



## *NOMS 2014 Tutorial Sessions*

### **TUTORIAL T2**

Monday, May 5, 2014

## **Manageability, Embedded Automation, Network Programming, and Autonomic Networking – Contradicting Concepts or Complementary Evolutions?**

*Michael Behringer, Patrick Charretour,  
Joe Clarke, Bruno Klauser, Jason Pfeifer,  
Cisco Systems, U.S.A., France, Switzerland*

NOMS 2014 Tutorial Co-chairs:  
Mehmet Ulema, Manhattan College, U.S.A.  
Burkhard Stiller, University of Zürich, Switzerland

# Manageability, Embedded Automation, Network Programming and Autonomic Networking – Contradicting Concepts or Complementary Evolutions?

Authors: Michael Behringer, Patrick Charretour, Joe Clarke, Bruno Klauser, Jason Pfeifer,  
Cisco Systems {mbehring | pcharret | jclarke | bklauser | jpfeifer}@cisco.com

# Manageability, Embedded Automation, Network Programming and Autonomic Networking – Contradicting Concepts or Complementary Evolutions?



Bruno Klauser

Consulting Engineer  
BN EMEAR CTO Team  
bklauser@cisco.com



Jason Pfeifer

Technical Marketing Engineer  
NOSTG Technical Marketing  
jpfeifer@cisco.com



Michael Behringer

Distinguished Systems  
Engineer  
BN EMEAR CTO Team  
mbehring@cisco.com



Joe Clarke

Distinguished Services  
Engineer  
Global Technical Center  
jclarke@cisco.com

# Agenda

- Welcome
- Industry Observations
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- Autonomic Networking
- Resources and References

# Welcome Aboard

## This Session IS

- About the Adoption of Manageability, Embedded Automation, Network Programming and Autonomic Networking
- Based on Real-Life Examples and Practitioner's Experience

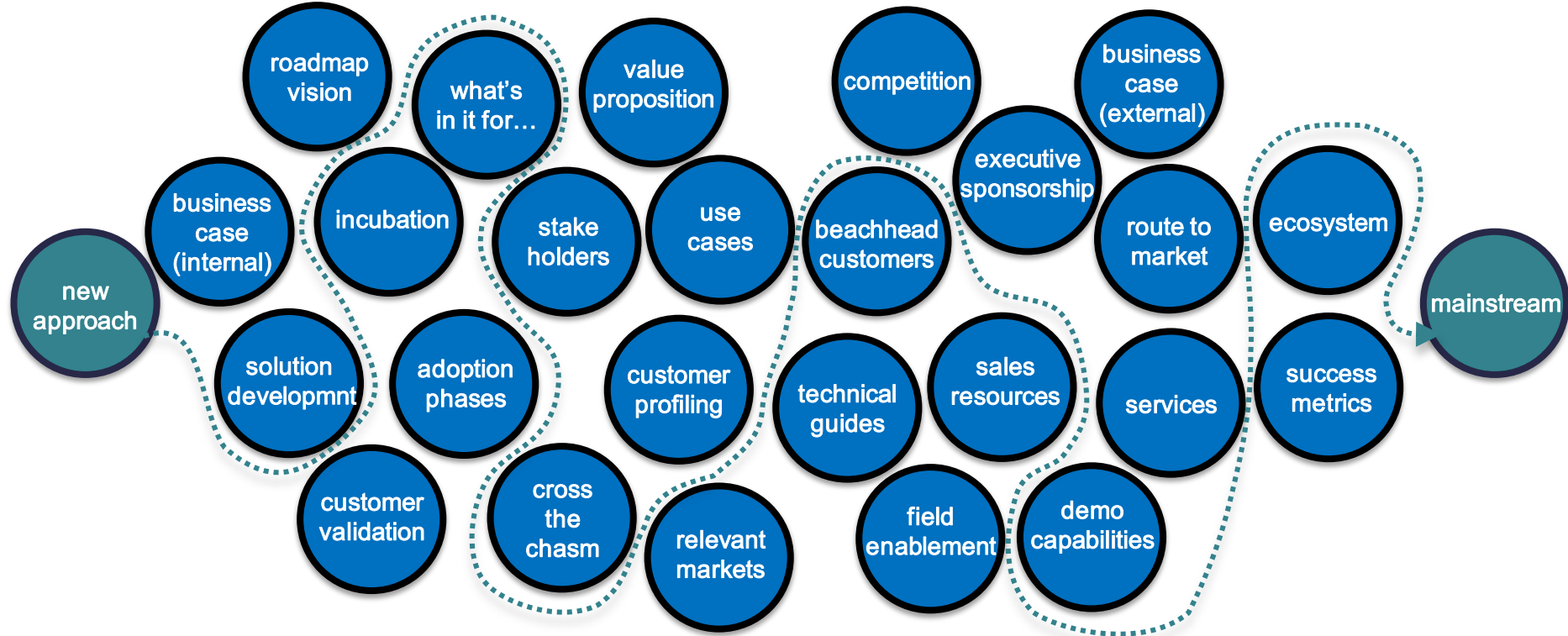
## This Session IS NOT

- An Introduction to SDN
- a deep dive of Standards, Protocols or Features

# Technology Incubation – 1/2



# Technology Incubation – 2/2



# Agenda

- Welcome
- **Industry Observations**
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- Autonomic Networking
- Resources and References

# Once Upon a Time ...



Applications were

- Monolithic
- Directly attached Storage
- Directly connected Terminal
- Local – Mainframe Room
- Static

# ... More Recently

Myriads of Things and Applications connected by the Network

Offices



Smart Industry



Smart Homes



Growing from Trillions to Tens of Trillions

Smart Agriculture



Smart Car



Smart Health



Growing from Billions to Trillions

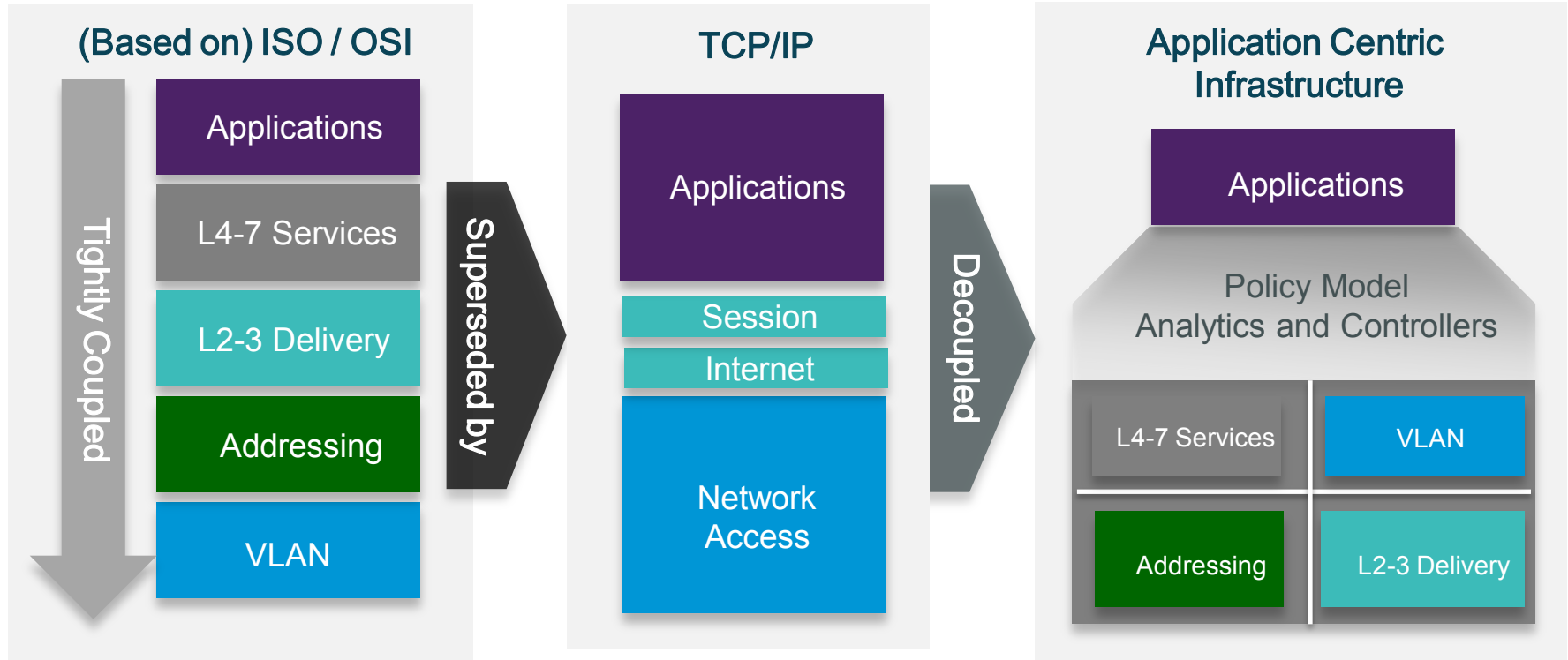
Applications are

- Distributed, 2-tier, 3-tier, n-tier
- Remote Users
- Remote Storage
- Remote Peers, Sensors, Actors, Sentinels, Satellites, Agents, ...
- Agile, Elastic

Source: Machina Research

© 2013-2014 Cisco and/or its affiliates. All rights reserved.

# Separation of Concerns



See also: <http://spectrum.ieee.org/computing/networks/osi-the-internet-that-wasnt>

© 2013-2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public

10

# Separation of Concerns – Layer 2 and ONF SDN

## What Is a Software Defined Network (SDN)?

“...In the SDN architecture, the **control and data planes are decoupled**, network intelligence and state are logically centralized, and the underlying network infrastructure is abstracted from the applications...”

Source: [www.opennetworking.org](http://www.opennetworking.org)

Intelligence

Connectivity

# SDN – Industry Initiatives

## What Is OpenFlow?

Open protocol that specifies interactions from the de-coupled control plane to the data plane



## What Is OpenDaylight?

Open-Source, **Community-Driven controller** for SDN Applications

Cisco are Co-Founder



## What is OpenStack?

**Opensource software** for building public and private Clouds - Compute, Networking and Storage.



## What is Overlay Network?

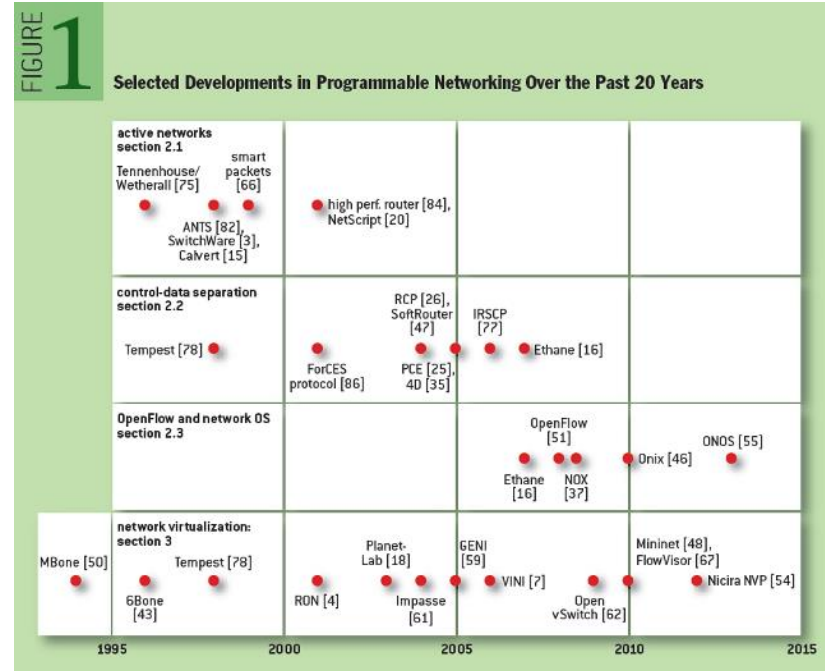
Overlay network is created on top of existing network infrastructure (physical and/or virtual) using a network protocol.

*“Computer programming is an art,  
because it applies accumulated  
knowledge to the world,  
because it requires skill and ingenuity,  
and especially  
because it produces objects of beauty”*

Donald Knuth, 1974  
(Author of “The Art of Computer Programming”)

# Emergence of Programmable Networks

- In Academia since the mid '90
  - Programmability
  - (Network Function) Virtualization
  - Initially Point Use Cases
- Definitions still evolving
- Common Concepts
- Bold Visions vs. Pragmatic Implementations



See also „The Road to SDN“ ACM Queue 2013: <http://queue.acm.org/detail.cfm?id=2560327>

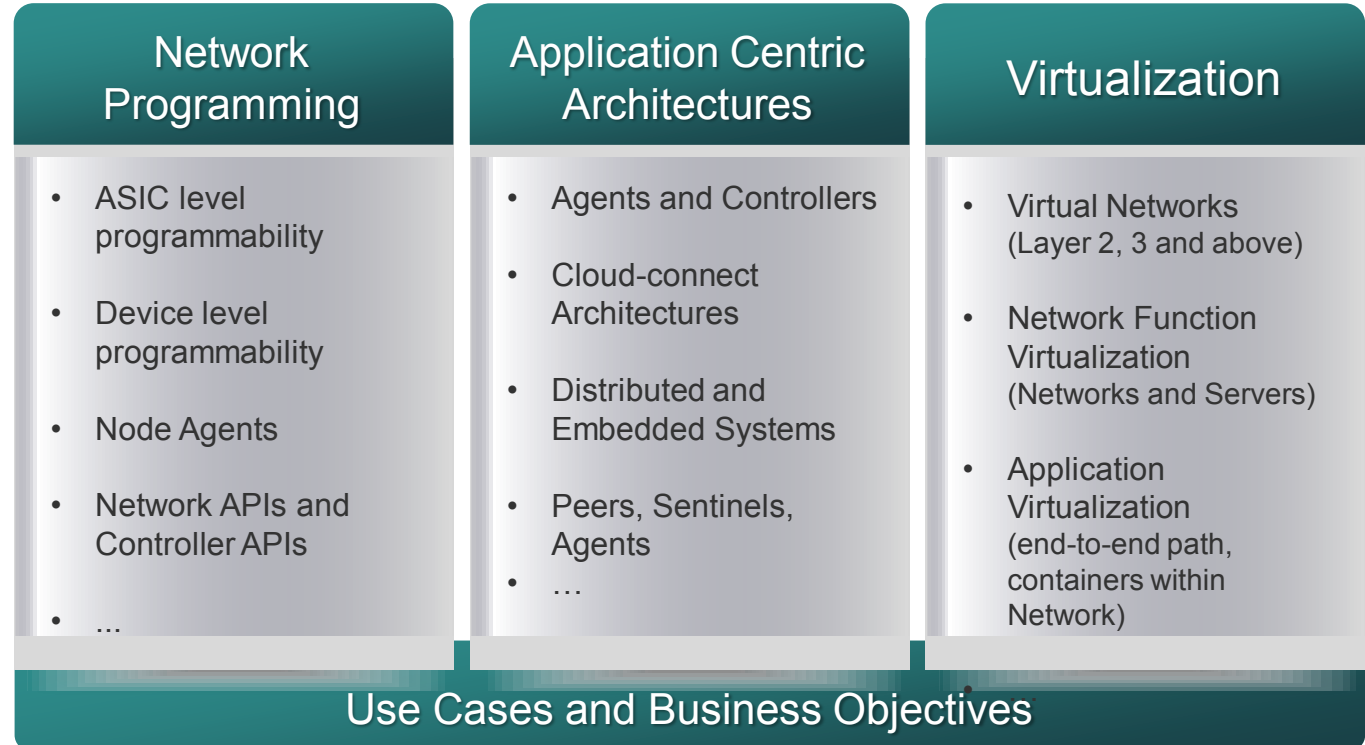
© 2013-2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public

14

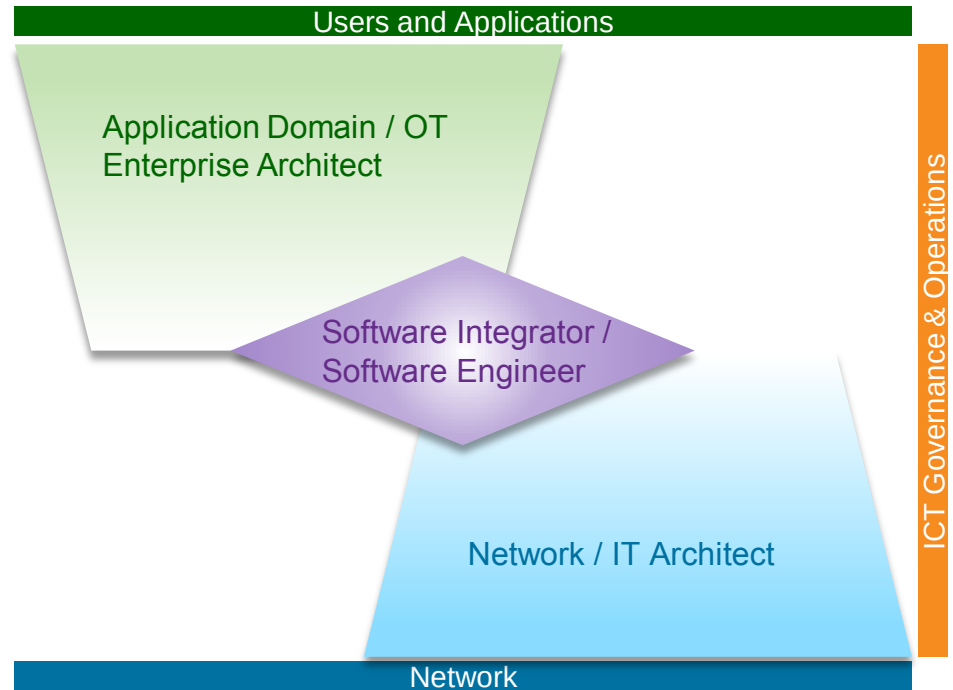
# Proliferation of 3 Main Concepts

- Common across SDN approaches
- Enabling capabilities
- Proliferating across domains

