Model-Driven Telemetry and Analytics

Steven Barth & Cristina Precup
Software Systems Engineers
Questions?
Use Cisco Spark to communicate with the speaker after the session

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

cs.co/ciscolivebot#BRKNMS-3537
Scope of this session

What we will cover:
• How Telemetry is consumed
• How you can implement a closed-loop usecase with OS software
• How Telemetry and Machine Learning can play together

What we will not cover:
• Telemetry and IOS XR basics

Telemetry Track @ CL Barcelona
• DEVNET-1710 on Wednesday General intro, end-to-end overview
• BRKSPG-2999 on Wednesday Advanced Telemetry: Devices, scale and security
• BRKNMS-3537 you are here!
• LTR-2578 on Fri. 09.00-13.00 Consuming Telemetry: hands-on guided lab
Agenda

• Introduction
• Demo: Link-quality assurance with Model-Driven Telemetry
• Streaming Telemetry data from a device to a collector
• Storing and processing Telemetry data in a time series platform
• Visualizing data and reacting to events
• Demo: Telemetry-based vBNG address pool management
• Machine Learning & Big Data use cases
• Conclusion
Traditional Monitoring Concepts
No Longer suited for Cloud-Scale Network Operations

Where Data Is Created
Sensing & Measurement

Scale Issues
SNMP
syslog
CLI

Where Data Is Useful
Storage & Analysis

Strong burden on back-end
Normalize different encodings, transports, data models, timestamps

Where Data Is Created

Subject to Change
Unstructured
Streaming Telemetry Concepts
Better suited for Cloud-Scale Network Operations

Where Data Is Created

Streaming Telemetry

Push paradigm
One consistent way to access Statistics, Oper state & Events @ all layers
High Performance: 10 sec
Multiple encodings & Transport

Where Data Is Useful

Volume: Scale of Data
Velocity: Analysis of Streaming Data
Variety: Different Forms of Data

Sensing & Measurement

Storage & Analysis
Streaming Telemetry
Design Vision

Performance
• Get as much data off the box as quickly as possible

Coverage
• Grant full access to all operational data on the box*

Automation
• Serialize the data in a flexible, efficient way that fits customers automated tools

*User needs to have the correct privileges
Demo: Link-quality assurance with Model-Driven Telemetry
Closed-loop Telemetry setup with open-source analytics tools

- IOS XRv 9000
- Pipeline
- InfluxDB & Kapacitor
- Cisco NSO
- InfluxDB
- Grafana
- Ansible
Demo: Link-quality assurance with Model-Driven Telemetry

Demo use case

Introduce impairment (latency 50ms)

Monitor quality metrics (CRC errors, dropped packets, latency, …)
Demo: Link-quality assurance with Model-Driven Telemetry

Which operational YANG model?

- Cisco-IOS-XR-infra-statsd-oper
- infra-statistics
  - interfaces
    - interface
      - cache
    - generic-counters
      - applique
        - availability-flag
        - broadcast-packets-received
        - broadcast-packets-sent
        - bytes-received
        - bytes-sent
  - carrier-transitions
  - crc-errors
    - framing-errors-received
    - giant-packets-received
    - input-aborts
    - input-drops
  - input-errors
    - input-ignored-packets
    - input-overruns
    - input-queue-drops
    - last-data-time
    - last-discontinuity-time
    - multicast-packets-received
    - multicast-packets-sent
    - output-buffer-failures
  - output-errors
    - output-ignored-packets
    - output-overruns
    - output-queue-drops
    - output-packets-transmitted

Which sensors to measure?

- Cisco-IOS-XR-ip-bfd-oper
  - bfd
    - ipv4-multi-hop-session-details
    - ipv4-single-hop-session-details
    - ipv4-bfd-do-mplste-head-session-details
    - ipv4-bfd-do-mplste-tail-session-details
    - ipv6-multi-hop-session-details
    - ipv6-single-hop-session-details
    - label-session-details
    - session-details
    - session-detail
      - association-information
        - destination-address
  - interface-name
    - location
    - lsp-ping-info
    - mp-download-state
    - owner-information
    - status-information
      - async-receive-statistics
      - async-transmit-statistics
        - desired-minimum-echo-transmit-interval
      - echo-received-statistics
      - echo-transmit-statistics
        - internal-label
        - last-state-change
      - latency-average

What are the keys / dimensions?
Streaming Telemetry data from a device to a collector
Router Configuration
IOS XR model-driven telemetry & GRPC

What to stream?

When to stream?

Where to listen for collector?

telemetry model-driven
  sensor-group sensor1
    sensor-path Cisco-IOS-XR-ip-bfd-oper:bfd/session-details/session-detail
    !
    subscription mdt-realtime
    sensor-group-id sensor1 sample-interval 5000
    !
  !
  grpc
    port 57777
    !
Pipeline
An open-source Telemetry collector

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

1. Input Stage

What protocol to use?
What encoding to use?
Where to connect to?

