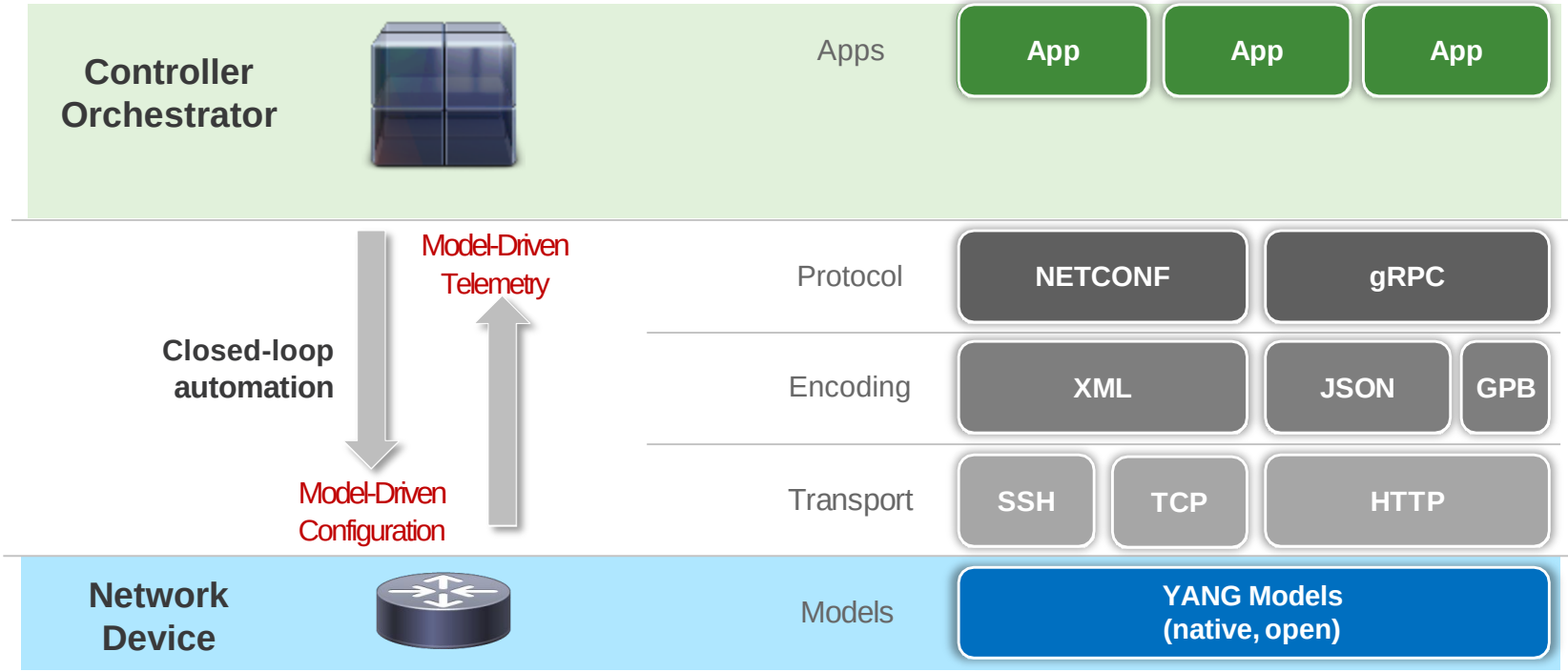
```
>>> import sys
>>> sys.path.append("/pkg/bin")
>>> from ztp_helper import ZtpHelpers
>>> from pprint import pprint

>>> ztp_obj=ZtpHelpers()
>>> cmd={"exec_cmd" : "show running-config"}
>>> ztp_obj.xrcmd(cmd)
>>> pprint(ztp_obj.xrcmd(cmd))
```

Output

```
Building configuration...
{'output': ['!! IOS XR Configuration version = 6.4.1',
            '!! Last configuration change at Sat Sep  8 19:07:36 2018 by ZTP',
            '!',
            'hostname r1',
            'banner motd ;',
            '-----------------------------------------------------------------------',
            'Router 1 (Cisco IOS XR Sandbox)',
            '-----------------------------------------------------------------------',
            ';',
            'logging console debugging',
            'service timestamps log datetime msec',
            'service timestamps debug datetime msec',
            'username admin',
            'group root-lr',
            'group cisco-support',
            'secret 5 $1$A4C9$oaNorr6BXDruE4gDd086L.',
            '!',
            'line console',
            'timestamp disable',
            'exec-timeout 0 0',
            '!',
            'vty-pool default 0 4 line-template VTY-TEMPLATE',
            'call-home',
            'service active',
            'contact smart-licensing',
            'profile CiscoTAC-1',
            'active',
            'destination transport-method http',
            '!',
            '!',
```

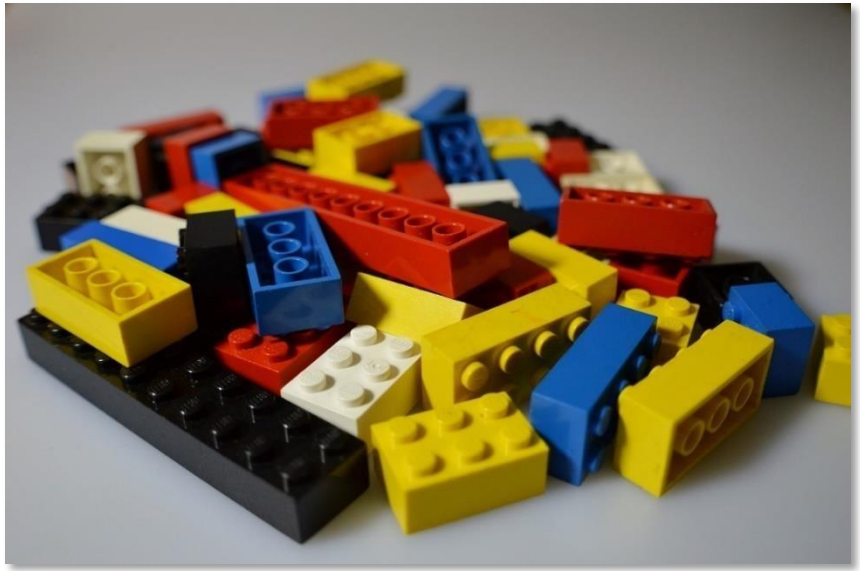
Model Driven Automation(NETCONF,gRPC,gNMI)–ANX