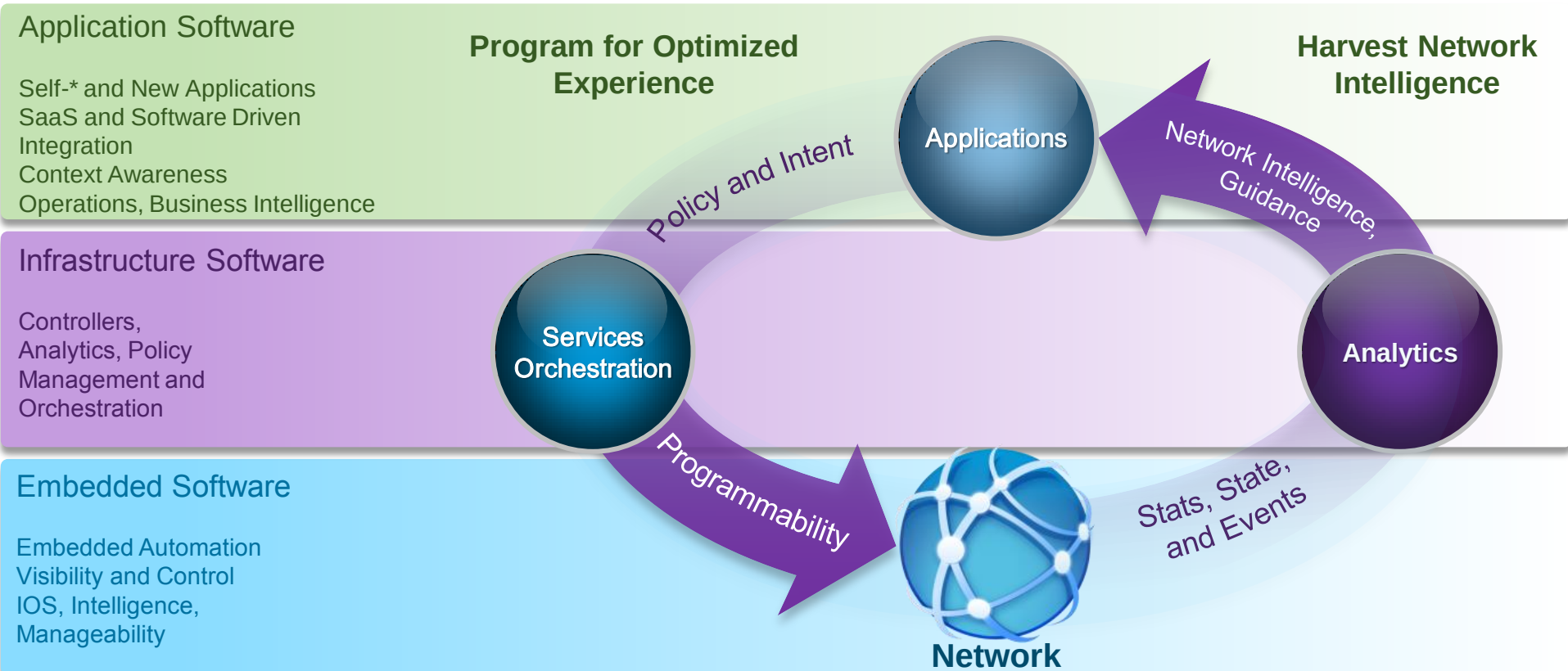


# How Many Architectures do You Build?

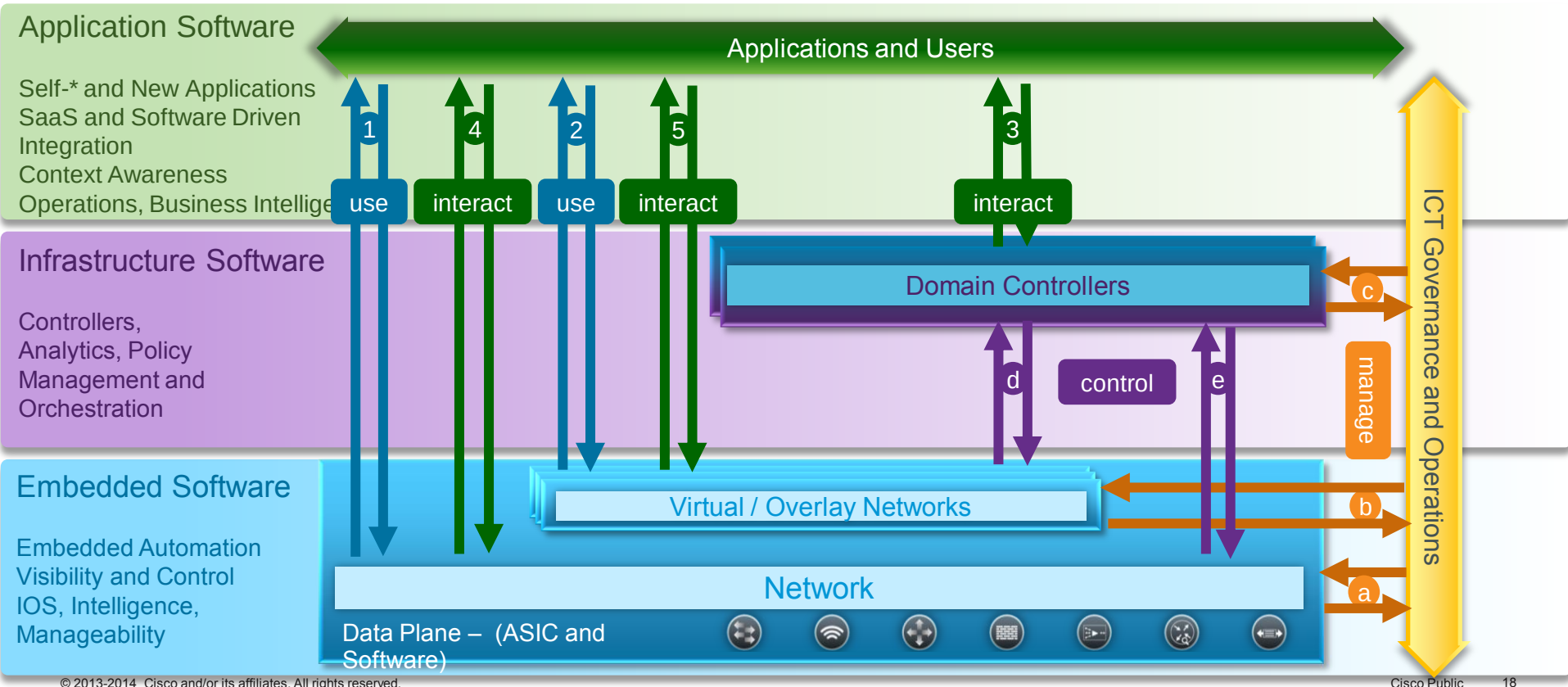
- Historically 3
  - Network and ICT
  - Business Applications / Operating Technology
  - ICT Governance and Operations
- Enabling Holistic Architectures
  - From System Integration towards APIs
  - From Declaration (Contracts) towards Introspection (Discovery)
  - From Engineering towards Runtime



# Abstractions and Programmatic Operations



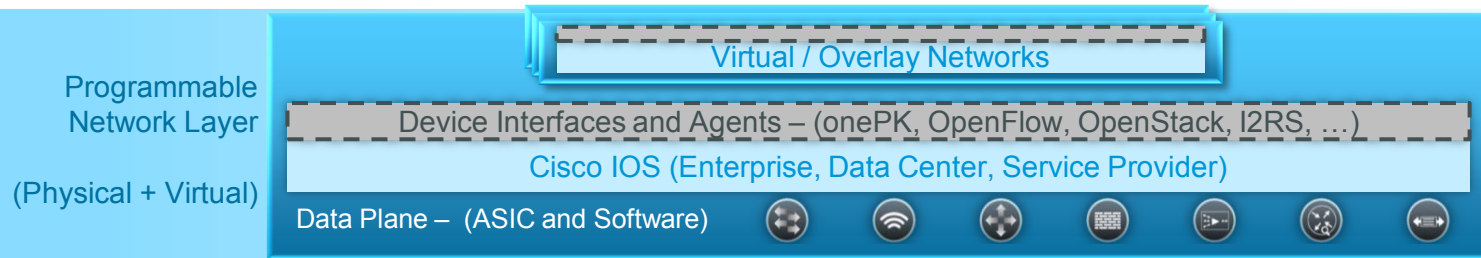
# Evolving Interactions



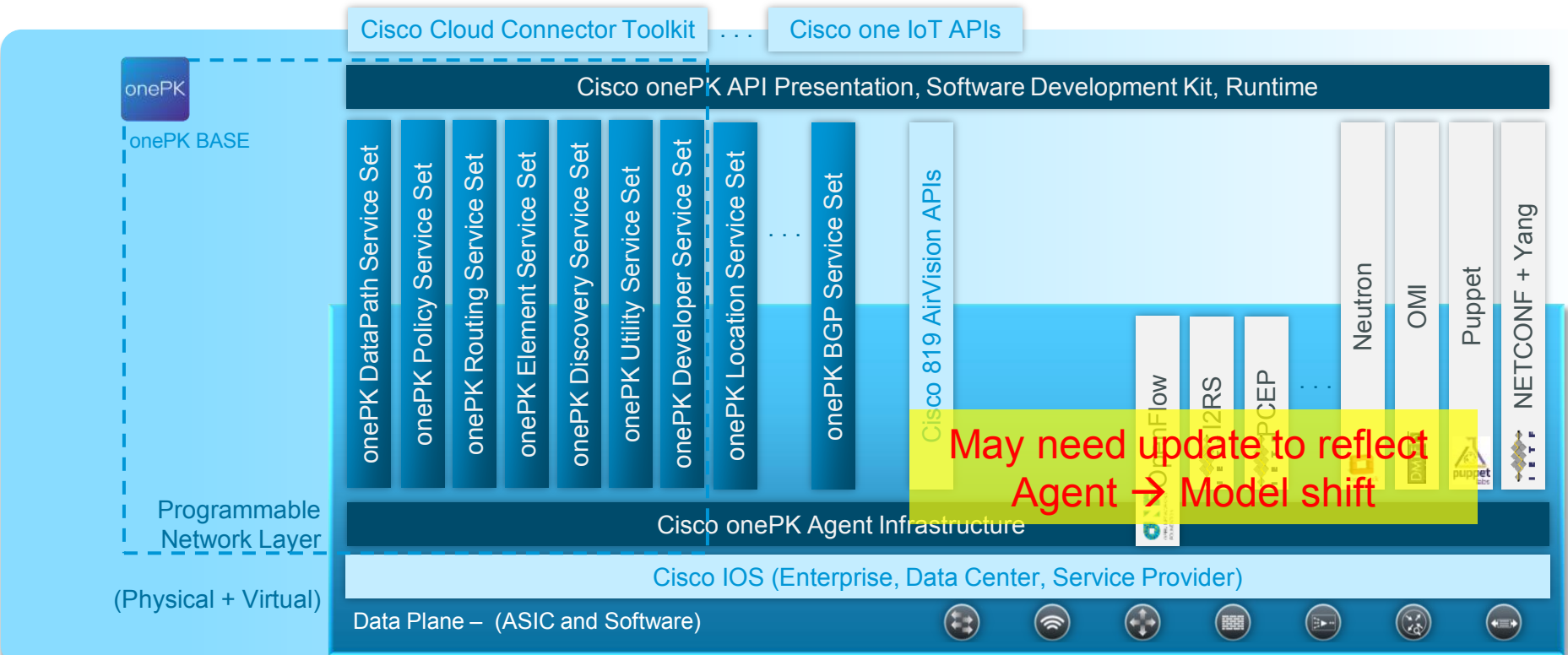
Cisco ONE



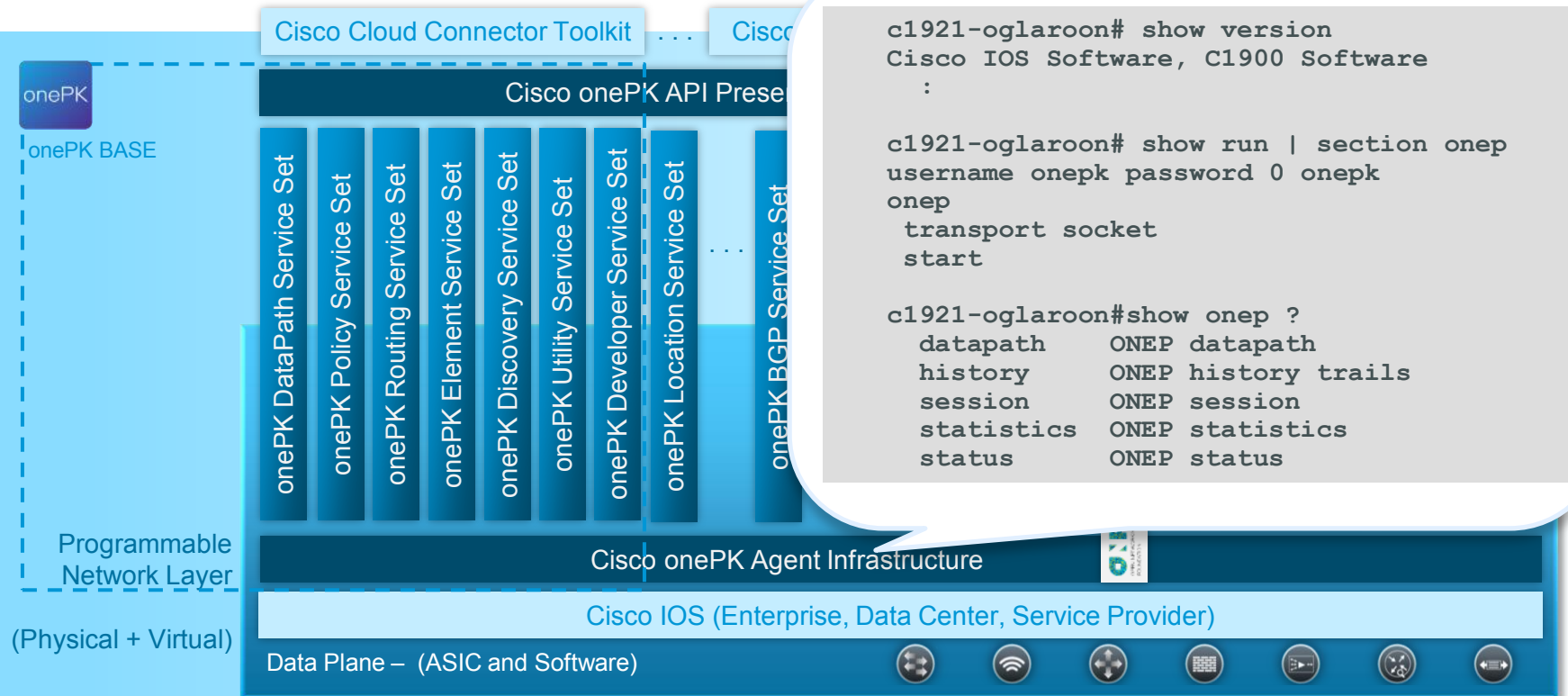
# Cisco ONE – Programmable Network Layer



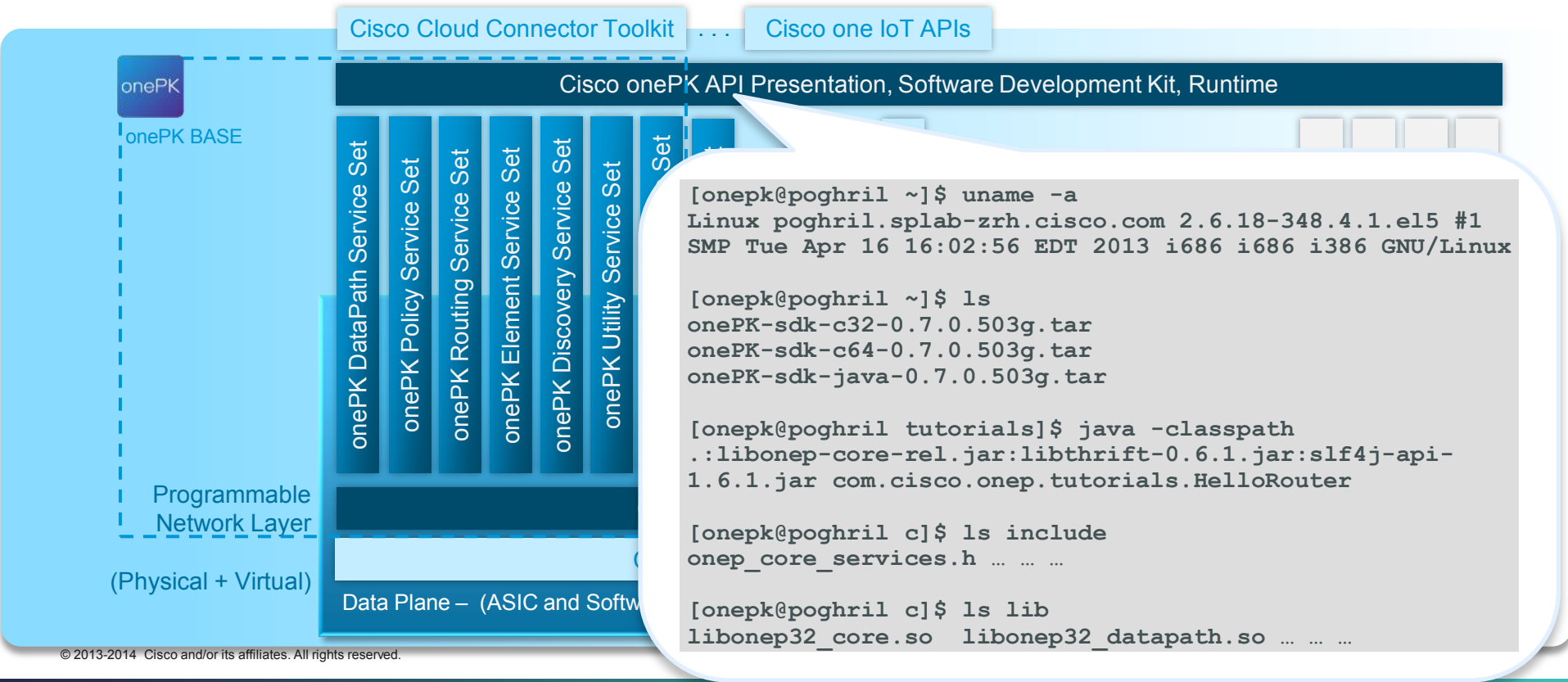
# Cisco ONE – Programmable Network Layer