[dialin_XRv-AS65510-3]
stage = xport_input
type = grpc
encap = gpb
encoding = gpbkv
server = 172.16.1.65:57777
Pipeline Configuration
2. Filter Stage

Which YANG model?

```
[{
  "basepath": "Cisco-IOS-XR-ip-bfd-oper:
    bfd/session-details/session-detail",
  "spec": {
    "fields": [
      {
        "name": "interface-name", "tag": true},
      {
        "name": "status-information", "fields": [
          {
            "name": "latency-average"
          }
        ]
      }
    ]
  }
}
```

Which values are key(s)?

What is measured?
Pipeline Configuration

3. Output Stage

What output type to use?

What filter to use? (previous slide)

Where to stream to?

```
[metrics_influx]
stage = xport_output
type = metrics
file = /etc/mdt-realtime/metrics.json
datachanneldepth = 1000
output = influx
influx = http://influxdb:8086
database = mdt_realtime
workers = 10
username = admin
```
Storing and processing Telemetry data in a time series platform
Processing Telemetry data in a time series platform

What is a time series database?

- Series of data points (e.g. link latency) with given keys (e.g. device + interface)
- Run statistical functions (e.g. maximum, moving average, …)
- Trigger alarms and events
## Processing Telemetry data in a time series platform

**InfluxDB**

 ![InfluxDB Interface]

---

**Query:**

```
select * from "Cisco-IOS-XRinfra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters" WHERE time > now() - 1m and Producer = 'XRv-AS65510-5';
```

---

### Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters

<table>
<thead>
<tr>
<th>time</th>
<th>EncodingPath</th>
<th>Producer</th>
<th>interface-name</th>
<th>bytes-received</th>
<th>bytes-sent</th>
<th>carrier-transitions</th>
<th>input-drops</th>
<th>crc-errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>2017-11-21T20:32:58.469Z</td>
<td>&quot;Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters&quot;</td>
<td>&quot;XRv-AS65510-5&quot;</td>
<td>&quot;GigabitEthernet0/0/0/1&quot;</td>
<td>607540757</td>
<td>605662790</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2017-11-21T20:32:58.469Z</td>
<td>&quot;Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters&quot;</td>
<td>&quot;XRv-AS65510-5&quot;</td>
<td>&quot;MgmtEth0/RP0/CPU0/0&quot;</td>
<td>328461869</td>
<td>10609581226</td>
<td>1</td>
<td>3239</td>
<td>0</td>
</tr>
<tr>
<td>2017-11-21T20:32:58.469Z</td>
<td>&quot;Cisco-IOS-XR-infra-statsd-oper:infra-statistics/interfaces/interface/latest/generic-counters&quot;</td>
<td>&quot;XRv-AS65510-5&quot;</td>
<td>&quot;GigabitEthernet0/0/0/3&quot;</td>
<td>17463641</td>
<td>26257053</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Processing Telemetry data in a time series platform

```javascript
var latency = stream
  .from()
  .database('mdt_realtime')
  .retentionPolicy('autogen')
  .measurement('Cisco-IOS-XR-ip-bfd-oper:bfd/session-details/session-detail')
  .where(lambda: "Producer"=='XRv-AS65510-5')
  .where(lambda: "interface-name"=='GigabitEthernet0/0/0/0')
  .where(lambda: "status-information__latency-average" > 0)

latency
  .where(lambda: "status-information__latency-average" <= 100000)
  .crit(lambda: "status-information__latency-average" <= 100000)
  .exec('/telemetry-action.sh', 'enable')

latency
  .where(lambda: "status-information__latency-average" > 100000)
  .crit(lambda: "status-information__latency-average" > 100000)
  .exec('/telemetry-action.sh', 'disable')
```
Visualizing and reacting to events
Visualizing Data
Using Grafana

- Monitoring system
- Basic statistics on the data
- Event-based alerting system

Threshold of 10 ms
Reacting to Events
Using NSO

Applications
REST, RESTConf, NETCONF, Java, Python, Erlang, CLI, Web UI

Engineers

Service Manager
Device Manager
Network Equipment Drivers (NEDs)

NETCONF, REST, SNMP, CLI, etc

Physical Networks
Virtual Networks
Network Apps

• VNFM
• Controller Apps
• EMS and NMS

Service Model
Device Model
Reacting to Events
Using Ansible

• Agentless
• Widely adopted
• Infrastructure as code
• Simple to use and learn: YAML playbooks
• Community and vendor driven
• Modular open-source framework, easily modified
• Leverage many common programming languages
Reconfiguring IOS XR devices using Ansible Playbooks