Model-Driven Manageability

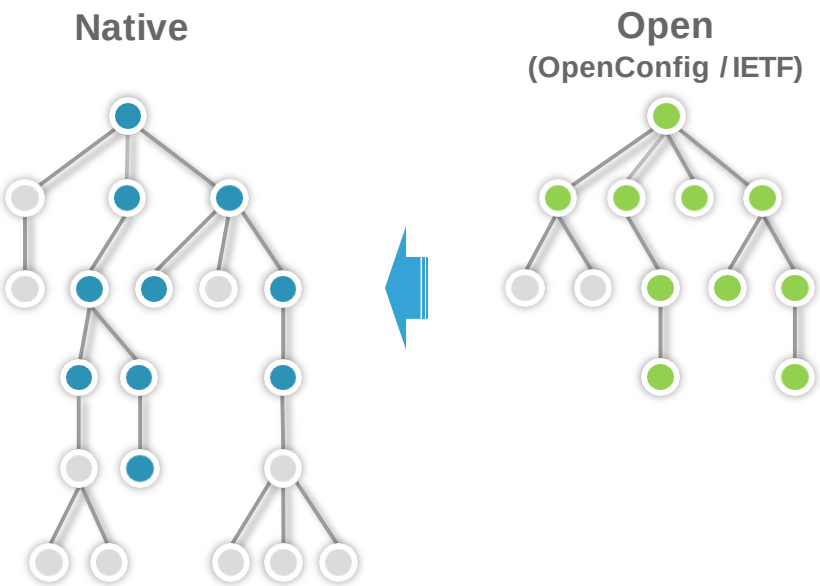


Benefits of Model-Driven Manageability

- Model based, structured, computer friendly
- Multiple model types (native, OpenConfig, IETF, etc.)
- Models decoupled from transport, protocol and encoding
- Choice of transport, protocol and encoding
- Model-driven SDKs for abstraction and simplification
- Wide standard support while leveraging open source



Data Models in Cisco IOS XR

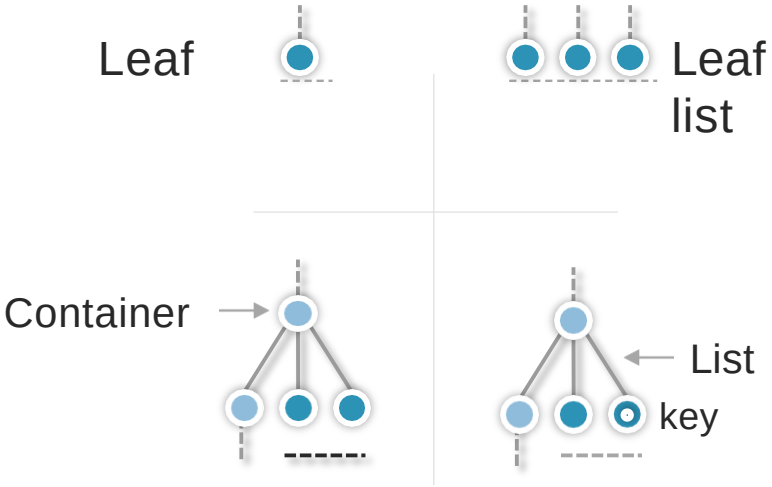


- Data (config and operational) and actions/commands (RPCs) in a tree structure
- Self-documented and shipped with devices
- Native (XR specific) and open (vendor neutral) models
- Native models provide most coverage
- Open (OpenConfig and IETF) provide reduced coverage
- Open models internally mapped to native models

Cisco IOS XR Native Data Models

- Provide most comprehensive coverage for device functionality
- Approximately ~385 models in XR 6.5.1 (800+ YANG files)
- A single model defines either configuration (cfg), operational state (oper) or an action/command (act)
  - Cisco-IOS-XR-ipv4-bgp-cfg
  - Cisco-IOS-XR-ipv4-bgp-oper
  - Cisco-IOS-XR-ipv4-bgp-act
- Models posted at
  - https://github.com/YangModels/yang/tree/master/vendor/cisco/xr

YANG



- Modeling language for networking
- Defines data hierarchy (config or oper), RPCs and notifications
- Main node types
  - Leaf – node with name, type and value (no children)
  - Leaf list – sequence of leafs (no children)
  - Container – node that groups nodes and has no type or value
  - List – Series of data instances generally with one or more keys
- Models extended through augmentations
- Unsupported nodes specified as deviations

YANG Model Example

YANG

```
container community-sets {
       description “Container for community sets";
       list community-set {
         key community-set-name;
         description "Definitions for community sets";
         leaf community-set-name {
           type string;
           description "name of the community set";
         }
         leaf-list community-member {
           type string {
             pattern '([0-9]+:[0-9]+)';
           }
           description "members of the community set";
         }
       }
     }
```

CLI

```
community-sets
  community-set C-SET1
    65172:1,
    65172:2,
    65172:3
  !
  community-set C-SET10
    65172:10,
    65172:20,
    65172:30
  !
!
```

Model Data Example

XML

```
<community-sets>
  <community-set>
    <community-set-name>C-SET1</community-set-name>
    <community-member>65172:1</community-member>
    <community-member>65172:2</community-member>
    <community-member>65172:3</community-member>
  </community-set>
  <community-set>
    <community-set-name>C-SET10</community-set-name>
    <community-member>65172:10</community-member>
    <community-member>65172:20</community-member>
    <community-member>65172:30</community-member>
  </community-set>
</community-sets>
```

CLI

```
community-sets
  community-set C-SET1
    65172:1,
    65172:2,
    65172:3
  !
  community-set C-SET10
    65172:10,
    65172:20,
    65172:30
  !
!
```

Model Data Example

JSON

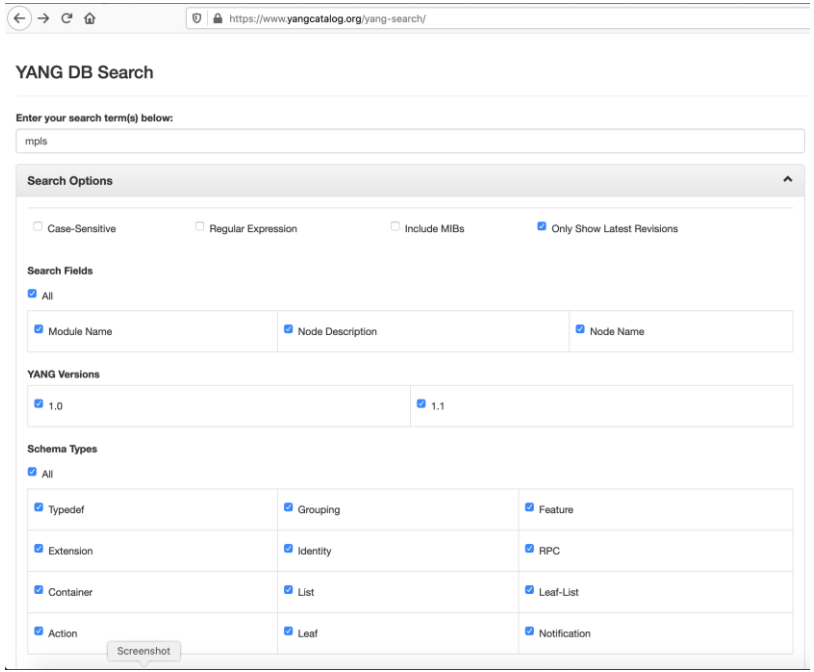
```
{   "community-sets": {
        "community-set": [
            {   "community-set-name": "CSET1",
                "community-member": [
                    "65172:1",
                    "65172:2",
                    "65172:3" ]
            },
            {   "community-set-name": "CSET10",
                "community-member": [
                    "65172:10",
                    "65172:20",
                    "65172:30" ]
            }
        ]
    }
}
```