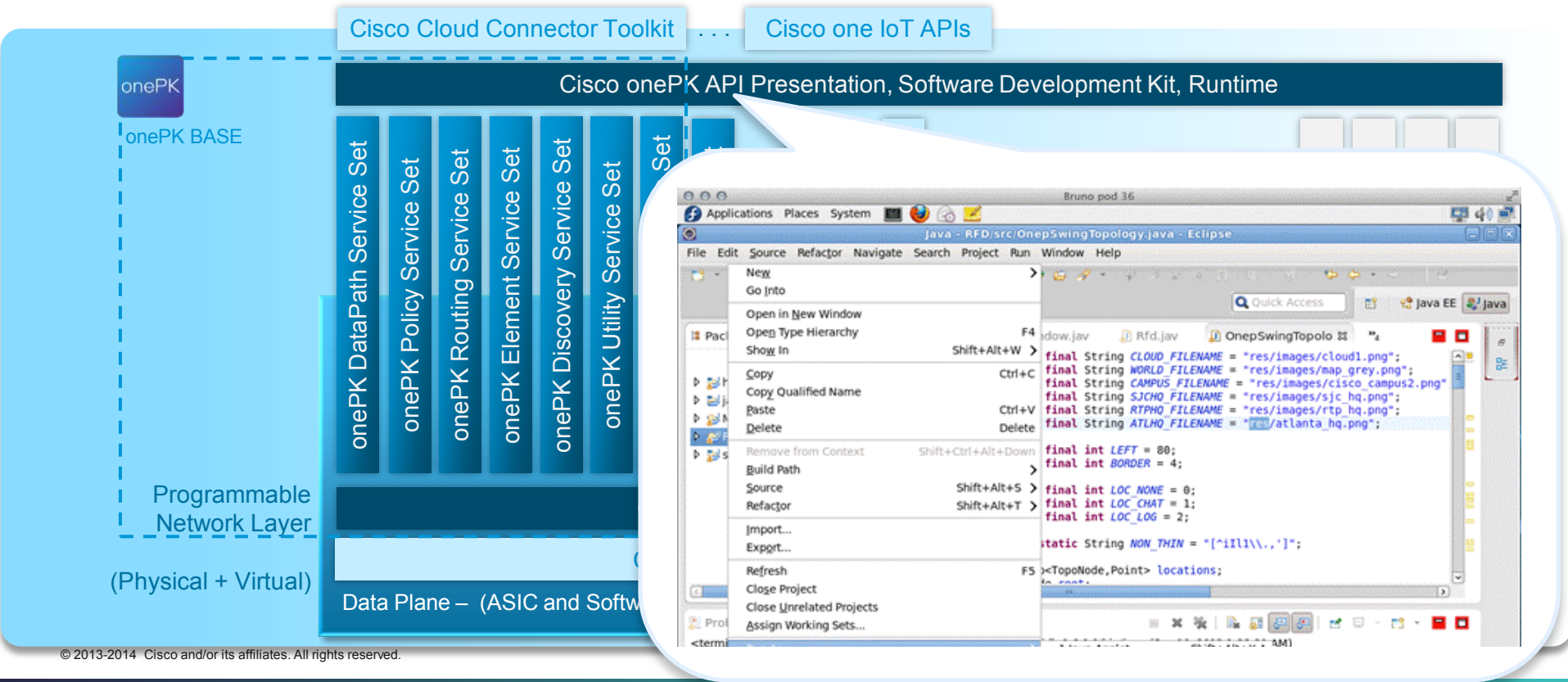
# Cisco ONE – Programmable Network Layer



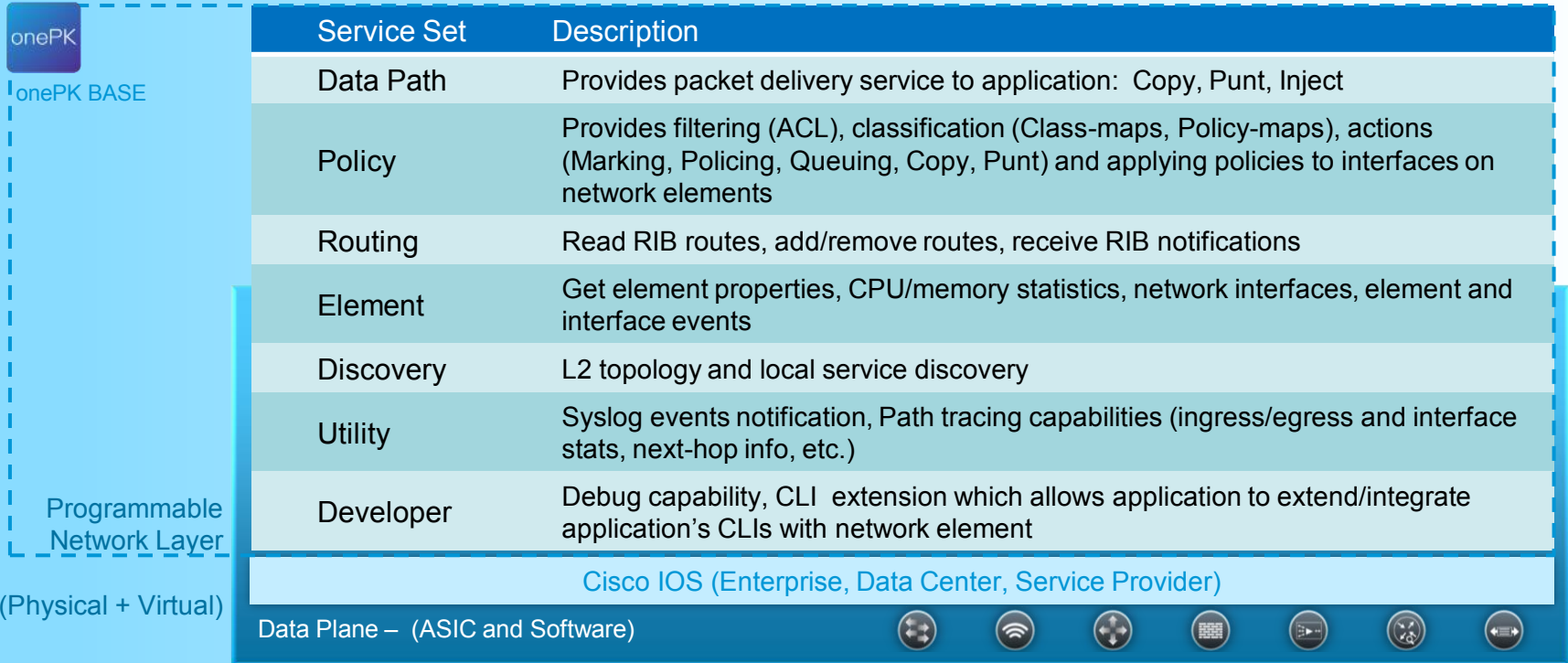
# Cisco ONE – Programmable Network Layer



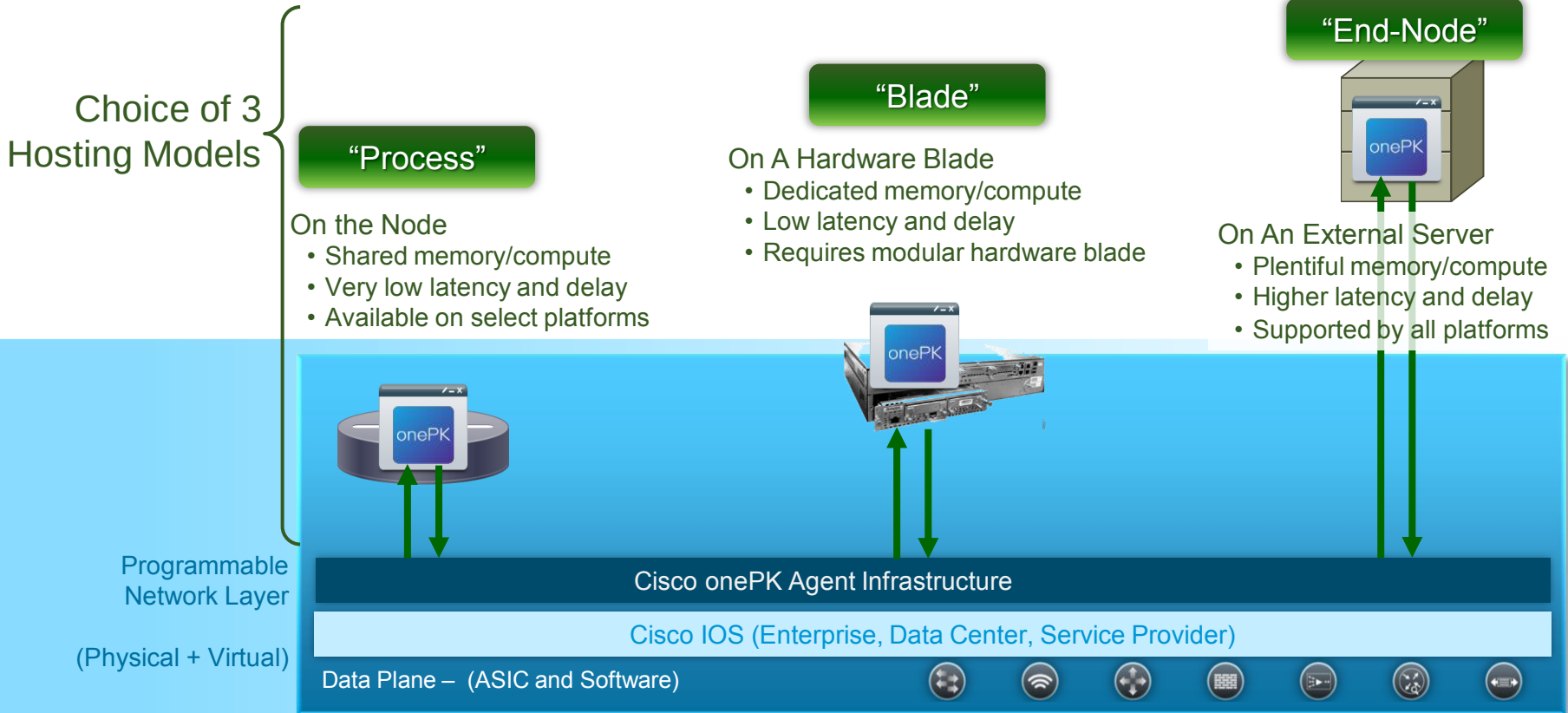
# Cisco ONE – Programmable Network Layer



# Cisco ONE – Programmable Network Layer – onePK



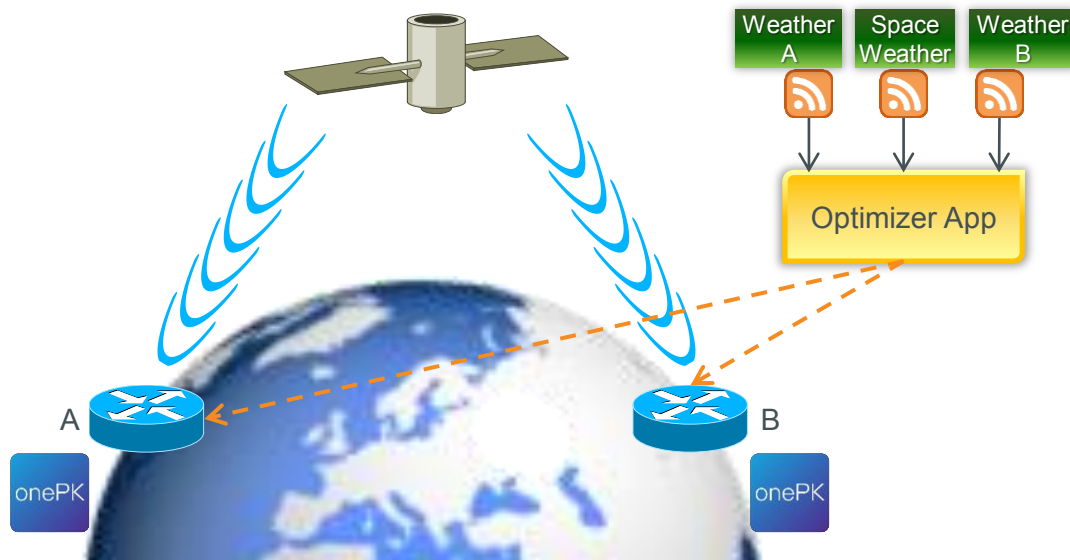
# Cisco ONE – Programmable Network Layer – onePK



# Example: Maximize Utilization of Dynamic Bandwidth

**Problem:** Bandwidth depends on ground weather and space weather, accurate short-term forecasts. How to optimize QoS configuration for dynamically changing bandwidth of satellite links?

**Solution:** Dynamically adjust ground station router config to maximize bandwidth



1. Acquire and Normalize Space Weather and Weather Conditions
2. Calculate optimal topology and configuration for upcoming period
3. Derive required configuration changes
4. Inject incremental changes and harvest required metrics
5. Repeat

# Cisco ONE – Cloud Services Router – CSR 1000v

## IOS-XE code base

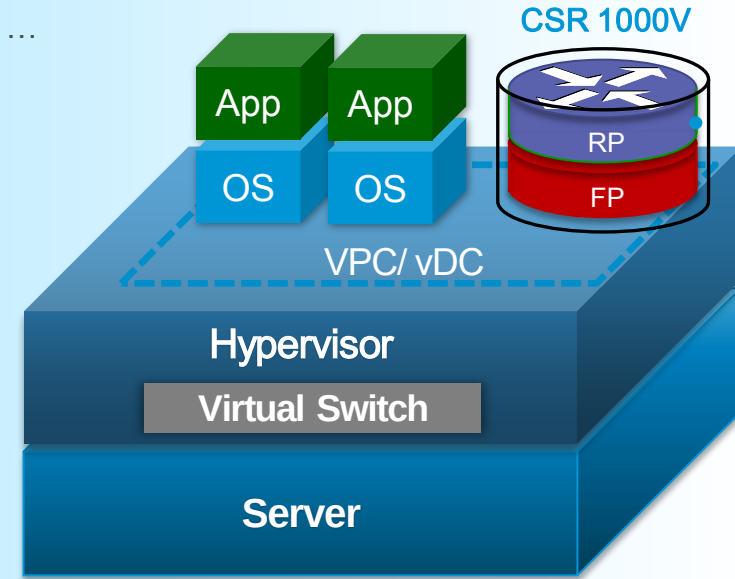
- Comprehensive feature set
- 4 month release cycle – 3.9 (March '13), 3.10 (July '13), 3.11 (Jan'14) ...

## Infrastructure Agnostic

- Cisco UCS, Dell, HP, etc.  
Intel and AMD processors supported
- Runs on vSwitch, dVS, N1KV, etc.. – no dependency
- VMware ESXi 5.1, Citrix Xen Server 6.1, KVM – RHEL 6.3, RHEV 3.1 supported
- Amazon AMI support in 3.11

## Footprint

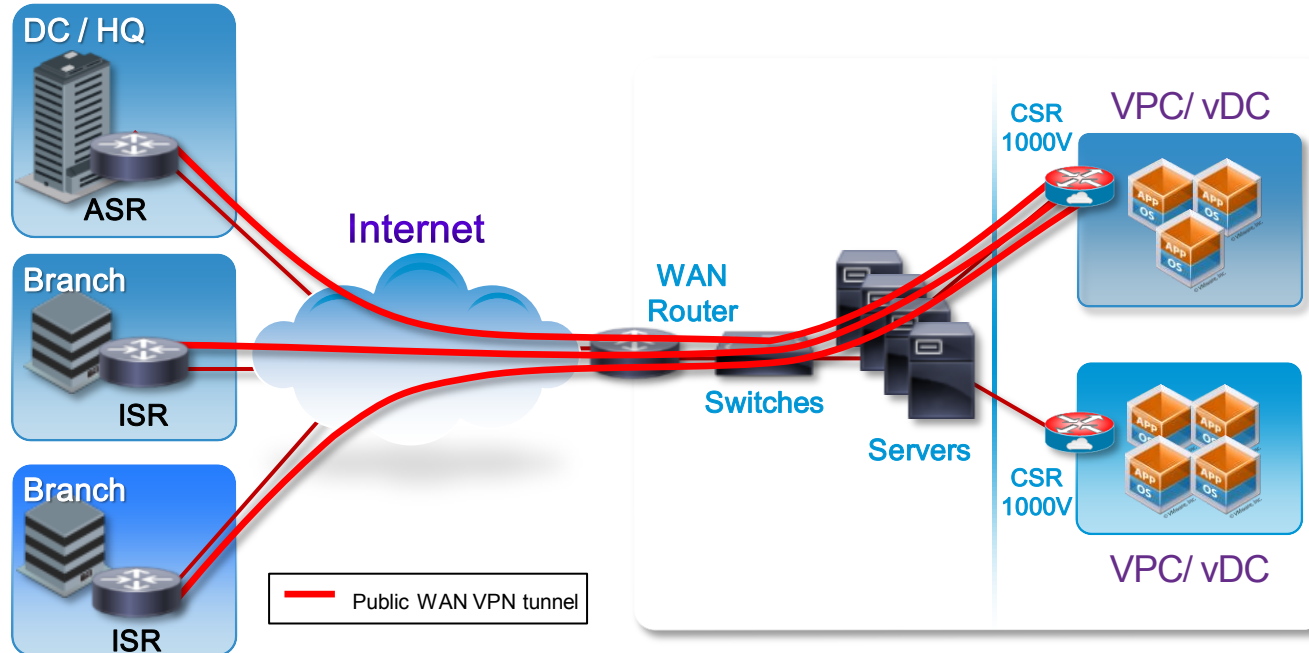
- 4 vCPU, 2 vCPU, 1vCPU supported.  
**Note:** 2 physical cores \* 2 = 4 vCPU with Hyperthreading
- 2.5 GB/1vCPU [default] , 4 GB/4vCPU
- 8 GB HD – Local, SAN, NAS supported



# Example: Secure VPN Gateway

**Problem:** How to securely connect to a virtual private cloud or virtual Data Centre where we can't deploy Hardware – across the public Internet?