Preamble: local execution
- hosts: all
gather_facts: no
connection: local

vars:
cli:
  host: "{{ ansible_ssh_host }}"
  username: "{{ ansible_ssh_user | default(cisco) }}"
  password: "{{ ansible_ssh_pass | default(cisco) }}"
  port: "{{ ansible_ssh_port | default(22) }}"

tasks:
- name: Avoid interface
  iosxr_config:
    provider: "{{ cli }}"
    parents:
    - "router isis {{ isis_name }}"
    - "interface {{ interface_name }}"
    lines: ["address-family ipv4 unicast metric maximum"]

Map inventory to playbook

Use built-in IOS XR module

Configuration to send
Assembling the open-source Telemetry platform
Docker and the evolution of virtualization

Monolithic Bare Metal
- App
- App
- App
- App
- Lib
- OS
- Hardware

Monolithic Virtual Machine
- App
- App
- Lib
- Guest OS
- Hypervisor
- Host OS
- Hardware

Application Containers with Docker
- App
- App
- App
- App
- Lib
- Host OS
- Hardware

Application Containers with Docker with Docker
- App
- App
- App
- App
- Lib
- Host OS
- Hardware
Assembling the open-source Telemetry platform

Overview on Docker

Registry
A hub of Docker images

Image
A complete app environment; base OS, application, libraries, dependency

Container
An instance of a Docker image

Build
Automated build with Dockerfile

Pull
Commit
Run

Push
Assembling the open-source Telemetry platform
Composing and orchestration with Docker Compose
Assembling the open-source Telemetry analytics platform
Composing and orchestration with Docker Compose

**grafana:**
- image: appcelerator/grafana:grafana-4.6.2
- volumes: 
  - "./grafana:/etc/extra-config/grafana"
- ports: 
  - "3000:3000"

**kapacitor:**
- build: kapacitor
- depends_on: [influxdb, pipeline, grafana]
- environment:
  - KAPACITOR_INFLUXDB=http://influxdb:8086
  - INVENTORY=xrv9000 ansible_ssh_host=...
  - INTERFACE=${XR_INTERFACE}
  - ISIS=${XR_ISIS}

**influxdb:**
- image: influxdb:1.4.2-alpine

**pipeline:**
- build: pipeline
- depends_on: [influxdb]
- environment:
  - MDT_DIALIN_PORT=57777
  - MDT_DIALIN_SUBSCRIPTIONS=...
  - INFLUX_URL=http://influxdb:8086
  - INFLUX_USERNAME=admin
  - INFLUX_PASSWORD=admin
  - INFLUX_DB=mdt_realtime
  - INFLUX_DB_CREATE=1
  - MDT_DIALIN_HOST_0=xrv9000 ...

Full demo source:
https://github.com/sbyx/telemetry-breakout
Demo: Telemetry-based vBNG address pool management
Demo: Telemetry-based vBNG address pool management

renew dhcp GigabitEthernet 2.800

DHCP bindings monitoring

BNG server
- GigE 0/0/0/2.800

DHCP client
- GigE 2.800
Machine Learning & Big Data
Machine Learning

How can you know in advance what your network state will be?

Collect | Observe | Alert | Take action

Learn | Predict events
**Machine Learning**

In a nutshell

- Inspired from the human learning process

  Learn from *experience* $E$ some *task* $T$ with *performance* $P$

- Offspring of probabilities, statistics and computer science

- All about data: an optimized approach to deduce correlation of data points

<table>
<thead>
<tr>
<th>Supervised Learning</th>
<th>Reinforcement Learning</th>
<th>Unsupervised Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Labeled Training data</strong></td>
<td><strong>Decision process with Reward system</strong></td>
<td><strong>Unlabeled Training data</strong></td>
</tr>
<tr>
<td>• Classification</td>
<td></td>
<td>• Clustering</td>
</tr>
<tr>
<td>• Regression</td>
<td></td>
<td>• Regression</td>
</tr>
</tbody>
</table>
Machine Learning
Applications

- Value prediction
- Computer Vision
- Object recognition
- Network security
- Signal processing
- Automated speech recognition
- Artificial Intelligence
- Recomender Systems
- Optmization
- Search engines
- Gaming
- Healthcare
- Bioinformatics sequences
- Brain-computer interfaces

Source: Google Trends
Machine Learning

Open questions for a network maintainer

- How does the load of my network evolve and where?
- What is the timeline for capacity upgrade?
- When can I upgrade a routing device software?
- When do I need traffic routing re-optimization?
- What is the impact of maintaining an optical fibre?
- Can I guarantee bandwidth?
Cisco WAN Automation Engine (WAE)
WAE provides tools for Network Optimization and Automation

- Predictive Model
  - Modeling
  - What if/predictive analysis
  - Global optimization

- Time Series Visibility
  - Assess historical and real-time data
  - Find and manage hot spots
  - Network efficiency analysis

- Model-Based Control and Configuration
  - Programmatic network control
  - Extensible, open data models

- Optimization and Automation
  - Real-time traffic balancing
  - Intelligent bandwidth scheduling
  - Automated service delivery
Proposal in the context of Network Monitoring

**Forecasting tool** predicting future network traffic, with key characteristics such as trend, seasonality and periodicity as well as localization, such as access, edge and core. Complements Cisco WAE solution.