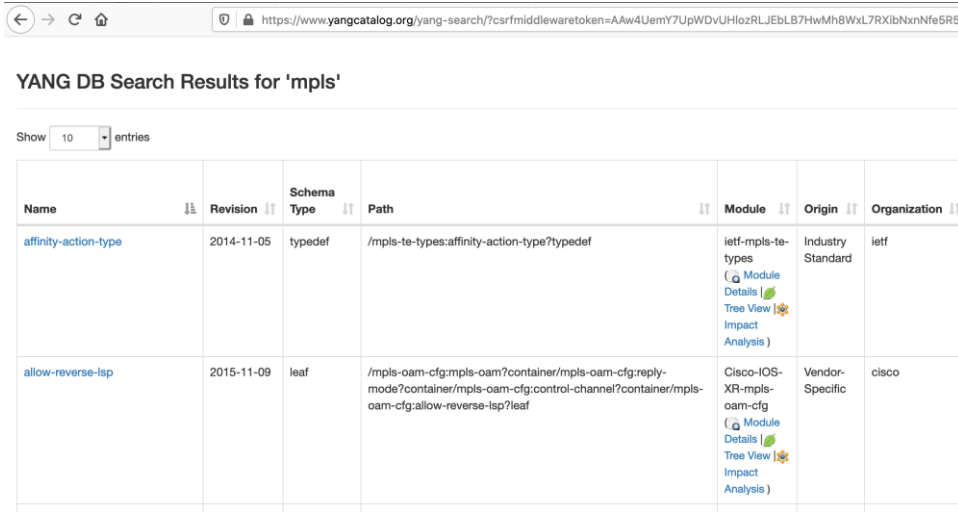
CLI

```
community-sets
  community-set C-SET1
    65172:1,
    65172:2,
    65172:3
  !
  community-set C-SET10
    65172:10,
    65172:20,
    65172:30
  !
!
```

Look for your Yang-Model…

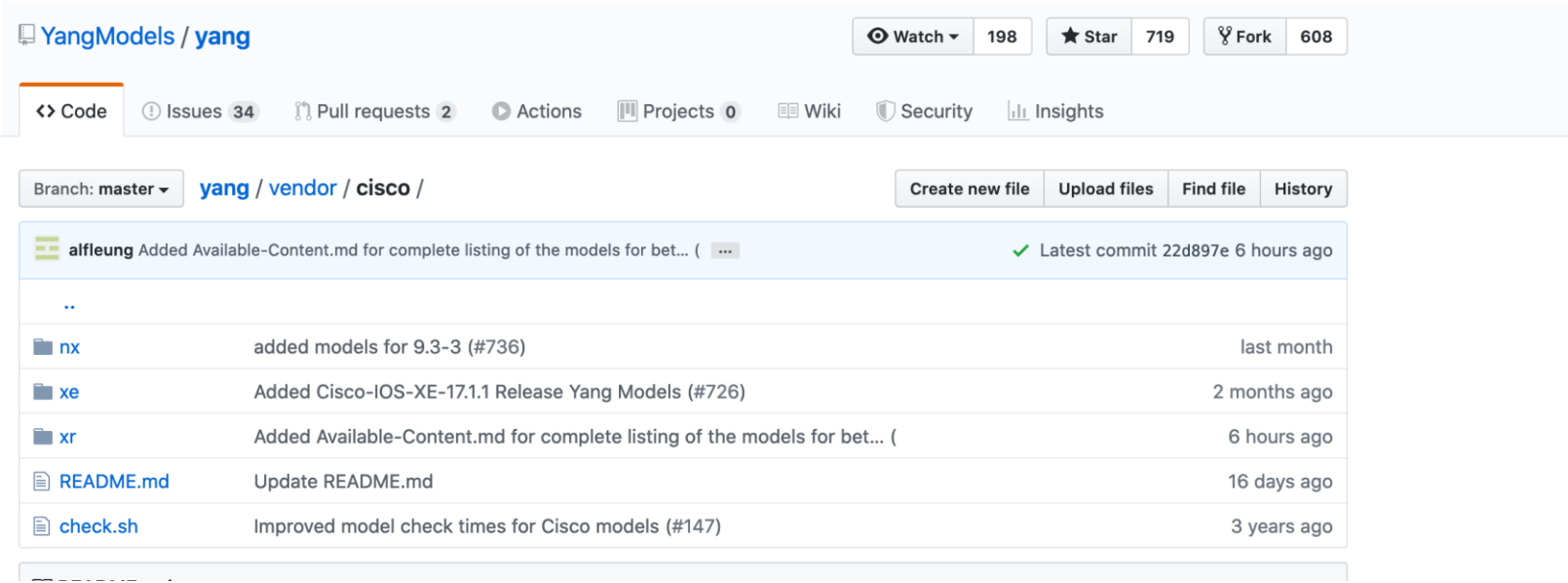
https://www.yangcatalog.org/yang-search/





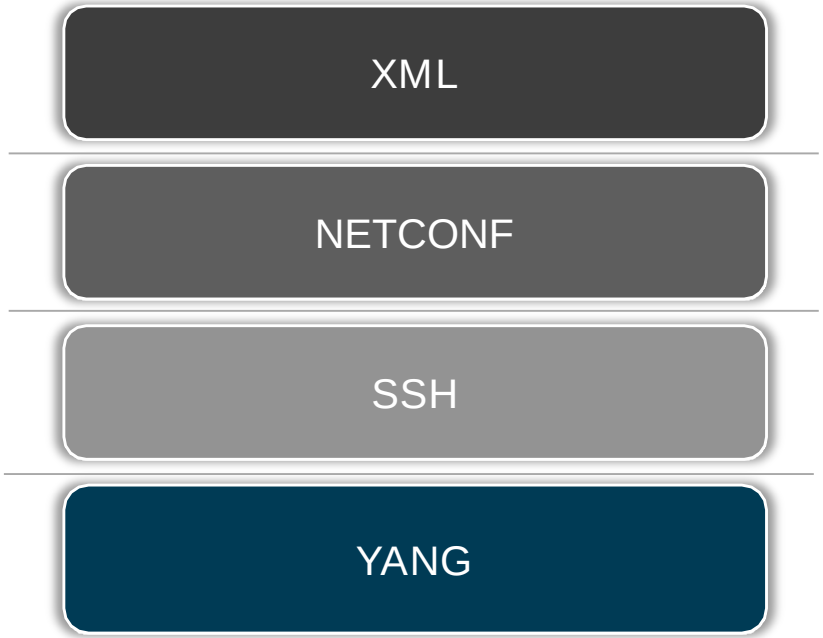
Where can I download Yang-models

https://github.com/YangModels/yang/tree/master/vendor/cisco



NETCONF Protocol Overview

- Rich functionality to manage configuration and operational (state) data
- Operations defined as RPCs (request / reply) in XML
- Client/app initiate request towards server/device
- Supports running, candidate and startup configurations
- Capability exchange during session initiation



Main NETCONF Protocol Operations

Operation	Description
get-config	Retrieve all or part of a specified configuration
edit-config	Loads all or part of a specified configuration (merge, replace, create, delete, remove)
copy-config	Create or replace an entire configuration datastore
get	Retrieve all or part of running configuration and device operational data
get-schema	Retrieve device schema (model)
lock	Lock entire configuration datastore (e.g. candidate)
unlock	Remove lock on entire configuration datastore (e.g. candidate)
close-session	Request graceful session termination

NETCONF Edit-Config Operations

Operation	Description
Merge	Merge configuration with existing configuration (default)
Replace	Replace configuration with existing configuration
Create	Create configuration if non-existent. Otherwise, return error. (non-idempotent*)
Delete	Delete configuration if non-existent. Otherwise, return error. (non-idempotent)
Remove	Remove configuration. Ignore if configuration non-existent.