**Solution:** Deploy VPN Gateway on Cloud Services Router 1000v



© 2013-2014 Cisco and/or its affiliates. All rights reserved.

## Challenges

- Inconsistent Security
- High Network Latency
- Limited Scalability

## Solutions

- IPSec VPN, DMVPN, EZVPN, FlexVPN
- Routing and Addressing
- Firewall, ACLs, AAA

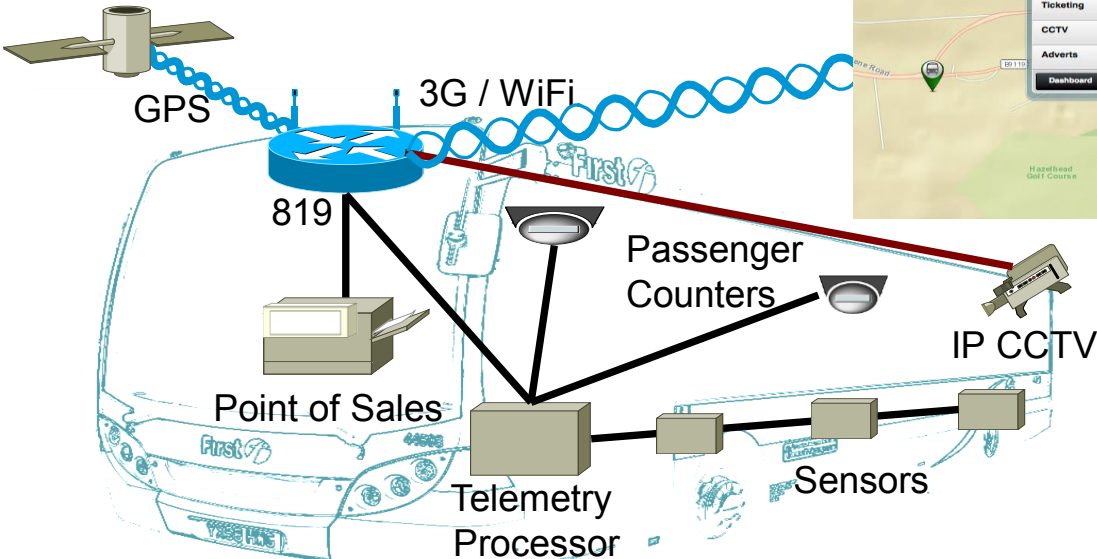
## Benefits

- Direct, Secure Access
- Scalable, Reliable VPN
- Operational Simplicity

# Example: Fleet Management and Telemetry

**Problem:** Fleet Management and public transport telemetry information

**Solution:** Use an ISR 819 to aggregate and communicate relevant onboard data

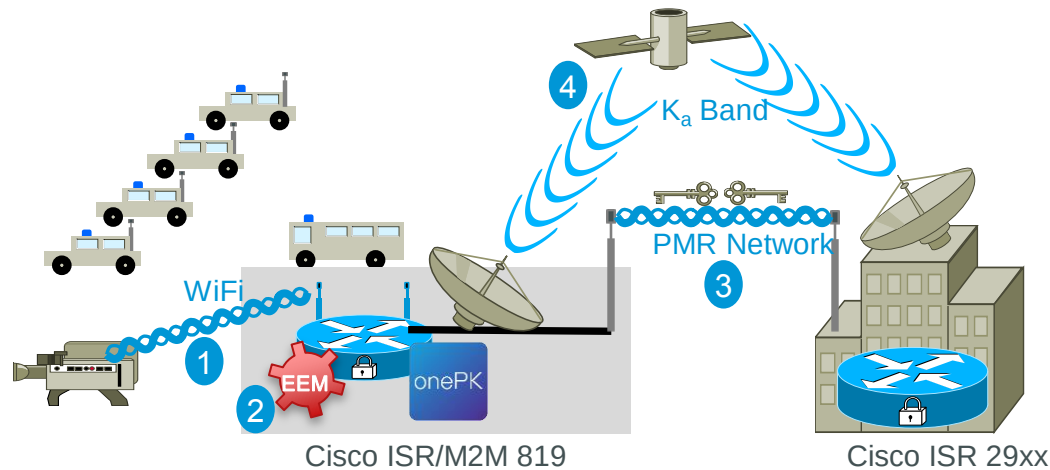


1. Provide onboard network via 819
2. Aggregate relevant data on 819 using custom Fleet Management Connector
3. Fleet/Bus state defines use of uplinks to Fleet Management Center

# Example: Emergency Response Network

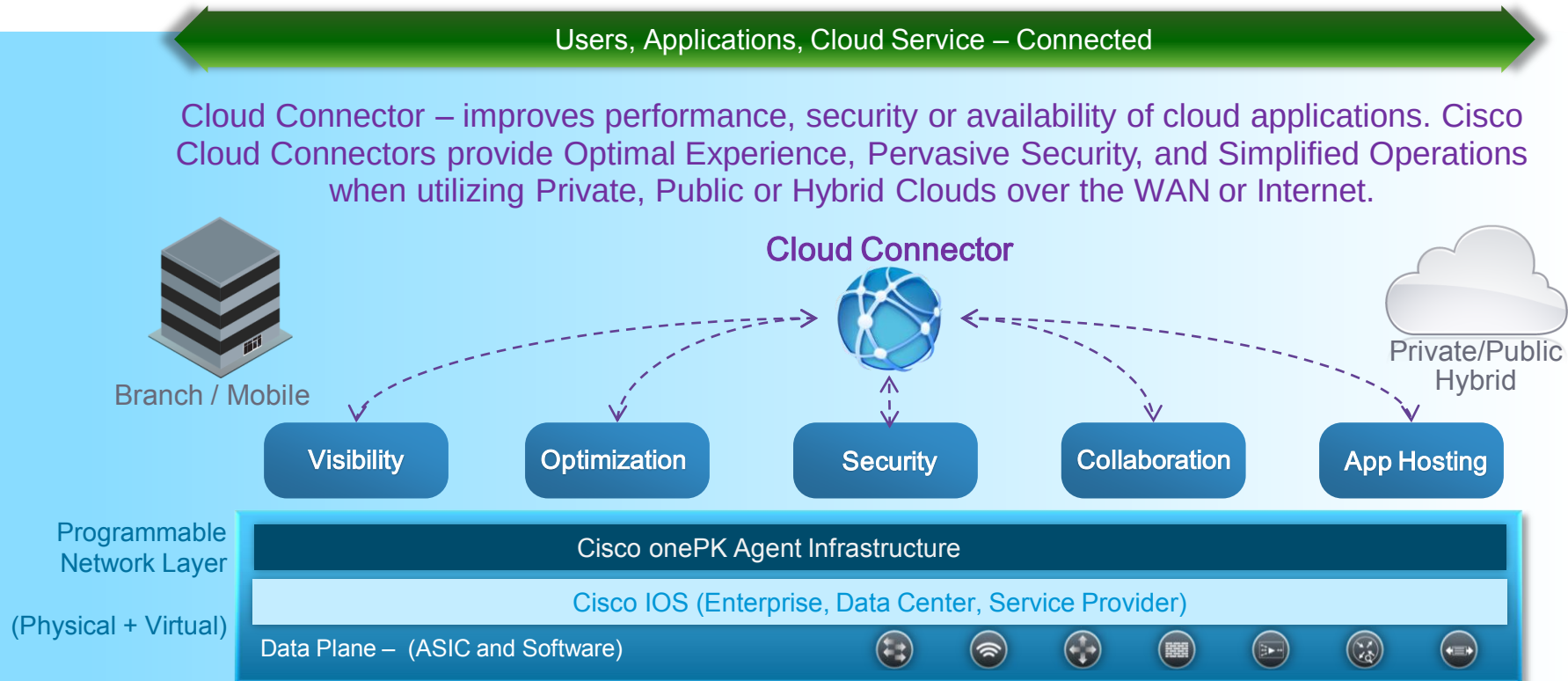
**Problem:** How to deliver secure, trusted, robust, cost-effective broadband connectivity to mobile emergency response units?

**Solution:** Use Network Programming based on Cisco onePK and Cisco IOS Embedded Event Manager to integrate low-cost, high-bandwidth options with accredited legacy radio connectivity:



1. Connect high-bandwidth forward clients via WiFi
2. Use Cisco IOS EEM for onboard system integration and adaptation
3. Use Cisco onePK to redirect IKE key exchange out-of-band via legacy radio
4. Secure IPsec tunnel via cost-effective high bandwidth K<sub>a</sub> Band
5. Reliable, secure emergency response network saving ~4M€ operating cost annually

# Emerging Design Pattern: Cloud Connect Architecture



# Cisco ONE – Controller Layer

Controller Layer  
(Orchestration + Analytics)

Programmable  
Network Layer  
(Physical + Virtual)

Virtual / Overlay Networks

Device Interfaces and Agents – (onePK, OpenFlow, OpenStack, I2RS, ...)

Cisco IOS (Enterprise, Data Center, Service Provider)

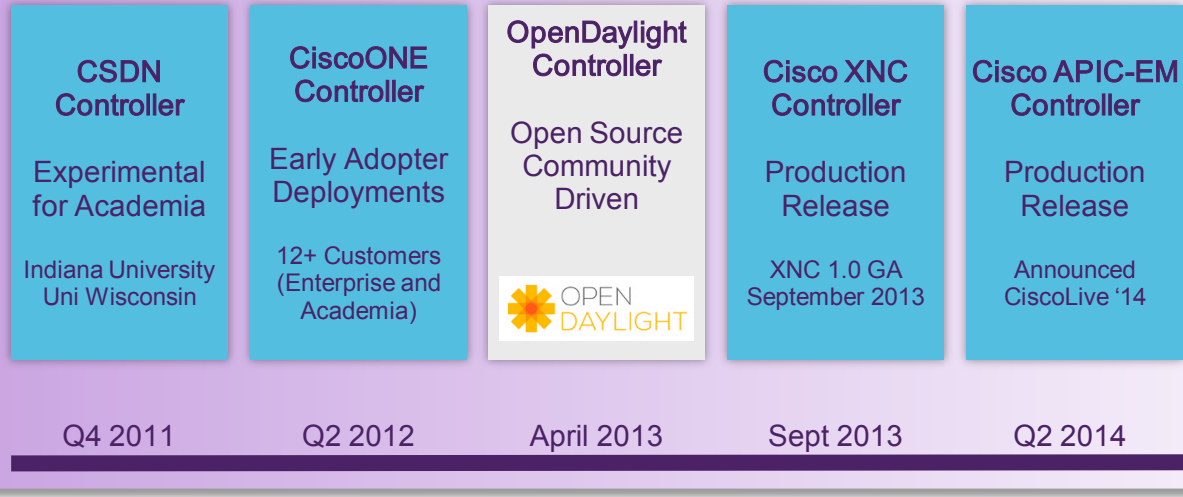
Data Plane – (ASIC and Software)



# Cisco ONE – Controller Layer

## Major Milestones of Controller Development

Controller Layer  
(Orchestration + Analytics)



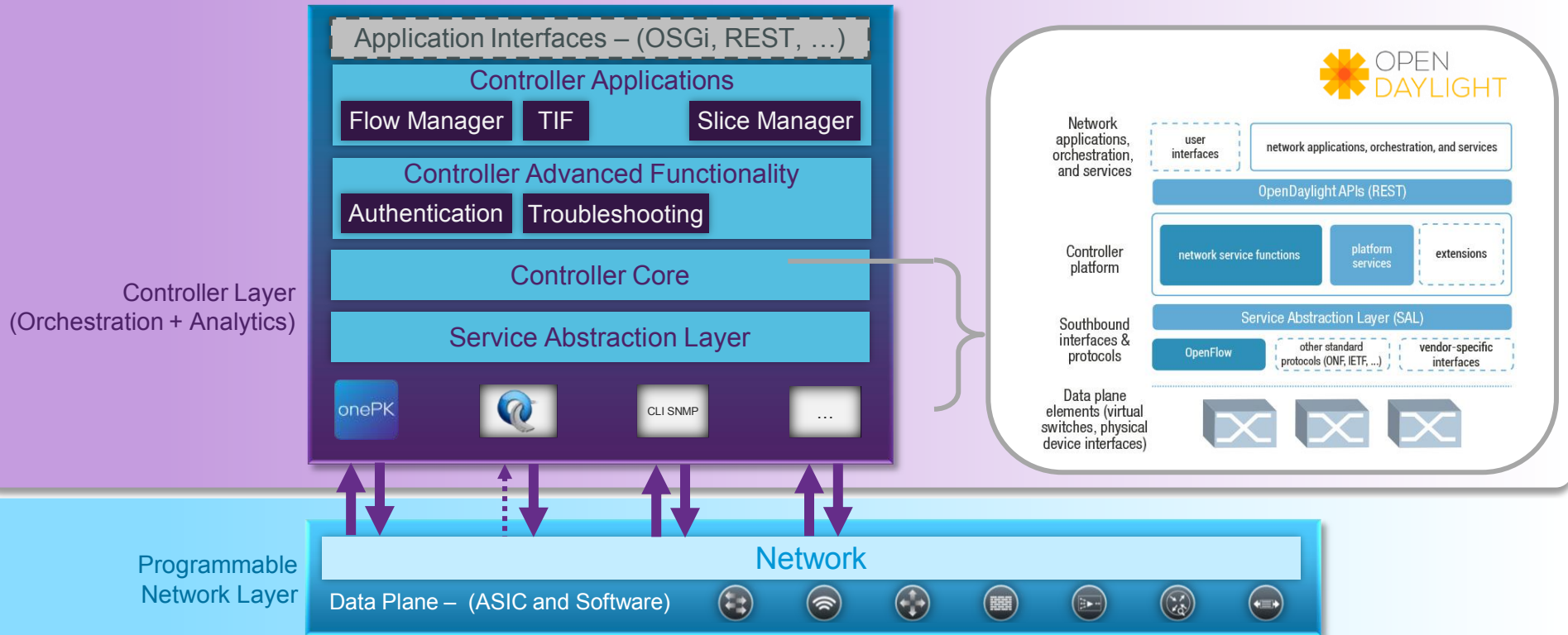
Programmable  
Network Layer

Network

Data Plane – (ASIC and Software)

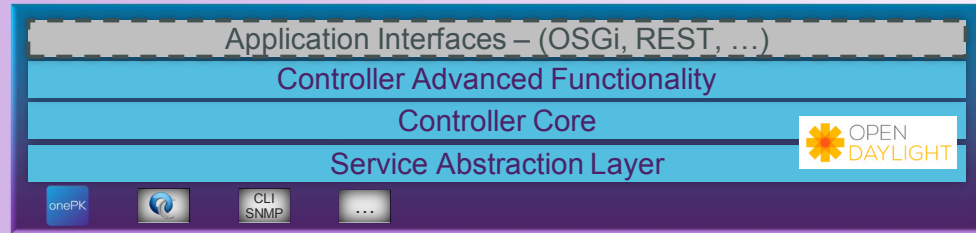


# Cisco ONE – Controller Layer

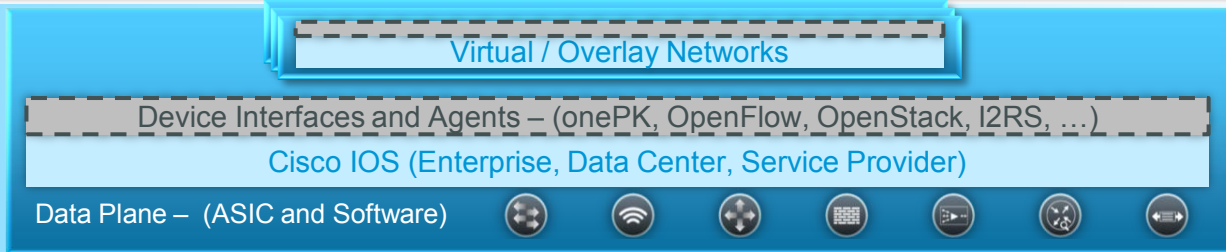


# Cisco ONE – Controller Layer

Controller Layer  
(Orchestration + Analytics)