Delivers operational efficiency

Allows network optimization

Provides services differentiation
Methodology
Main components of the development

- Explore statistical characteristics
- Identify roadblocks

Data access & understanding

- Solve missing data
- Apply transforms
- Extract features

Data preprocessing

- Assess performance on unseen data
- Check for robustness and lack of bias

Testing & Forecasting

- Validate the method

Modelling & Training

Traffic forecasting

- Iterative process
- Deep learning
- Validate the method
Performance and Forecasting

LSTM Forecasting

- Original Traffic [Mbps]
- Forecasted Traffic [Mbps]
Performance and Forecasting

LSTM Forecasting

K-fold Training Score

<table>
<thead>
<tr>
<th>3-month dataset</th>
<th>MSE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Traffic [Mbps]</td>
<td>0.00166</td>
<td>7461821.55 Mbps²</td>
</tr>
<tr>
<td>Forecasted Traffic [Mbps]</td>
<td>0.04072</td>
<td>2731.63 Mbps</td>
</tr>
</tbody>
</table>

Test Score

<table>
<thead>
<tr>
<th>3-month dataset</th>
<th>MSE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Traffic [Mbps]</td>
<td>0.00135</td>
<td>6805089.28 Mbps²</td>
</tr>
<tr>
<td>Forecasted Traffic [Mbps]</td>
<td>0.03681</td>
<td>2608.65 Mbps</td>
</tr>
</tbody>
</table>
Outlook

• Enabled pattern identification of network traffic
• Reliable anticipation of events in the network
• A complementary tool for intelligent network adaptation

Observe, learn and understand the network state

Adapt network based on an intelligent decision-making process
## Big Data

### Motivation and role

We can learn, but how do we improve our speed and scale?

<table>
<thead>
<tr>
<th>Trends</th>
<th>Areas of Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increased network capacity and bandwidth</td>
<td>Complexity of the network</td>
</tr>
<tr>
<td>Cloud services</td>
<td>Policy inconsistency</td>
</tr>
<tr>
<td>Changing traffic patterns</td>
<td>Inability to scale</td>
</tr>
<tr>
<td>Continuous flow of data (application content and services)</td>
<td>Lack of high-performance data processing</td>
</tr>
</tbody>
</table>

✓ Distributed training per network node or interface

✓ Grid-based search for a model that optimally captures the features of the data
Big Data
In a nutshell

Not only a vast amount of data…

• Various data sources
• Collection with high velocity
• Large amounts of data
• High granularity of the data
• Application of high-performance algorithms

Rather, a set of technologies that support large-scale collection and provide a platform for enabling meaningful insight to data.
Big Data
Solution: Platform for Network Data Analytics (PNDA)

PNDA as a data platform

- Open source
- Data aggregation with high throughput with Apache Kafka
- Environment for OSS data storage and exploration
- Parallel processing for rapid operations
- Supports batch processing applications
- Predictive analysis on time-series data

http://www.pnda.io/
Agenda

- Introduction
- Demo: Link-quality assurance with Model-Driven Telemetry
- Streaming Telemetry data from a device to a collector
- Storing and processing Telemetry data in a time series platform
- Visualizing data and reacting to events
- Machine Learning & Big Data use cases
- Demo: Telemetry-based vBNG address pool management
- Conclusion
Cisco Spark

Questions?
Use Cisco Spark to communicate with the speaker after the session

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

cs.co/ciscowebot#BRKNMS-3537
• Please complete your Online Session Evaluations after each session

• Complete 4 Session Evaluations & the Overall Conference Evaluation (available from Thursday) to receive your Cisco Live T-shirt

• All surveys can be completed via the Cisco Live Mobile App or the Communication Stations

Don’t forget: Cisco Live sessions will be available for viewing on-demand after the event at www.ciscolive.com/global/on-demand-library/.
Continue Your Education

- Recreate the first demo in your lab! Get the code at https://github.com/sbyx/telemetry-breakout
- Try it out by yourself in our instructor-led lab tomorrow from 09.00-13.00!
- Learn more about Telemetry, Containers and other IOS XR 6.x features and visit @xrdocs on Github! https://xrdocs.github.io

- Demos in the Cisco campus
- Walk-in Self-Paced Labs
- Lunch & Learn
- Meet the Engineer 1:1 meetings
Main Message

Telemetry takes you from monitoring the network to understanding it.