* Cannot be applied multiple times without changing the result beyond the initial application

Overview of gRPC on Cisco IOS XR

- Google RPC provides a general (open source) RPC framework
- Interface definition in Cisco IOS XR specifies device operations
- Functional subset of NETCONF
- Simple client development
- High performance

Protocol Operations in Cisco IOS XR Interface

Operation	Description
GetConfig	Retrieve configuration
MergeConfig	Merge configuration
DeleteConfig	Delete configuration
ReplaceConfig	Replace configuration
CommitReplace	Replace entire configuration
GetOper	Retrieve operational data
CliConfig	Merge configuration data in CLI format
ShowCmdTextOutput	Retrieves CLI show-command output data

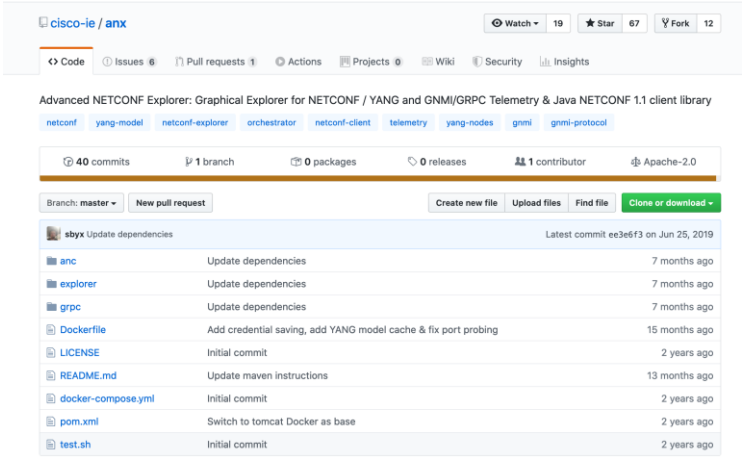
Protocol Operations in OpenConfig gNMI Interface

Operation	Description
capabilities	Discover device capabilities (models, encodings, version, extensions)
get	Retrieve device state
set	Modify device state (delete, replace, update)
subscribe	Subscribe to device update

gNMI Implementation in Cisco IOS XR

- Based on gNMI v0.4.0
- Introduced in release 6.5.1
- Set and Get RPCs use JSON_IETF (RFC 7951) and ASCII (CLI) encoding
- Subscribe RPC
  - Paths must consider data aggregation points (no arbitrary paths)
  - No aliases

ANX(Advanced NETCONF Explorer)



https://github.com/cisco-ie/anx

✓ Clone this Git

git clone https://github.com/cisco-ie/anx.git

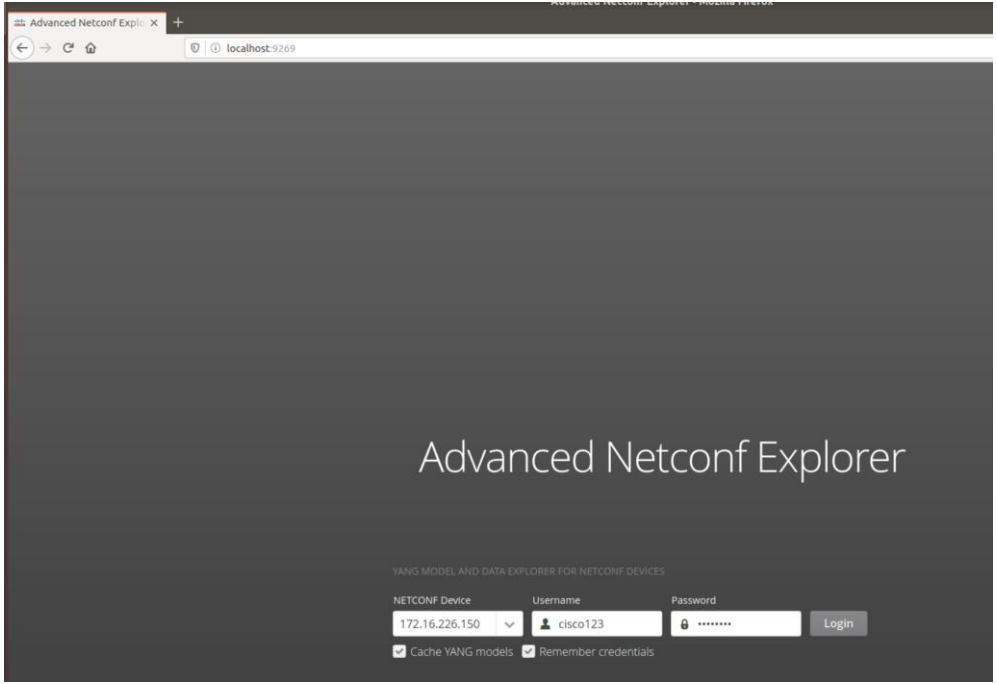
✓ Change to the anx directory

#docker-compose up – d

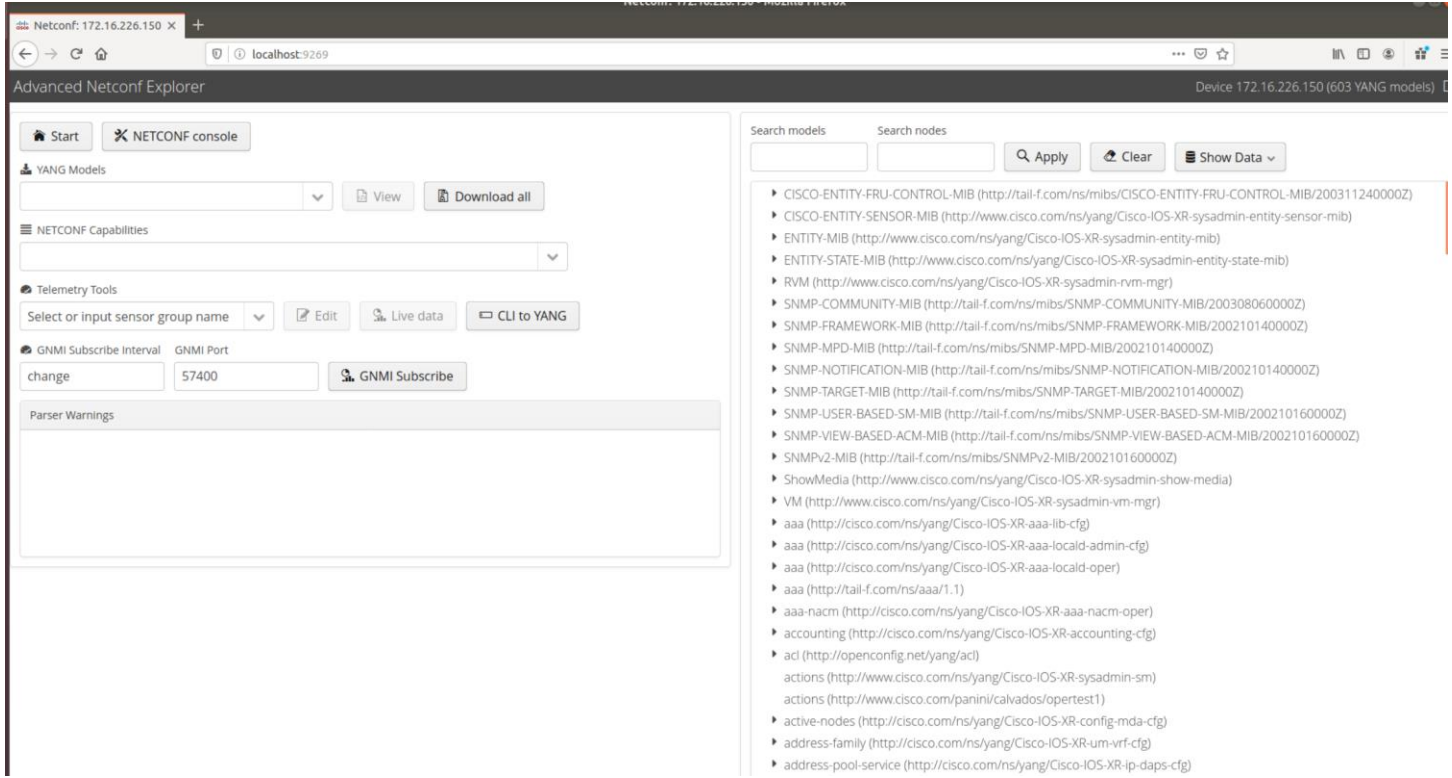
```
shambhu@shambhu-virtual-machine:/$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
anx_anx             latest              8e95e943ed19        2 days ago          165MB
```