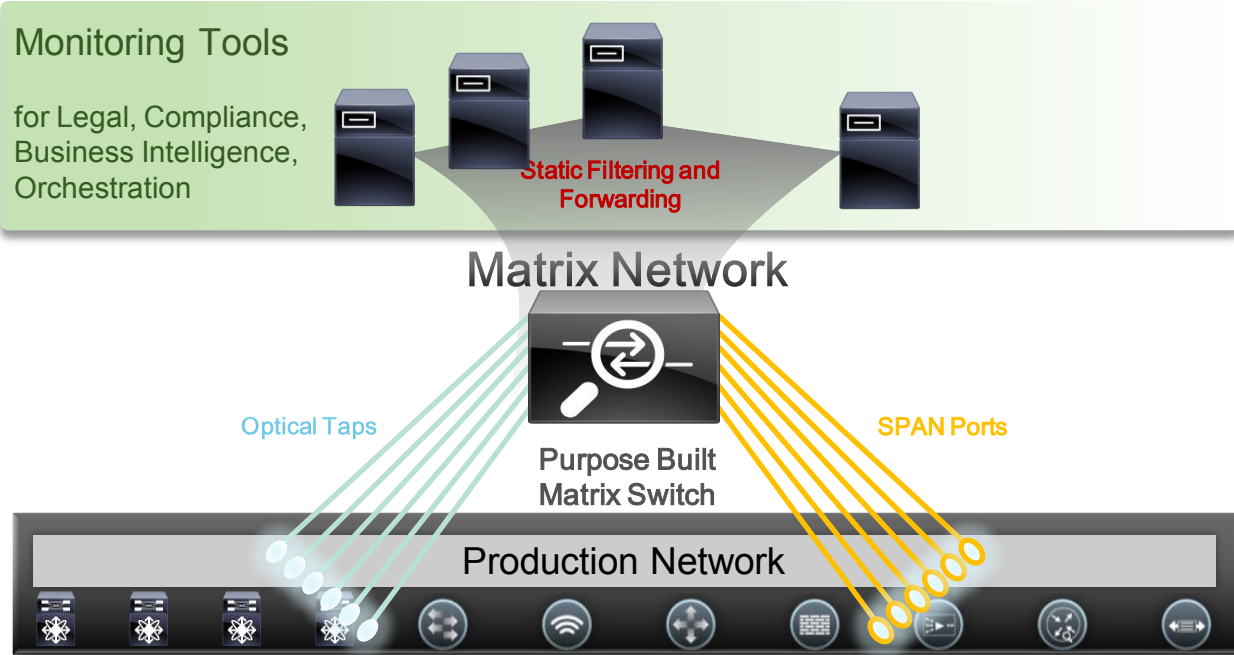
Programmable  
Network Layer  
(Physical + Virtual)



# Example: Monitor Manager Solution – 1/3

**Problem:** How to build and operate a flexible, programmable and cost-effective monitoring network?

**Traditional Approach:** Static, purpose-built Matrix Network



- Purpose-built specialized Equipment
- Engineering Integration
- Certified Tools
- Static Filtering and Forwarding
- ➔ Inflexible and expensive

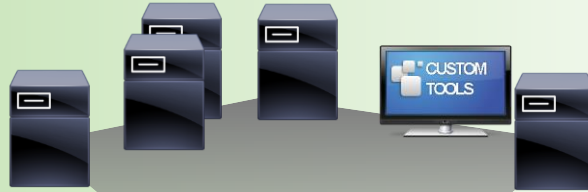
# Example: Monitor Manager Solution – 2/3

**Problem:** How to build and operate a flexible, programmable and cost-effective monitoring network?

**Solution:** Dynamic Monitor Manager Solution based on Cisco ONE Controller

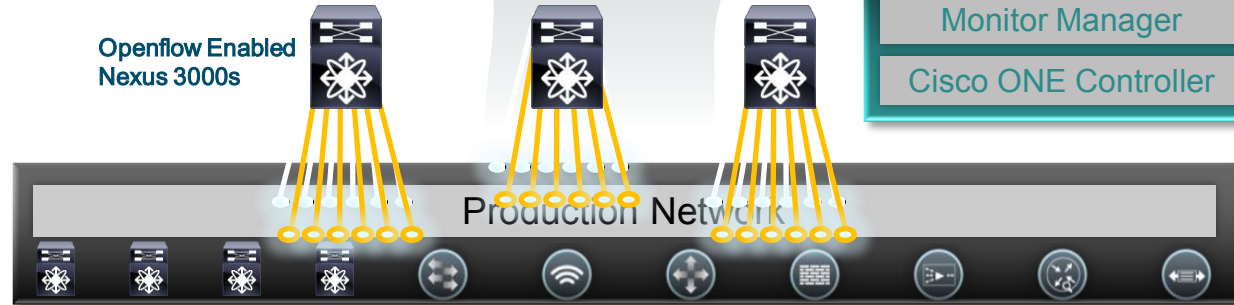
## Monitoring Tools

for Legal, Compliance,  
Business Intelligence,  
Orchestration



Dynamic Filter and Forwarding  
Event Driven / Real Time

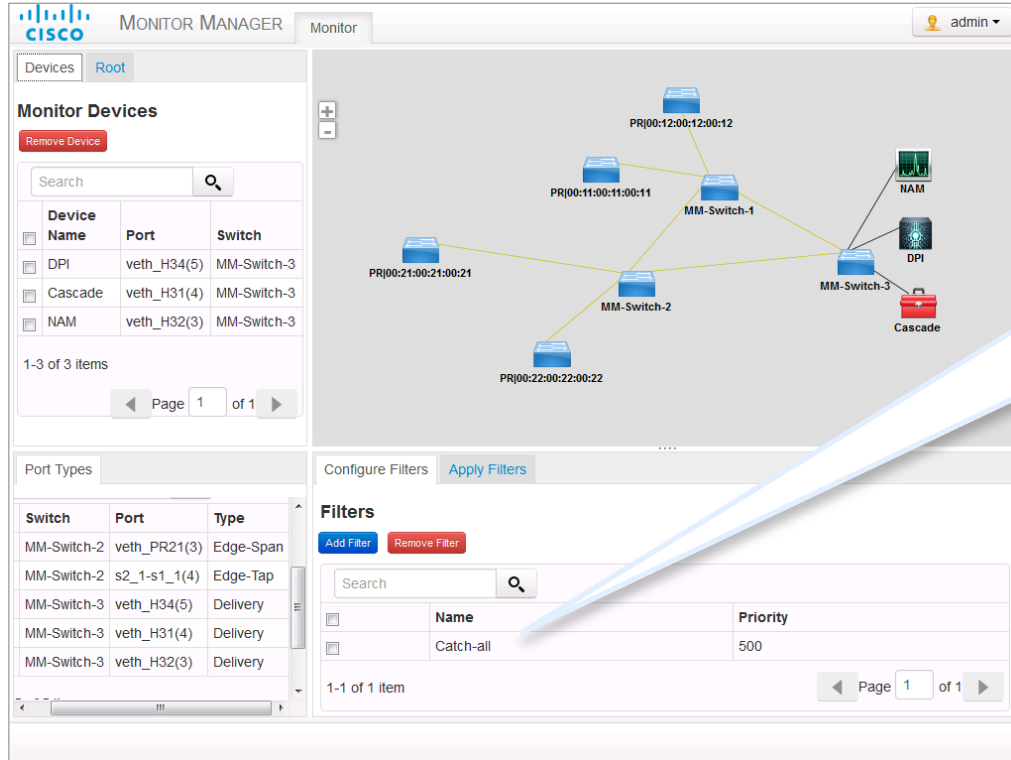
Openflow Enabled  
Nexus 3000s



- Agent Enabled general purpose Networking (Nexus 3000s initially)
- Operational Integration
- Open APIs and UI
- Dynamic Filtering and Forwarding

→ Agile and cost effective

# Example: Monitor Manager Solution – 3/3



The screenshot displays the Cisco Monitor Manager web interface. The top navigation bar includes the Cisco logo, the title "MONITOR MANAGER", and a "Monitor" tab. A user profile "admin" is visible in the top right. The left sidebar contains a "Monitor Devices" section with a "Remove Device" button and a search bar. Below this is a table of monitored devices:

| Device                   | Name    | Port        | Switch      |
|--------------------------|---------|-------------|-------------|
| <input type="checkbox"/> | DPI     | veth_H34(5) | MM-Switch-3 |
| <input type="checkbox"/> | Cascade | veth_H31(4) | MM-Switch-3 |
| <input type="checkbox"/> | NAM     | veth_H32(3) | MM-Switch-3 |

Below the table, it indicates "1-3 of 3 items" and a pagination control showing "Page 1 of 1".

The main area shows a network topology diagram with several switches (MM-Switch-1, MM-Switch-2, MM-Switch-3) and endpoints (PRj00:12:00:12, PRj00:11:00:11:00:11, PRj00:21:00:21:00:21, PRj00:22:00:22:00:22, NAM, DPI, Cascade). A large blue arrow points from the "Filters" section to the "1) Create a Filter for relevant Traffic" callout.

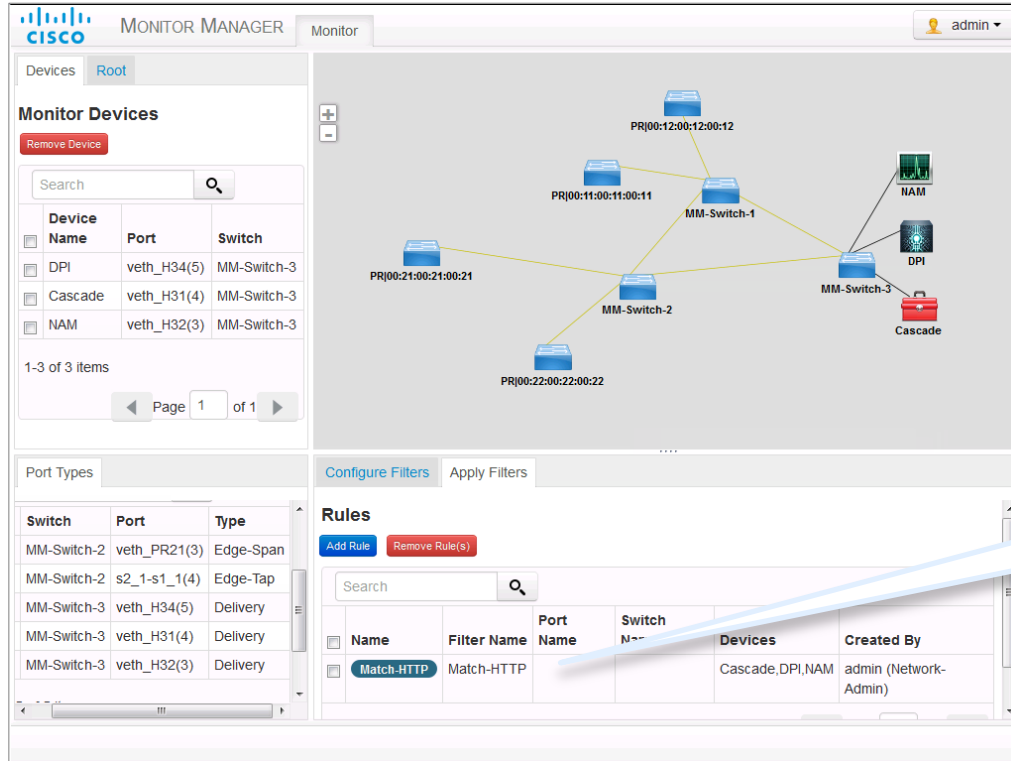
The bottom section, titled "Configure Filters", includes an "Apply Filters" button and a "Filters" table:

|                          | Name      | Priority |
|--------------------------|-----------|----------|
| <input type="checkbox"/> | Catch-all | 500      |

Below the table, it indicates "1-1 of 1 item" and a pagination control showing "Page 1 of 1".

1) Create a Filter for relevant Traffic

# Example: Monitor Manager Solution – 3/3



The screenshot displays the Cisco Monitor Manager web interface. The top navigation bar includes the Cisco logo, the title "MONITOR MANAGER", a "Monitor" tab, and a user profile for "admin".

**Monitor Devices Section:**

- Buttons: "Devices", "Root", "Remove Device".
- Search bar.
- Table:

| Device Name | Port        | Switch      |
|-------------|-------------|-------------|
| DPI         | veth_H34(5) | MM-Switch-3 |
| Cascade     | veth_H31(4) | MM-Switch-3 |
| NAM         | veth_H32(3) | MM-Switch-3 |

1-3 of 3 items  
Page 1 of 1

**Port Types Section:**

| Switch      | Port         | Type      |
|-------------|--------------|-----------|
| MM-Switch-2 | veth_PR21(3) | Edge-Span |
| MM-Switch-2 | s2_1-s1_1(4) | Edge-Tap  |
| MM-Switch-3 | veth_H34(5)  | Delivery  |
| MM-Switch-3 | veth_H31(4)  | Delivery  |
| MM-Switch-3 | veth_H32(3)  | Delivery  |

**Network Topology Diagram:** A central "MM-Switch-1" is connected to "MM-Switch-2" and "MM-Switch-3". "MM-Switch-2" is connected to ports "PRJ00:21:00:21:00:21" and "PRJ00:22:00:22:00:22". "MM-Switch-1" is connected to ports "PRJ00:12:00:12:00:12" and "PRJ00:11:00:11:00:11". "MM-Switch-3" is connected to "NAM", "DPI", and "Cascade".

**Configure Filters / Apply Filters Section:**

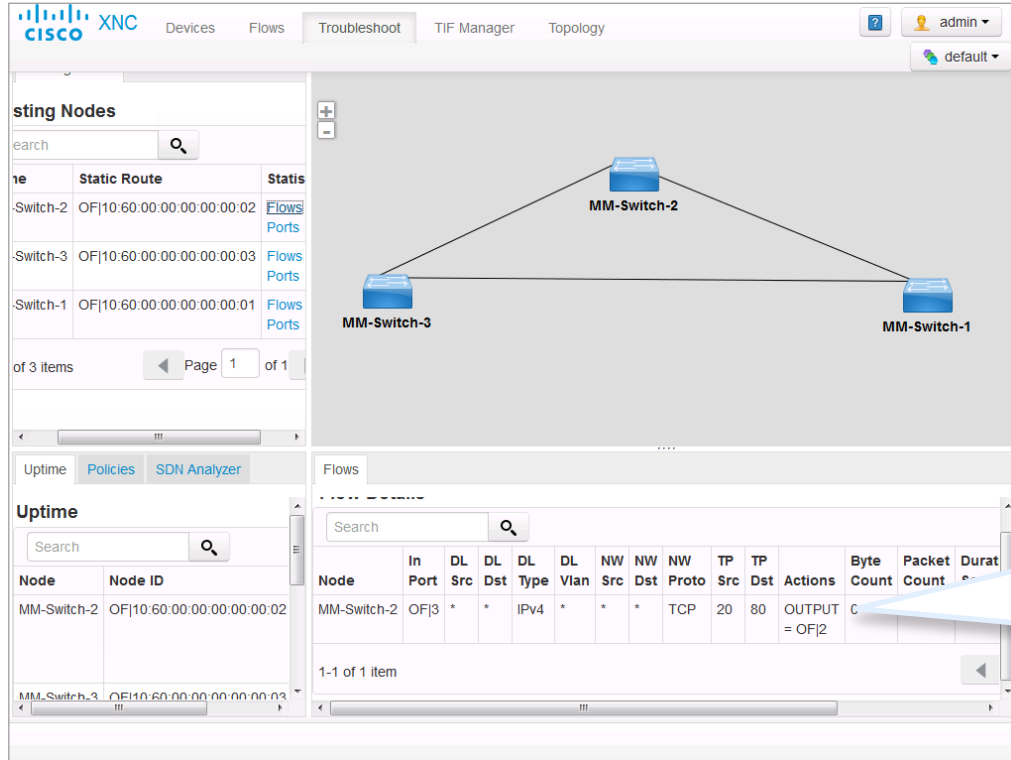
**Rules Table:**

| Name       | Filter Name | Port Name | Switch Name | Devices         | Created By            |
|------------|-------------|-----------|-------------|-----------------|-----------------------|
| Match-HTTP | Match-HTTP  |           |             | Cascade,DPI,NAM | admin (Network-Admin) |

1) Create a Filter for relevant Traffic

2) Apply Filter to Rule (Source / Destination Ports)

# Example: Monitor Manager Solution – 3/3



The screenshot displays the Cisco XNC Monitor Manager interface. The top navigation bar includes 'XNC', 'Devices', 'Flows', 'Troubleshoot', 'TIF Manager', and 'Topology'. The 'Topology' tab is active, showing a network diagram with three switches: MM-Switch-1, MM-Switch-2, and MM-Switch-3. MM-Switch-2 is at the top, connected to MM-Switch-1 and MM-Switch-3. MM-Switch-1 and MM-Switch-3 are also connected to each other. On the left, the 'Existing Nodes' table lists the switches with their static routes and status. Below it, the 'Uptime' section shows a table of node uptime. At the bottom, the 'Flows' section displays a table of traffic flows.