ANX(Advanced NETCONF Explorer)

```
shambhu@shambhu-virtual-machine:/$ docker images | grep anx
anx_anx             latest              8e95e943ed19        2 days ago          165MB
shambhu@shambhu-virtual-machine:/$ docker start anx_anx_1
anx_anx_1
shambhu@shambhu-virtual-machine:/$ docker ps | grep anx
836484db6531        anx_anx             "catalina.sh run"   2 days ago          Up 4 minutes        0.0.0.0:9269->8080/tcp   anx_anx_1
shambhu@shambhu-virtual-machine:/$
```



ANX(Advanced NETCONF Explorer)

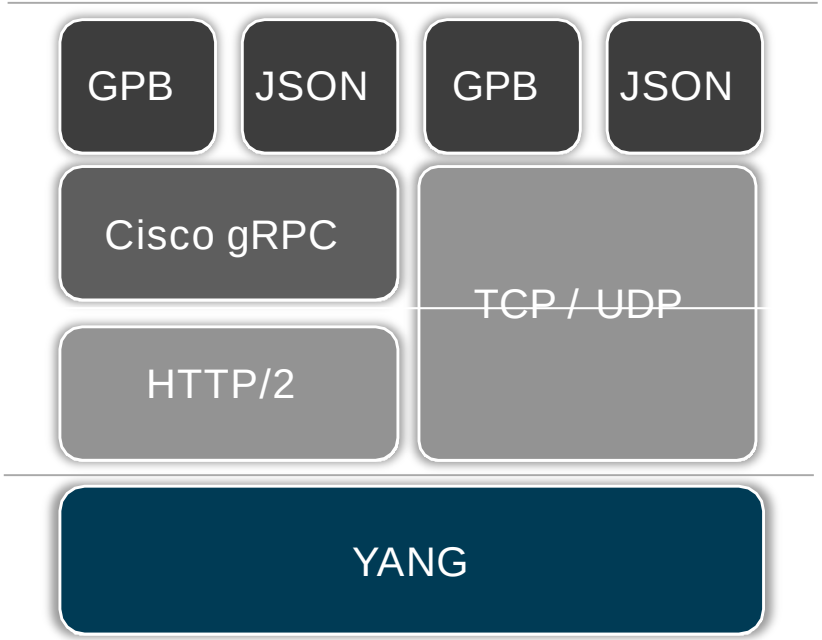


Streaming Telemetry– ANX, Pipeline (architecture and operation)

Telemetry

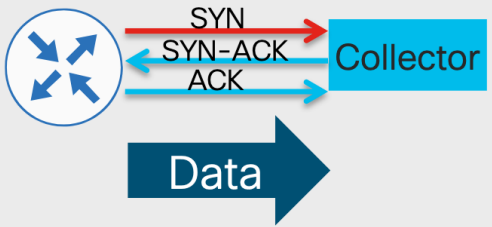
Overview of Telemetry on Cisco IOS XR

- Loosely-coupled stack
  - Data encoding (JSON vs GPB)
  - Transport (HTTPv2 vs TCP vs UDP)
  - Data model (native vs open)
- Session initiation
  - Dial-in (transient destination)
  - Dial-out (persistent destination)
- Flexible data streaming modes (frequency vs event driven)



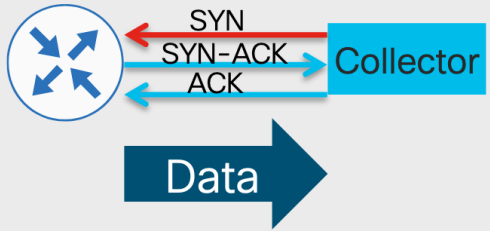
MDT Modes...

Dial-Out



- TCP & gRPC (from 6.1.1)
- UDP (from 6.2.1)

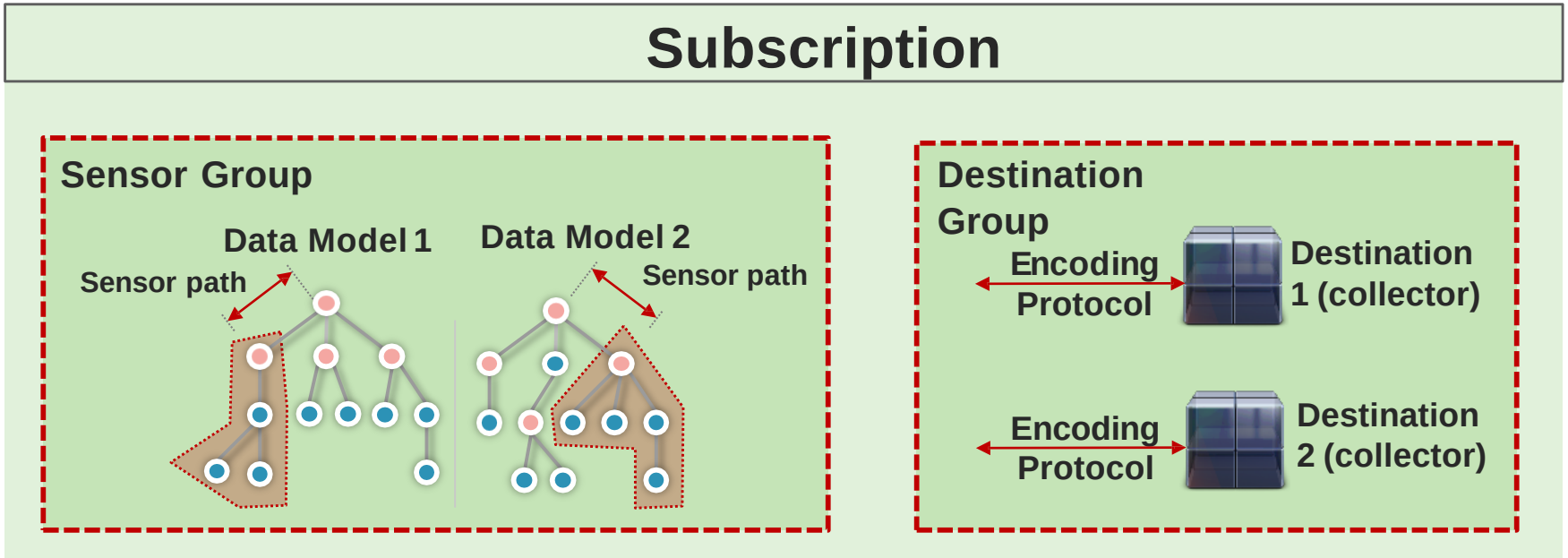
Dial-In



- gRPC only (from 6.1.1)

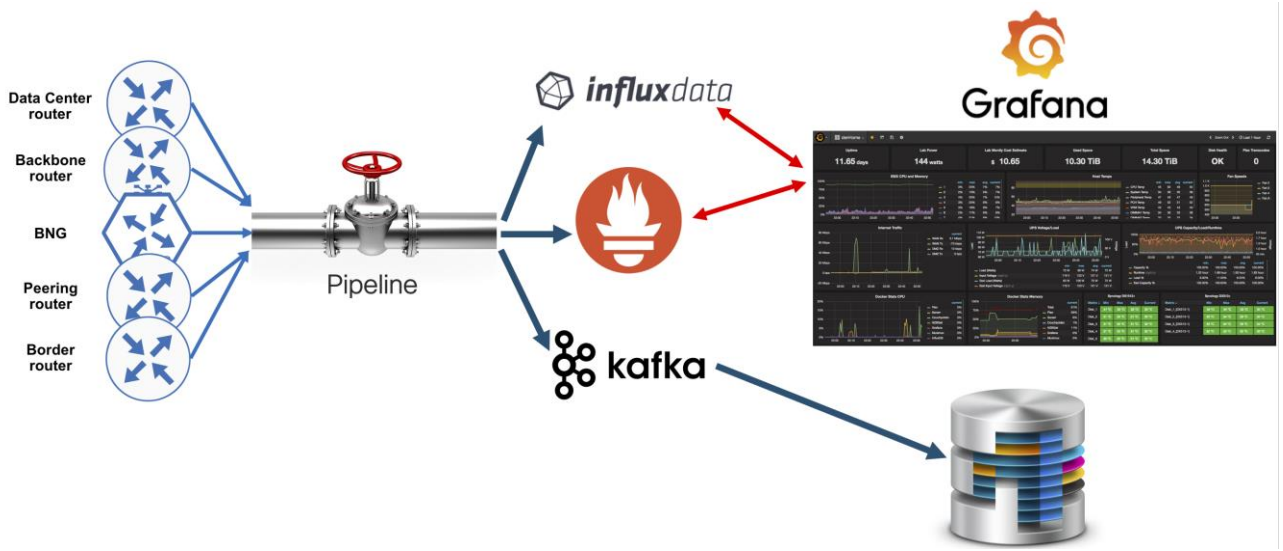
Telemetry Subscriptions

```
sensor-group memory
  sensor-path Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters
!
```



Aggregation point

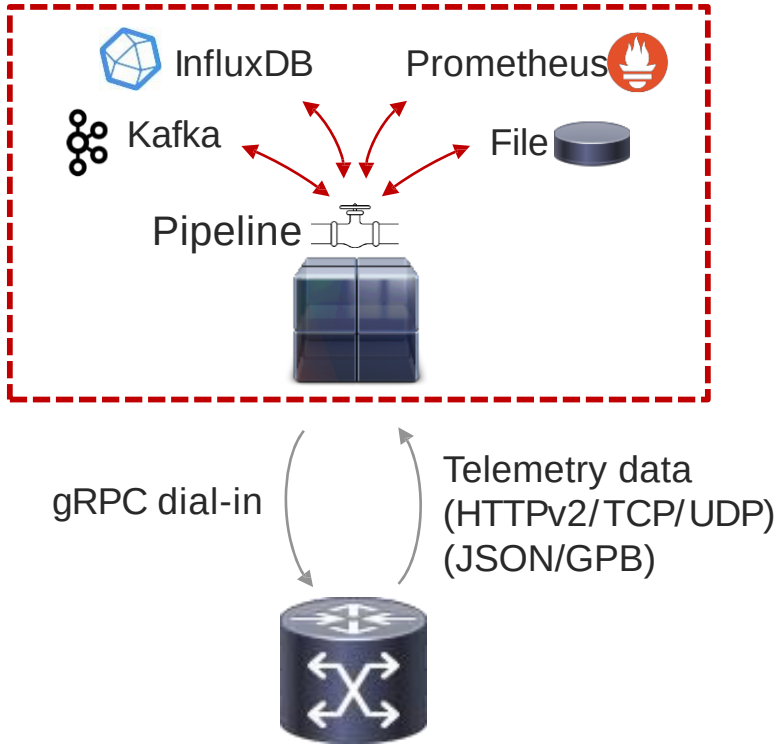
Telemetry tool (Pipeline)



https://github.com/cisco/bigmuddy-network-telemetry-pipeline

Pipeline – An Open-Source Telemetry Collector

- Collector for telemetry data
- Performs basic encoding transformation
- Data producer for Kafka, InfluxDB, Prometheus, etc.
- Supports dial-in and dial-out sessions



XR config

```
telemetry model-driven
 destination-group DG1
  address-family ipv4 172.16.226.132 port 5432
   encoding self-describing-gpb
   protocol tcp
  !
 !
 sensor-group memory
  sensor-path Cisco-IOS-XR-wdsysmon-fd-oper:system-monitoring/cpu-utilization
  sensor-path Cisco-IOS-XR-nto-misc-oper:memory-summary/nodes/node/summary
  sensor-path Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/data-rate
  sensor-path Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters
!
 subscription SUB
  sensor-group-id memory sample-interval 5000
  destination-id DG1
```

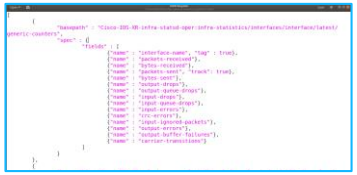
Pipeline Config

Input from device

pipeline.conf

```
[default]

id = pipeline

[fromRouters]
stage = xport_input
type = tcp
encap = st
listen = :5432
logdata = off
keepalive_seconds = 3


[toInflux]
stage = xport_output
type = metrics
file = metrics.json
output = influx
influx = http://localhost:8086
database = mdtdb
```