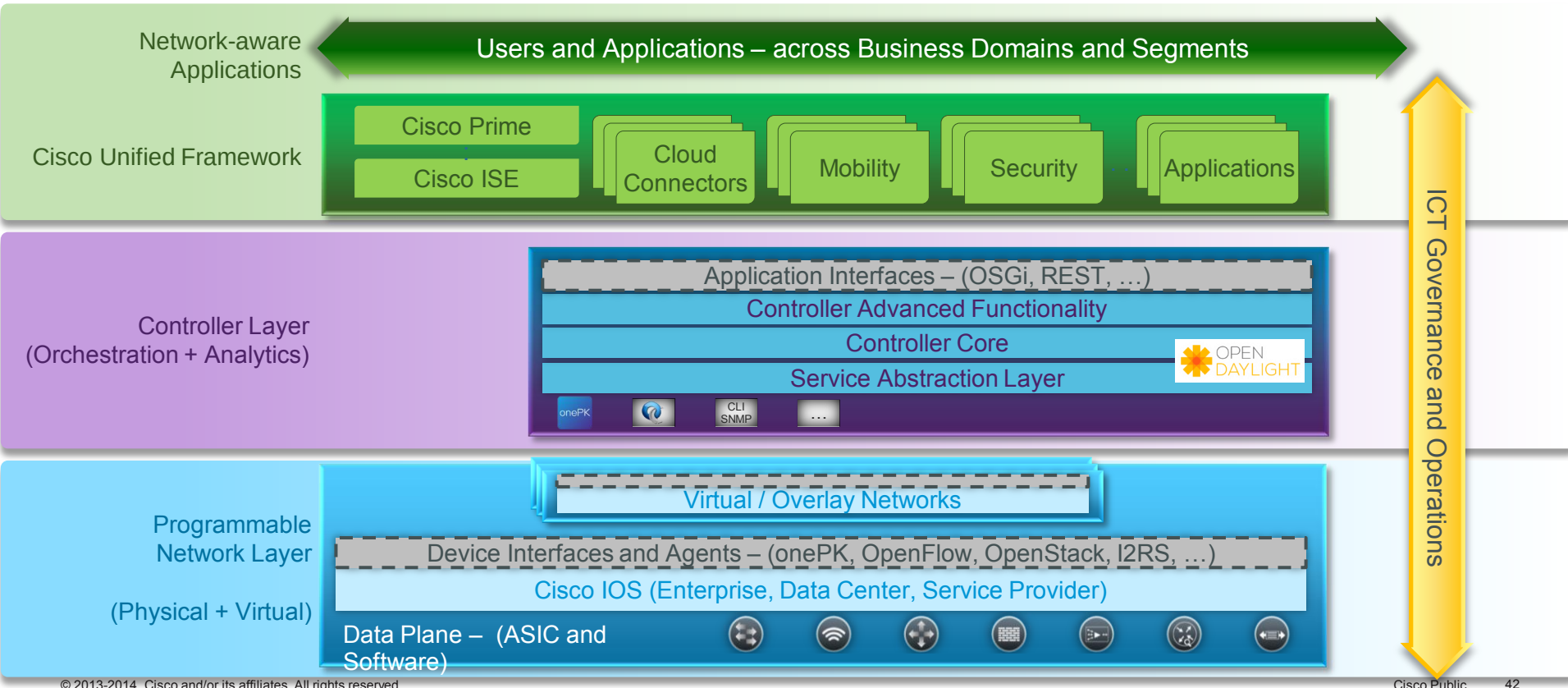
| Node        | In Port | DL Src | DL Dst | DL Type | DL Vlan | NW Src | NW Dst | NW Proto | TP Src | TP Dst | Actions      | Byte Count | Packet Count | Durat |
|-------------|---------|--------|--------|---------|---------|--------|--------|----------|--------|--------|--------------|------------|--------------|-------|
| MM-Switch-2 | OF3     | *      | *      | IPv4    | *       | *      | *      | TCP      | 20     | 80     | OUTPUT = OF2 |            |              |       |

1) Create a Filter for relevant Traffic

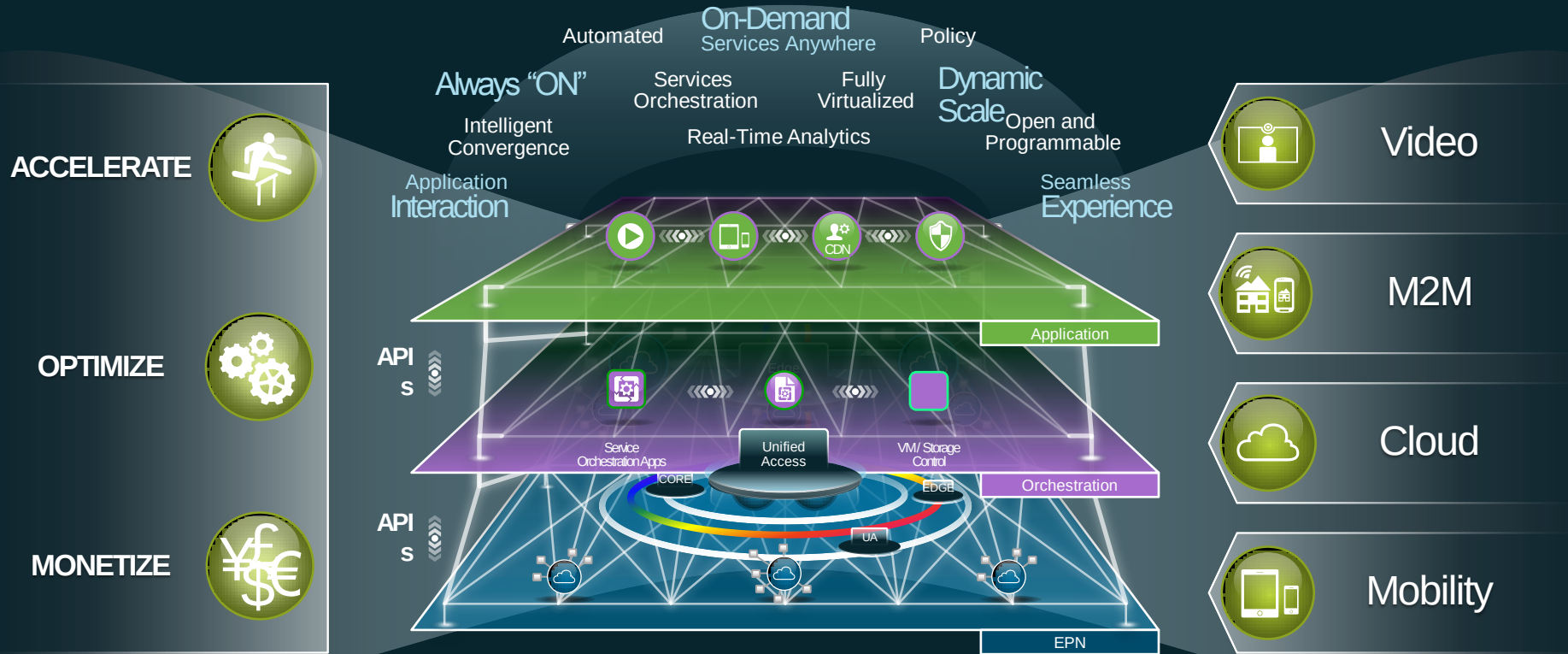
2) Apply Filter to Rule (Source / Destination Ports)

3) Monitor / Troubleshoot actual Traffic Flows as needed

# Cisco ONE – Open Network Environment



# SP Domain : Evolved Programmable Network Architecture



# Agenda

- Welcome
- Industry Observations
- **Network Programmability**

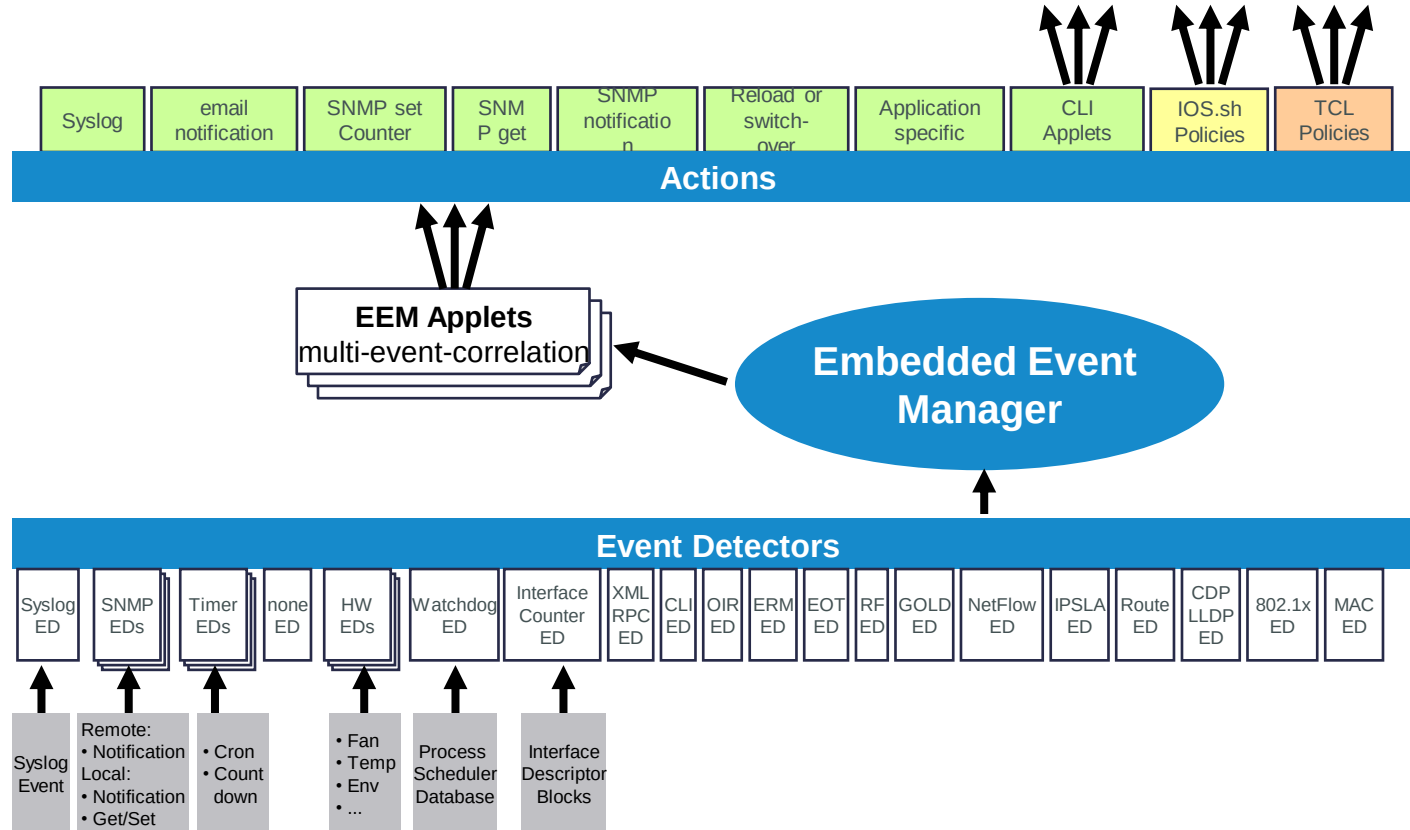
Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- Autonomic Networking
- Resources and References

# Embedded Event Manager (EEM)





# Example: EEM Applets – Loops, Variables

**Problem:** None in Particular

**Solution:** Have fun exploring EEM Applet capabilities

```
event manager applet 99-bob
  description written by bklauser inspired by http://www.99-bottles-of-beer.net
  event none
  action 100 set b 99
  action 110 while $b gt 1
  action 120 puts "$b bottles of beer on the wall, $b bottles of beer."
  action 130 decrement b
  action 140 puts "Take one down, pass it around, "
  action 150 puts "$b bottles of beer on the wall.\n"
  action 160 end
  action 170 puts "$b bottle of beer on the wall, $b bottle of beer."
  action 180 puts "Take one down, pass it around, "
  action 190 puts "no more bottles of beer on the wall.\n"
  action 200 puts "No more bottles of beer on the wall, "
  action 210 puts "no more bottles of beer."
  action 220 puts "Go to the store and buy some more, "
  action 230 puts "99 bottles of beer on the wall.\n"
!
alias exec sing event manager run 99-bob
```

Setting a Variable

While Loop – {

Decrementing a Variable

While Loop – }

Referencing a Variable

Using an Alias to run our Applet

**See also:** <http://www.99-bottles-of-beer.net/language-cisco-ios-embedded-event-manager-applet-2909.html>

© 2013-2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public

46

# Packaging Network Automations

**Problem:** Cisco IOS Embedded Automation Systems often include multiple configuration items, files, checks and procedures – how to ensure they are deployed consistently?

**Solution:** Cisco EASy provides a simple packaging mechanism and open-source EASy Installer. A developer guide is available online to assist with the creation of EASy packages.

- Package Description
- Pre-Requisite Verification
- Pre-Installation Config
- Pre-Installation Exec
- Environment Variables
- Configuration
- Files
- Post-Requisite Verification
- Post-Installation Config
- Post-Installation Exec
- Uninstall

EASy Installer

=

Menu Guided Installation

+

MyPackage.tar



```
Router# easy-installer tftp://10.1.1.1/mypackage.tar flash:/easy
-----
Configure and Install EASy Package 'mypackage-1.03'
-----
1. Display Package Description
2. Configure Package Parameters
3. Deploy Package Policies
4. Exit

Enter option: 2
```

See: <http://www.cisco.com/go/easy>

EASy Package guide: <http://tools.cisco.com/squish/cEAe3>

© 2013-2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public

47

# Embedded Automation Systems (EASy)

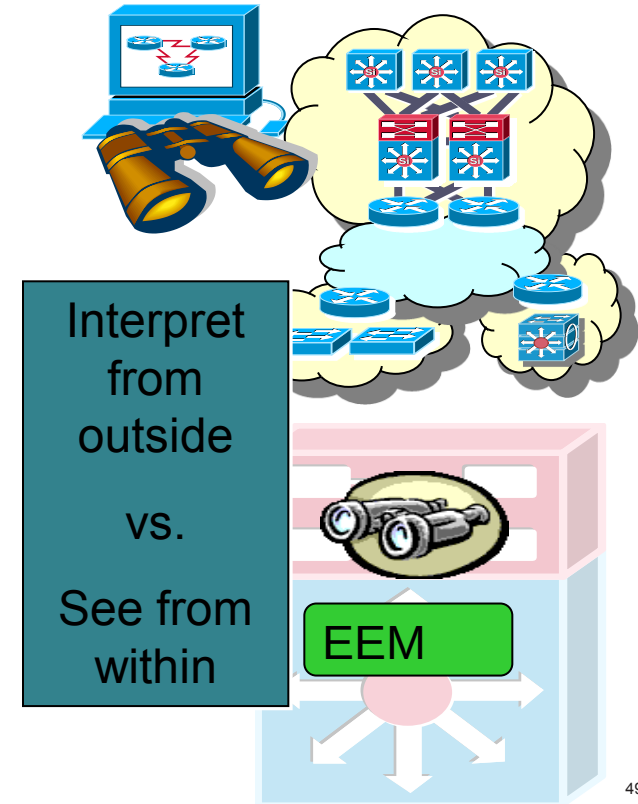


1. Browse and Download EASy Packages  
[www.cisco.com/go/easy](http://www.cisco.com/go/easy)
2. Make Sure to also download EASy Installer
3. Browse Other Embedded Automations  
[www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
4. Learn About The Technology Under The Hood  
[www.cisco.com/go/instrumentation](http://www.cisco.com/go/instrumentation)  
[www.cisco.com/go/eem](http://www.cisco.com/go/eem)  
[www.cisco.com/go/pec](http://www.cisco.com/go/pec)
5. Discuss, Ask Questions, Suggest Answers  
[supportforums.cisco.com](http://supportforums.cisco.com)  
[supportforums.cisco.mobi](http://supportforums.cisco.mobi)
6. Upload your own Examples to CiscoBeyond  
[www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
7. Engage via [ask-easy@cisco.com](mailto:ask-easy@cisco.com)

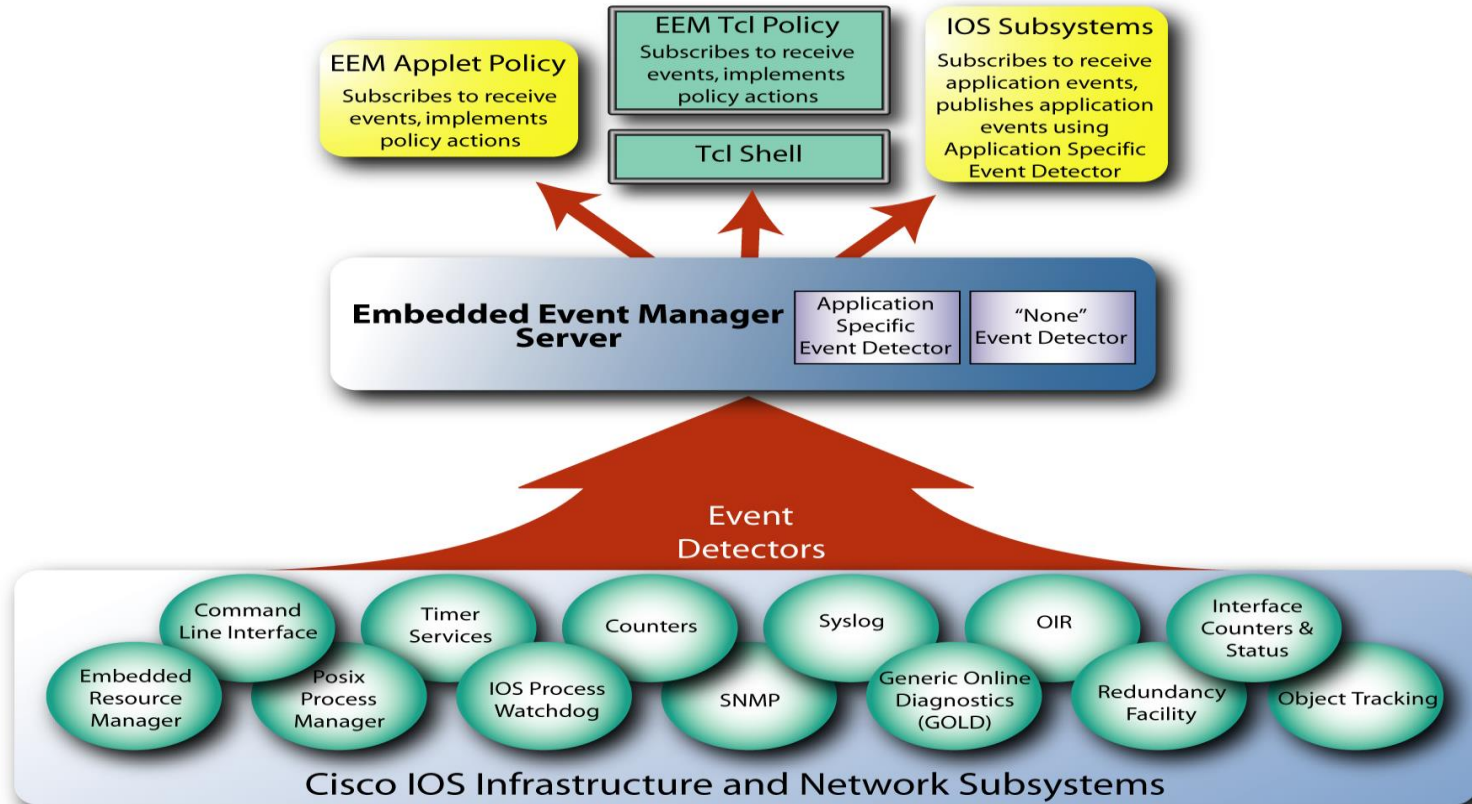


# Getting An Insider's View with EEM

- **RELIABLE** – Works when connectivity to external systems may not be.
- **QUICK** – Onboard logic provides instant reaction when certain conditions are detected. Wins precious time to capture critical information.
- **DETAILED** -- An insider's view allow you get more granular information than available externally.
- **EVENT-DRIVEN** – EEM supports many event detectors integrated with IOS modules to generate events and avoid polling.
- **DISTRIBUTED** – Scripts are distributed to each network devices and run locally.