Metric.json

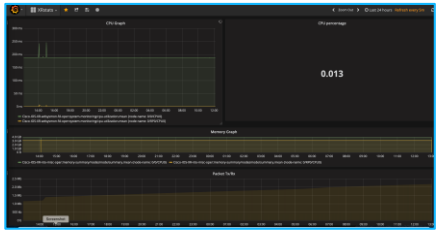


Output to Influxdb

InfluxDB

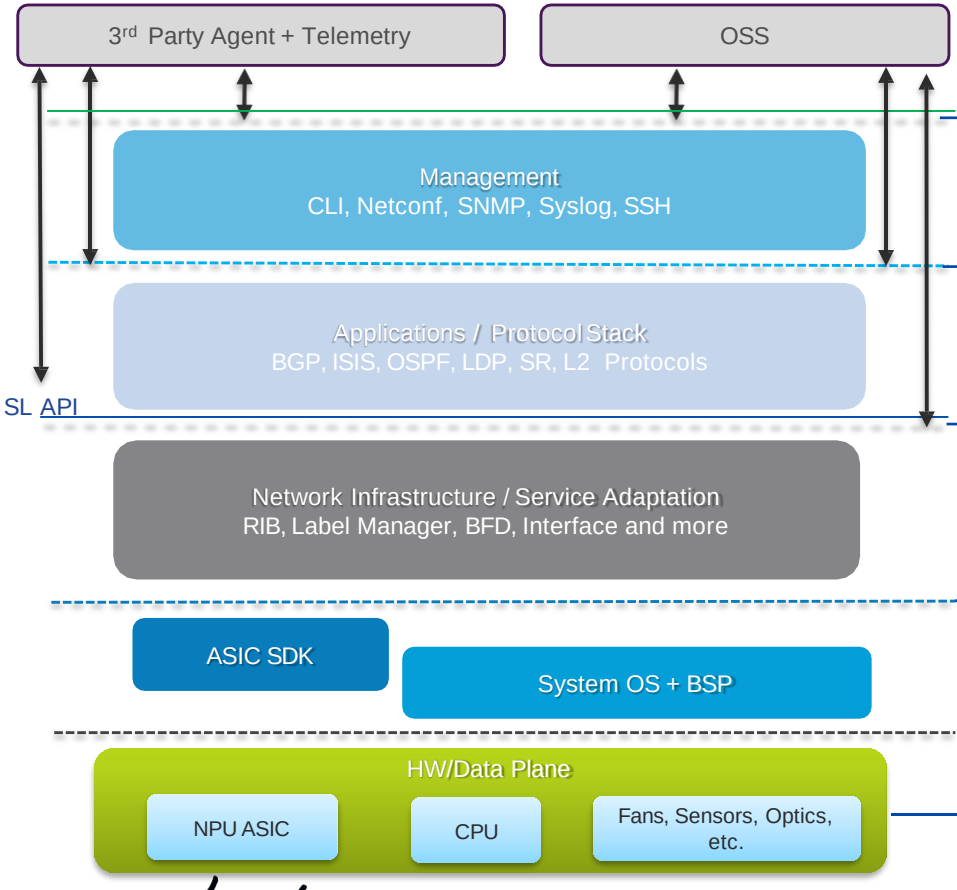
InfluxDB

Query



Service Layer API

De-Layering The Network Stack



Management/Presentation Layer

Provides access to configure and manage the stack through Network config/oper DB: Yang Models, CLI.

Application/Protocol Layer

Provides APIs into the Routing Protocols (BGP, IGP, SR, etc.)

Network Infrastructure Layer / Service Adaptation Layer

- Acts as the bridge between the Application Layer and the HW
- Presents abstractions to the Application/Protocol Layer

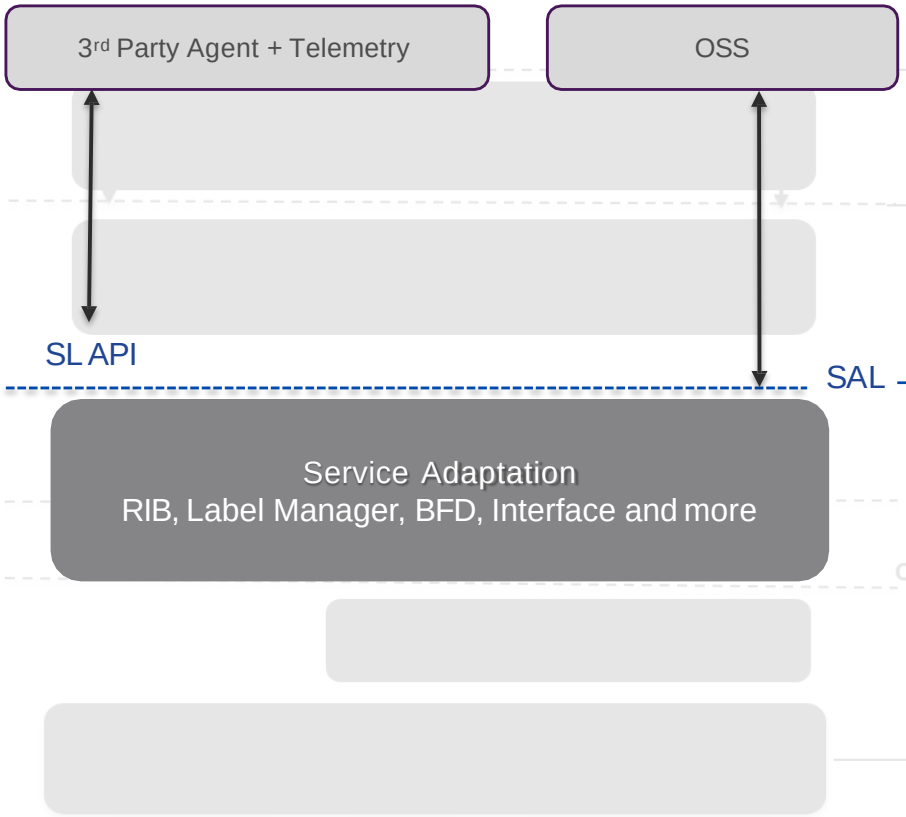
System OS - Linux Kernel
BSP(Board Support Package) - Boot Loader, Device Drivers, etc.

ASIC SDK and drivers for the SDK

Hardware
Consists of ASIC/Chipset from HW vendors + CPU, Fans, Sensors

cisco Live!

Zooming in .....