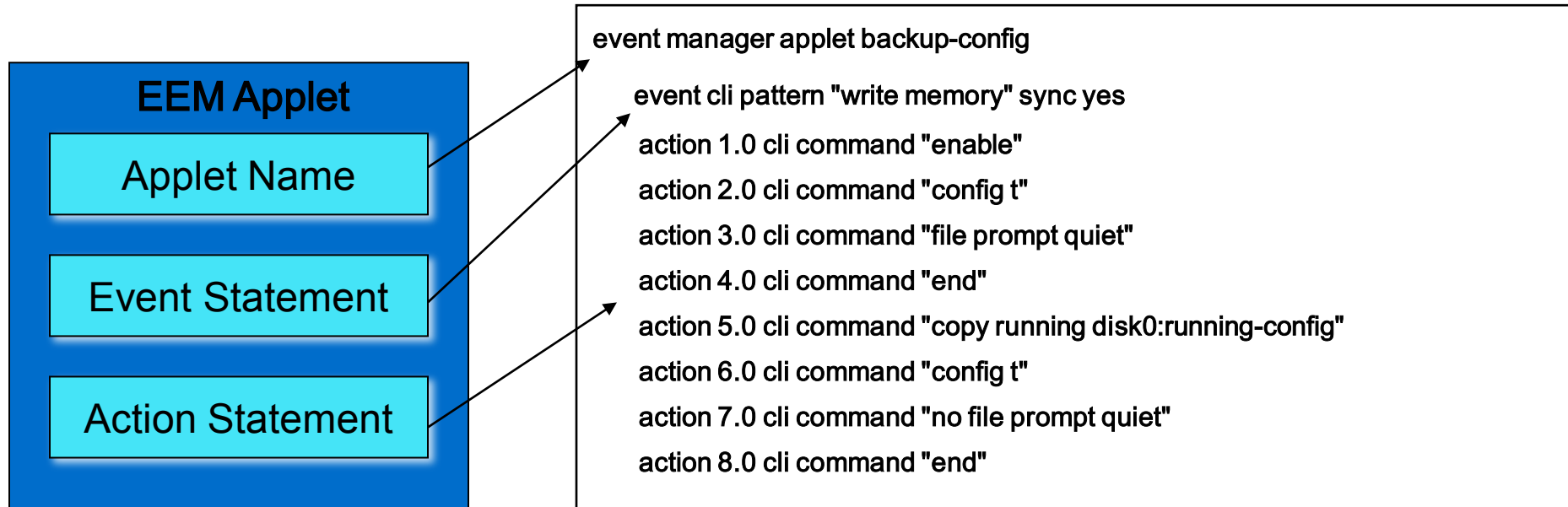


# EEM Basic Architecture



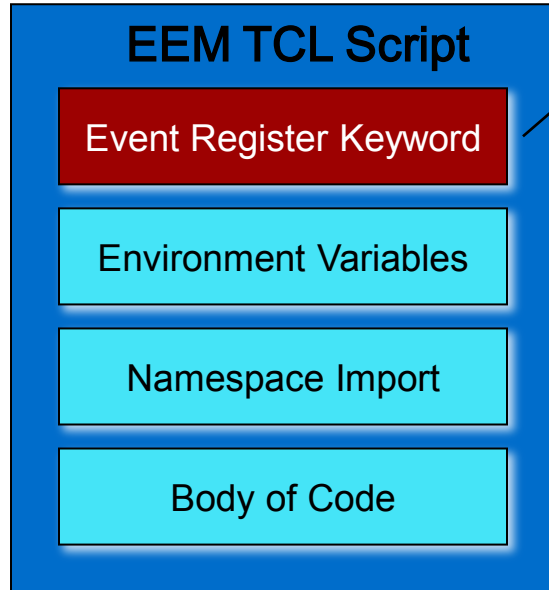
# EEM Applet Overview

An applet is defined on CLI - once entered it becomes part of the configuration



# EEM – Writing TCL Scripts

The Event Register Keyword defines which event this script will be using, and the parameters that define specifics of the event that is to be monitored...



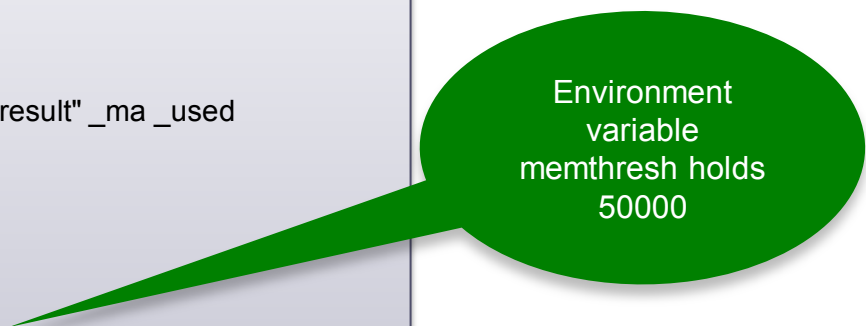
```
::cisco::eem::event_register_syslog pattern "%LINK-3-UPDOWN" maxrun 60  
  
namespace import ::cisco::eem::*  
namespace import ::cisco::lib::*  
  
array set arr_einfo [event_reqinfo]  
if {$_cerno != 0} {  
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \  
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]  
    error $result  
}  
  
set msg $arr_einfo(msg)  
  
set msg_type "Null"  
set state "Null"  
  
regexp {[^ ]*changed state to up} $msg match state  
  
<snip>
```

# Managed Network Use Case – Check for Memory Spikes

**Problem:** You want to be notified when memory spikes over a given

**Solution:** Use EEM, and save context between runs to check for memory spikes.

```
event manager applet memtest
event timer watchdog time 30
action 100 set _last_saved "0"
action 102 cli command "enable"
action 103 cli command "show mem stat | in Processor"
action 104 regexp "Processor\s+[0-9A-F]*\s+[0-9]*\s+([0-9]*).*" "$_cli_result" _ma _used
action 105 handle-error type ignore
action 106 context retrieve key savekey
action 107 handle-error type exit
action 108 if $_used gt "$_last_saved"
action 109 subtract $_used $_last_saved
action 110 if $_result gt "$memthresh"
action 111 syslog msg "WARNING: Memory jumped more than $memthresh bytes: $_result"
action 112 end
action 113 end
action 114 set _last_saved "$_used"
action 115 context save key savekey variable "_last_saved"
```



Environment  
variable  
memthresh holds  
50000

# Managed Network Use Case – Monitor Memory Usage

**Problem:** Monitoring memory spikes are good, but we also want to monitor total processor memory.

**Solution:** Use the integration of EEM + ERM to trigger an EEM event when processor memory is greater than 80%

```
event manager applet memtest
resource policy
policy critmem global
system
memory processor
critical rising 80 interval 5
user global critmem
```

```
event manager applet totmemcheck
event resource policy critmem
action 100 mail server "<server>" to "<to>" from "<from>" subject "Total memory too high"
```

# Dynamic SLAs and Custom High Availability

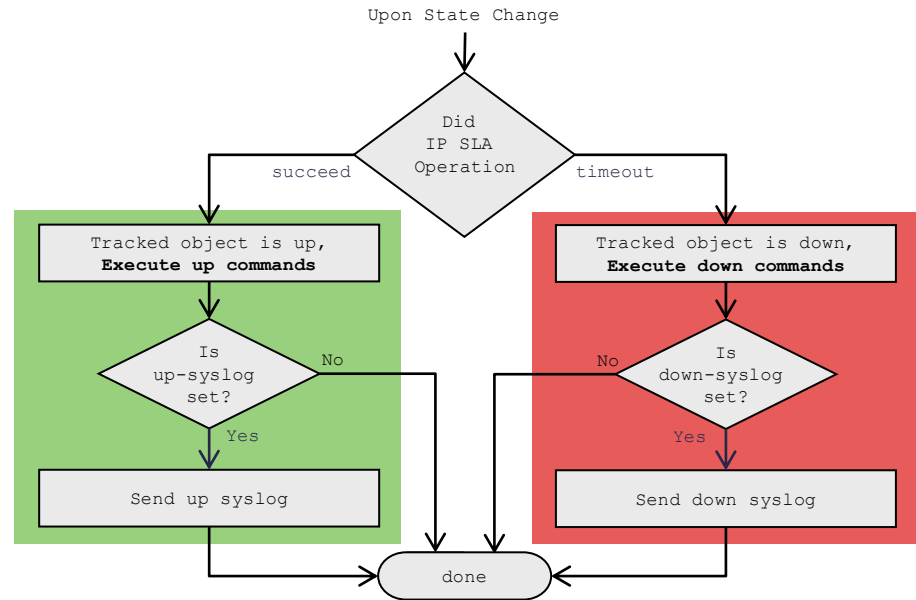
**Problem:** Define – Monitor – Alert was yesterday. Today's SLAs often require preventive, mitigating or optimizing actions to happen automatically.

**Solution I:** Use configurable point features where available

**Solution II:** Use EEM with a generic Event Detector

**Solution III:** Use EEM with a specific Event Detector

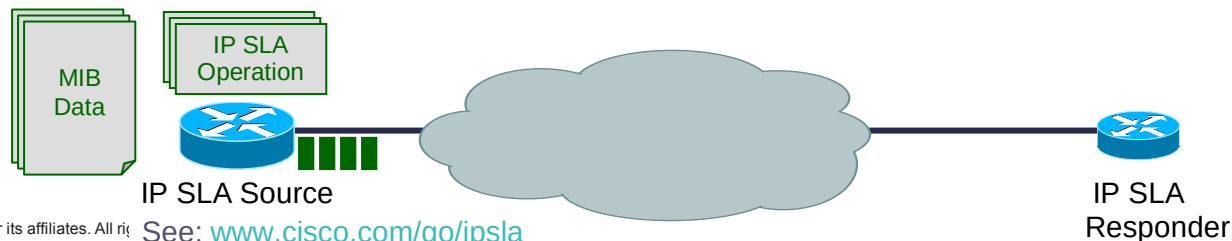
**Solution IV:** Use onePK to program for external dynamic metrics and/or algorithms



# Recap – Cisco IP SLA

- Active probing by injecting synthetic test traffic
- Experience and Adoption across markets and technology domains
- Vast range of Cisco and 3rd Party NMS tool support

| Metrics    | Latency       |             | Jitter       | Packet Loss  |                |           | Connectivity |          |          |      |     |
|------------|---------------|-------------|--------------|--------------|----------------|-----------|--------------|----------|----------|------|-----|
| Domains    | IP            | Ethernet    |              | MPLS         | VoIP           |           | Services     |          | Medianet |      |     |
| Operations | ICMP Echo     | ICMP Jitter | UDP PathEcho | TCP Connect  | 802.1ag Jitter | LSP Trace | PWE3 VCCV    | H.323 GD | SIP GD   | HTTP | DNS |
|            | ICMP PathEcho | UDP Echo    | UDP Jitter   | 802.1ag Echo | LSP Ping       | LSP Tree  | H.323 CS     | SIP CS   | DHCP     | FTP  |     |



# Dynamic SLAs and Custom High Availability – I

## **Solution I:** Configuring IP SLA reaction triggers:

```
RouterA(config)#  
ip sla 10  
    icmp-echo 3.3.3.3  
    frequency 10  
ip sla reaction-configuration 10 react timeout threshold-type consecutive 3 action-type trapAndTrigger  
ip sla schedule 10 life forever start-time now  
ip sla reaction-trigger 10 20  
logging on  
ip sla logging trap  
snmp-server host nms_server version 2c public  
snmp-server enable traps syslog
```

## Sending SNMP trap with IP SLAs embedded threshold

# Dynamic SLAs and Custom High Availability – II

## **Solution II:** Using IP SLA and generic EEM Event Detector:

### IP SLA

```
ip sla 10
icmp-echo 3.3.3.3
timeout 500
frequency 3
ip sla schedule 10 life forever start-time now
```

### Embedded Object Tracking (EOT)

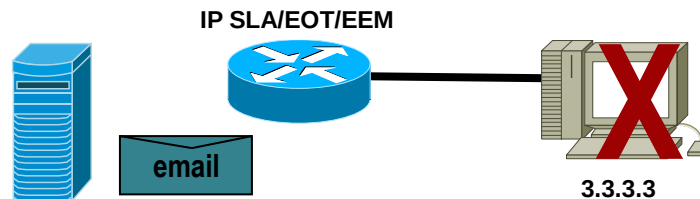
```
track 10 rtr 10 reachability
delay down 10 up 20
```

### Environment Variables

(\$\_\* variables to be defined)

### EEM Applet

```
event manager applet email_server_unreachable
event track 10 state down
action 1.0 syslog msg "Ping has failed, server unreachable!"
action 1.1 cli command "enable"
action 1.2 cli command "del /force flash:server_unreachable"
action 1.3 cli command "show clock | append server_unreachable"
action 1.4 cli command "show ip route | append server_unreachable"
action 1.5 cli command "more flash:server_unreachable"
action 1.6 mail server "$_email_server" to "$_email_to" from "$_email_from" subject "Server Unreachable: ICMP-Echos Failed" body "$_cli_result"
action 1.7 syslog msg "Server unreachable alert has been sent to email server!"
```



# Dynamic SLAs and Custom High Availability – III

## Solution III: Using IP SLA and specific EEM Event Detector:

```
Router(config)# ip sla 10
Router(config-ip-sla)# icmp-echo 3.3.3.3

Router(config)# ip sla enable reaction-alerts

Router(config)# ip sla reaction-config 1 react Timeout
    action-type none threshold-type consecutive 3

Router(config)# ip sla schedule 10 start now

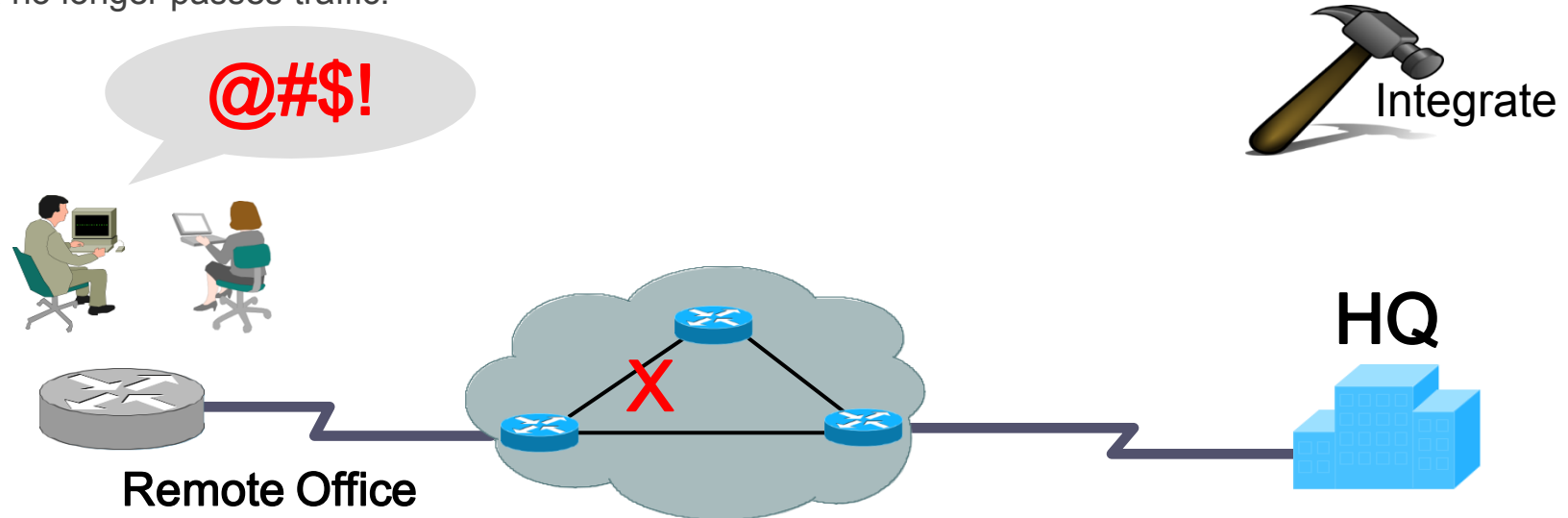
Router(config)# event manager applet my-test
router(config-applet)# event ipsla operation-id 10 reaction-type Timeout
router(config-applet)# action 1.0 syslog priorities emergencies
    msg "IP SLA operation $_ipsla_oper_id to server XYZ has timed out"
```

Trigger an Embedded Event Manager Applet when the IP SLA operation threshold is crossed

# Managed Network Use Case - Staying Connected

**Problem:** You are connected to a network that may go down without taking down the local interface. You need to be able to detect when the network is no longer capable of passing traffic, and fail over to a secondary interface.

**Solution:** Use IP SLA together with object tracking and the Embedded Event Manager to test connectivity to through the network, and fail over to a redundant interface (a tunnel) when the main network no longer passes traffic.