Network Infrastructure Layer / Service Adaptation Layer

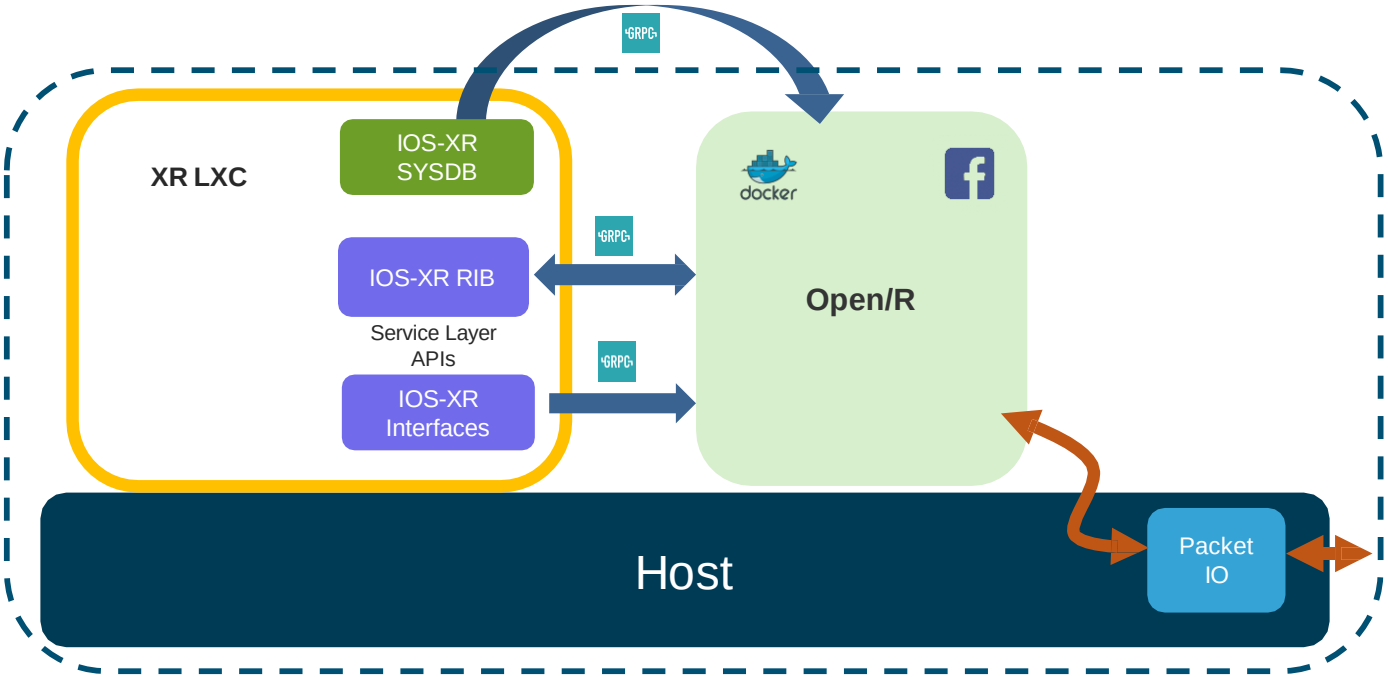
- Acts as the bridge between the Application Layer and the HW
- Presents abstractions to the Application/Protocol Layer
- Highly Performant API outside the Central Config/Oper Database context

Bring your own Protocol/Agent

On-box agents and custom protocols that co-exist with standard protocols to influence routing.

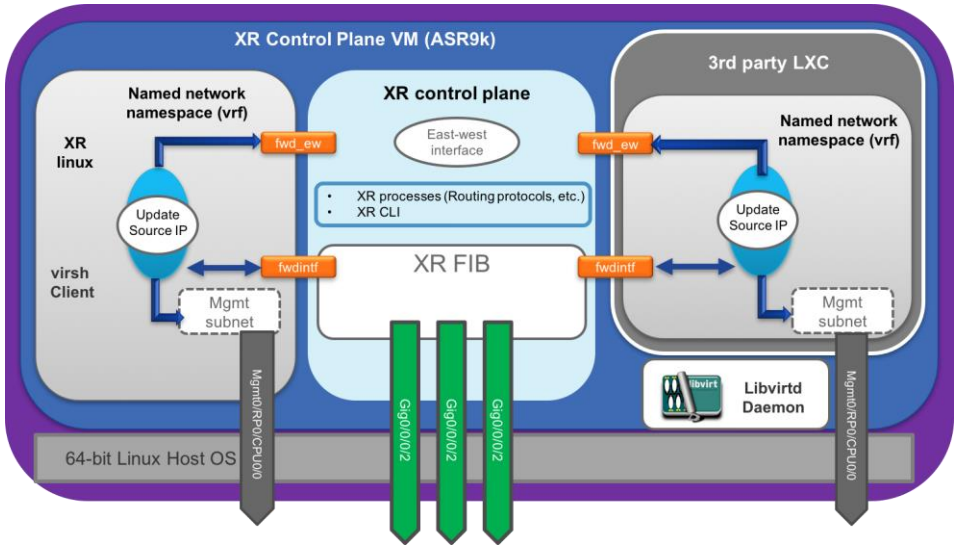
Running Open/R on XR

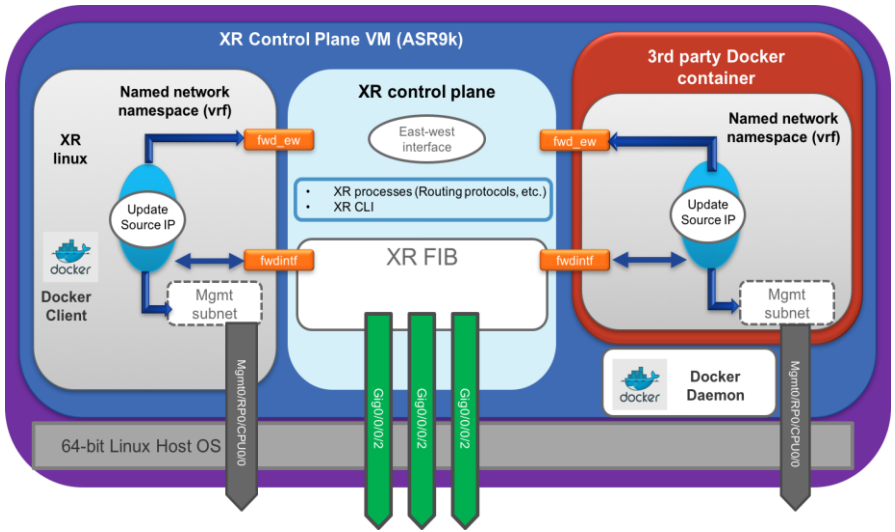
https://github.com/akshshar/openr-xr



APP Hosting

App Hosting Architecture





App Hosting

Open/R

```
RP/0/RP0/CPU0:rtr1#bash
 Fri Feb 16 22:46:52.944 UTC
[rtr1:~]$ docker images
REPOSITORY                TAG       IMAGE ID        CREATED              SIZE
11.11.11.2:5000/openr     latest    fdddb43d9600    33 seconds ago       1.829 GB
[rtr1:~]$
```

```
[host:~]$ docker version
Client:
 Version:      1.10.0
 API version:  1.22
 Go version:   go1.4.2
 Git commit:   abaf4ef
 Built:        Mon Apr 25 14:00:32 2016
 OS/Arch:      linux/amd64

Server:
 Version:      1.10.0
 API version:  1.22
 Go version:   go1.4.2
 Git commit:   cb6da92
 Built:        Tue Aug  8 22:08:35 2017
 OS/Arch:      linux/amd64
```

Demo

Further Readings...

- Advanced Topics in Cisco IOS Telemetry : BRKSPG-2503
- Advanced IOS-XR Programmability Using Service Layer APIs: BRKSPG-3000
- ZTP : https://xrdocs.io/device-lifecycle/tutorials/2016-08-26-working-with-ztp/

Stay Subscribed
https://xrdocs.io/

Complete your online session survey

- Please complete your session survey after each session. Your feedback is very important.
- Complete a minimum of 4 session surveys and the Overall Conference survey (starting on Thursday) to receive your Cisco Live t-shirt.
- All surveys can be taken in the Cisco Events Mobile App or by logging in to the Content Catalog on ciscolive.com/emea.

Cisco Live sessions will be available for viewing on demand after the event at ciscolive.com.

cisco Live!

Continue your education

Demos in the Cisco Showcase

Walk-In Labs

Meet the Engineer 1:1 meetings

Related sessions

Thank you

cisco Live!

You make **possible**