# Staying Connected – IP SLA Detector

Ping an IP address that should be reachable at headquarters using IP SLA. Track the reachability of the IP SLA detector.

```
ip sla 1
icmp-echo 10.1.1.1 source-interface Serial0/0
ip sla schedule 1 life forever start-time now
track 1 ip sla 1 reachability
delay up 60
```

When the IP SLA collector goes down, bring up the backup interface.

```
event manager applet wan_down
event track 1 state down
action 1.0 syslog msg "WAN network is no longer passing traffic"
action 2.0 cli command "enable"
action 3.0 cli command "config t"
action 4.0 cli command "int tun0"
action 5.0 cli command "no shut"
action 6.0 cli command "end"
action 7.0 syslog msg "Failed over to tunnel interface"
```

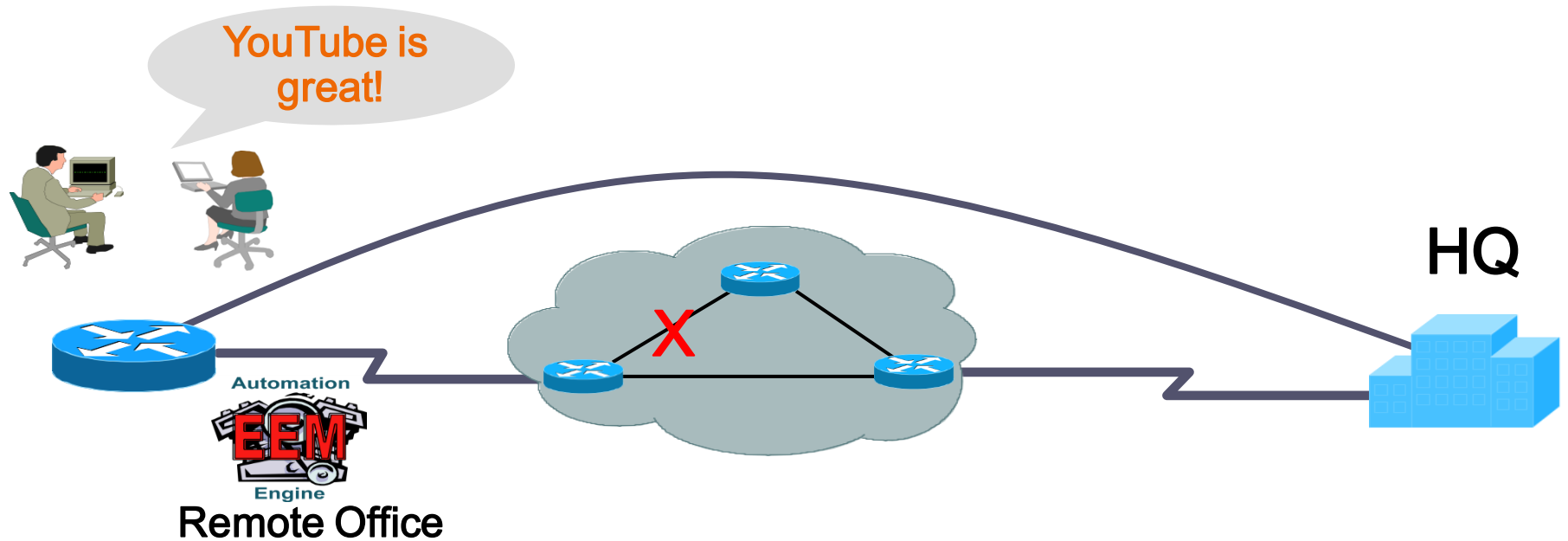
**Requires EEM 2.2 or Higher**

# Staying Connected – EEM Track Applet (Cont.)

When the host is reachable again, move back to using the main interface

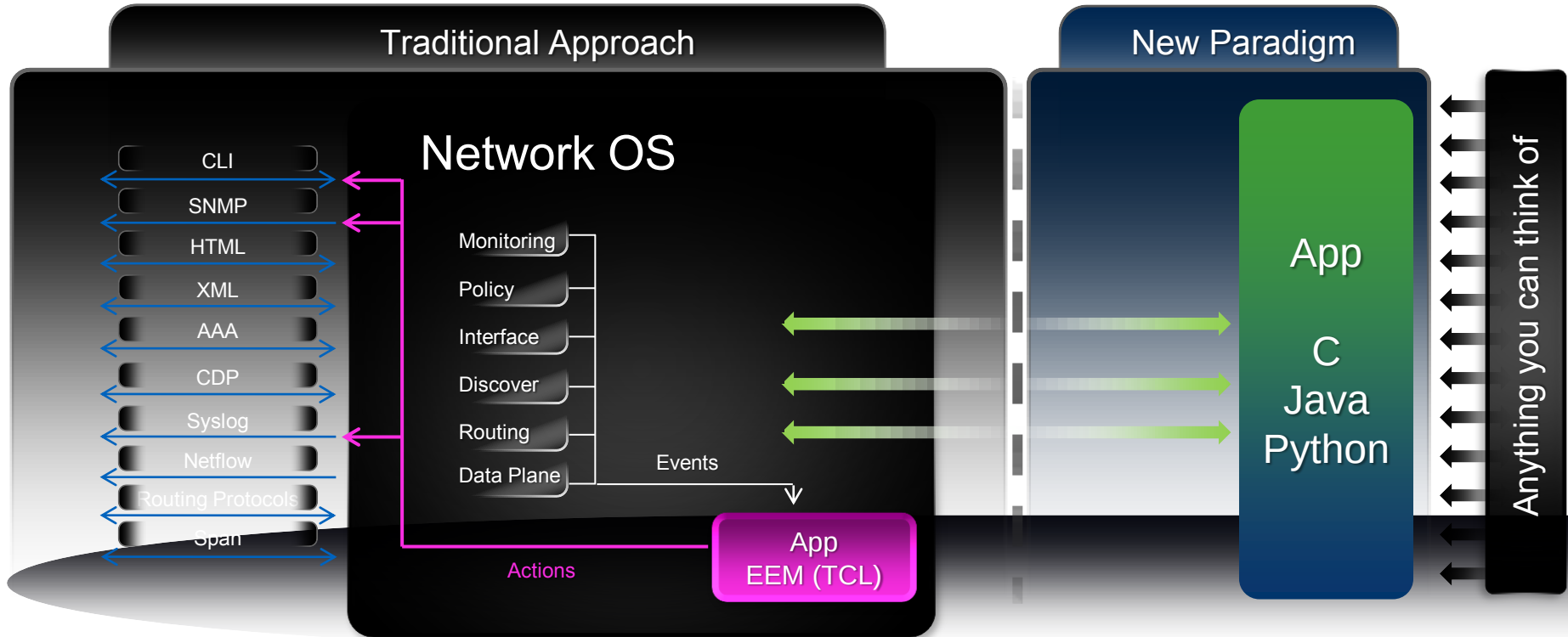
```
event manager applet wan_up
event track 1 state up
action 1.0 syslog msg "WAN network has been restored"
action 2.0 cli command "enable"
action 3.0 cli command "config t"
action 4.0 cli command "int tun0"
action 5.0 cli command "shut"
action 6.0 cli command "end"
action 7.0 syslog msg "Returned traffic to the main interface"
```

# Staying Connected

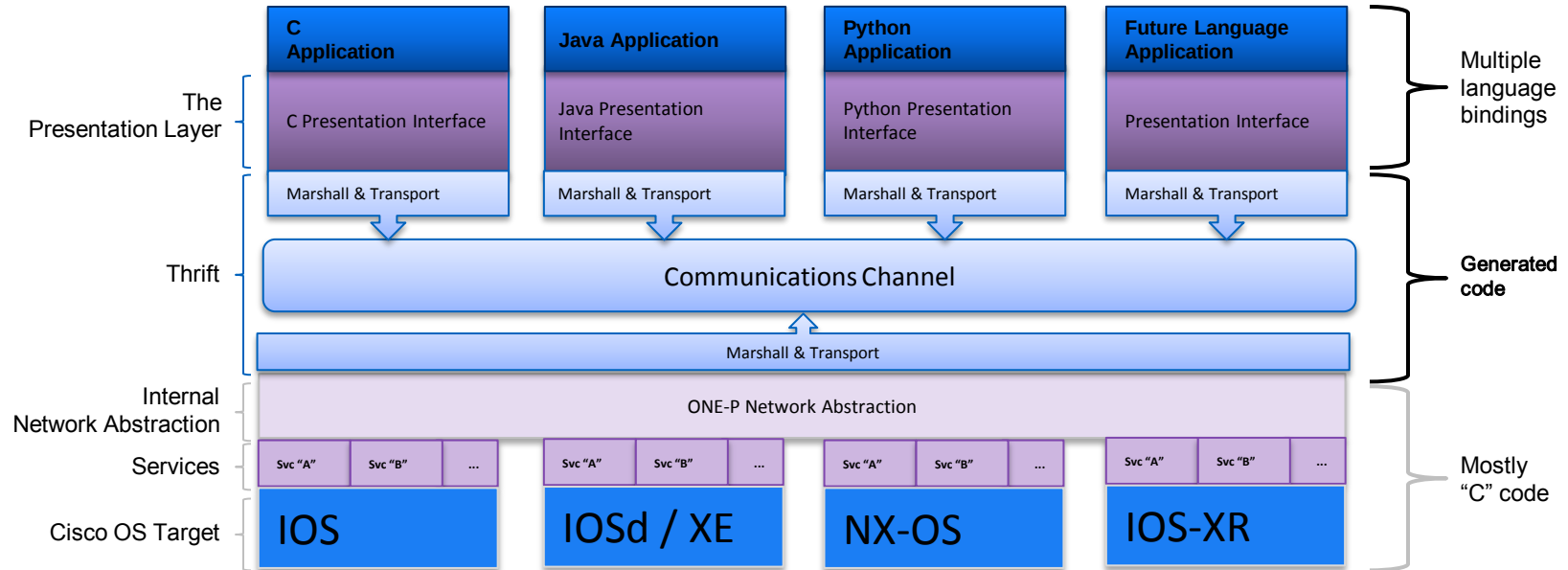


# Enhancing the Experience with Programmatic APIs

# Evolving How We Interact With The Network OS



# The End-To-End View



NOTE: Initially only "C" "Java" "Python" versions of the "Presentation API" will be provided. "C++", and possibly other (e.g., Perl) bindings will likely follow (at relatively low cost).

# APIs are Grouped in Service Sets

| Service Set | Description                                                                                                                                                                        |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Path   | Provides packet delivery service to application: Copy, Punt, Inject                                                                                                                |
| Policy      | Provides filtering (NBAR, ACL), classification (Class-maps, Policy-maps), actions (Marking, Policing, Queuing, Copy, Punt) and applying policies to interfaces on network elements |
| Routing     | Read RIB routes, add/remove routes, receive RIB notifications                                                                                                                      |
| Element     | Get element properties, CPU/memory statistics, network interfaces, element and interface events                                                                                    |
| Discovery   | L2 topology and local service discovery                                                                                                                                            |
| Utility     | Syslog events notification, Path tracing capabilities (ingress/egress and interface stats, next-hop info, etc.)                                                                    |
| Developer   | Debug capability, CLI extension which allows application to extend/integrate application's CLIs with network element                                                               |

# Where Do onePK Applications Run?

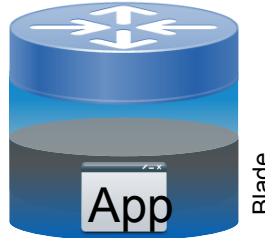
Choose the Hosting Model that Suits Your Platform and Your Application



## On An External Server

- Plentiful memory/compute
- Higher latency and delay
- Supported on by all platforms

“End-Node”



## On A Hardware Blade

- Dedicated memory/compute
- Low latency and delay
- Requires modular hardware blade

“Blade”



## On the Router

- Shared memory/compute
- Very low latency and delay
- Requires modular software architecture

“Process”

# onePK Application Models

# Feedback Loop Applications

## Integration of application and the network

### Application domain tasks

Gather, Analyze, Receive Requests

Makes a decision, pushes back to Network Element

### Network domain tasks

Act, Observe, Notify

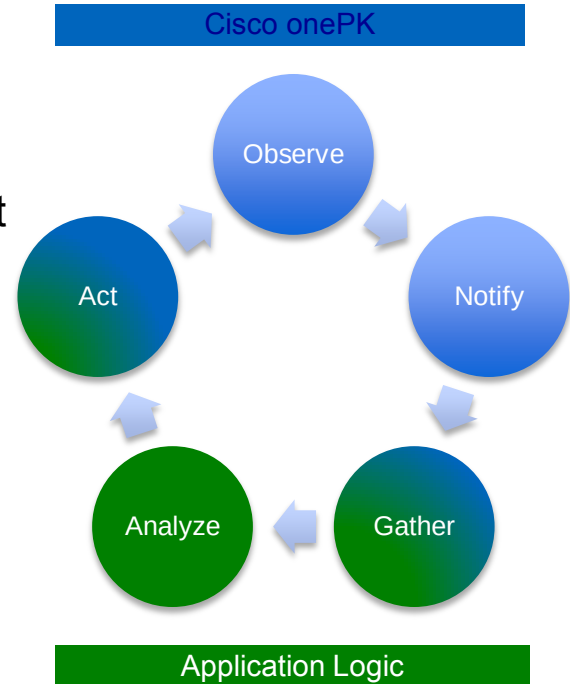
Application can delegate rules to network to enable the network to take local decisions

### Examples

Auto fix on MTU mismatch

Backup interface manipulation

Dynamically apply policy as needed.



# Agent Model Applications

Agent application resides on NE, utilizes onePK API library

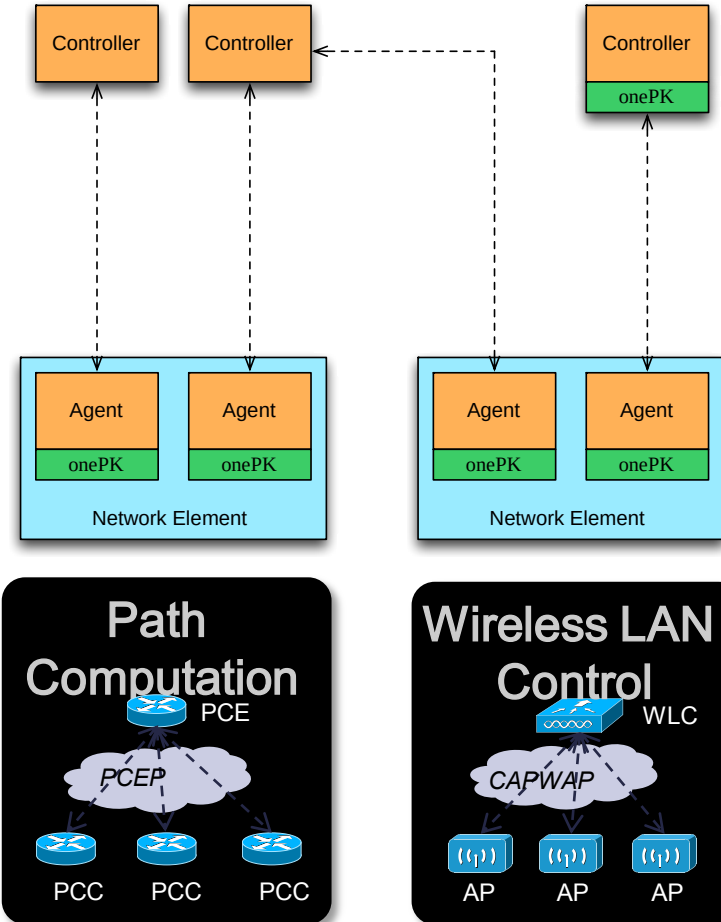
Choice of communication methods between agent and controller

Choice of where bulk of processing will occur.

Controller typically has network wide view, agent has individual box view.

## Examples

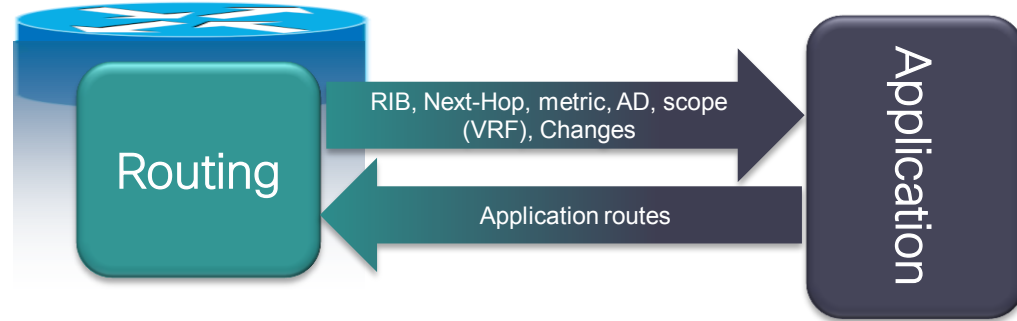
Web application with REST interface  
Management over XMPP



# Deep Dive into Core Features

## Routing

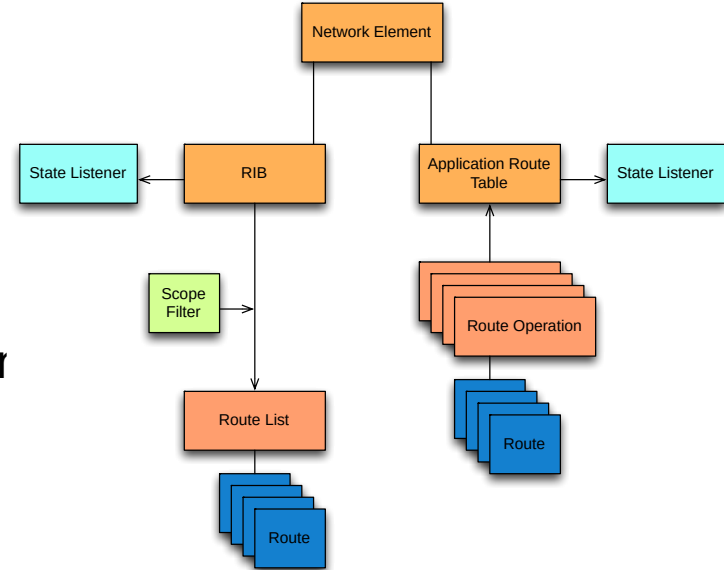
# Routing Services



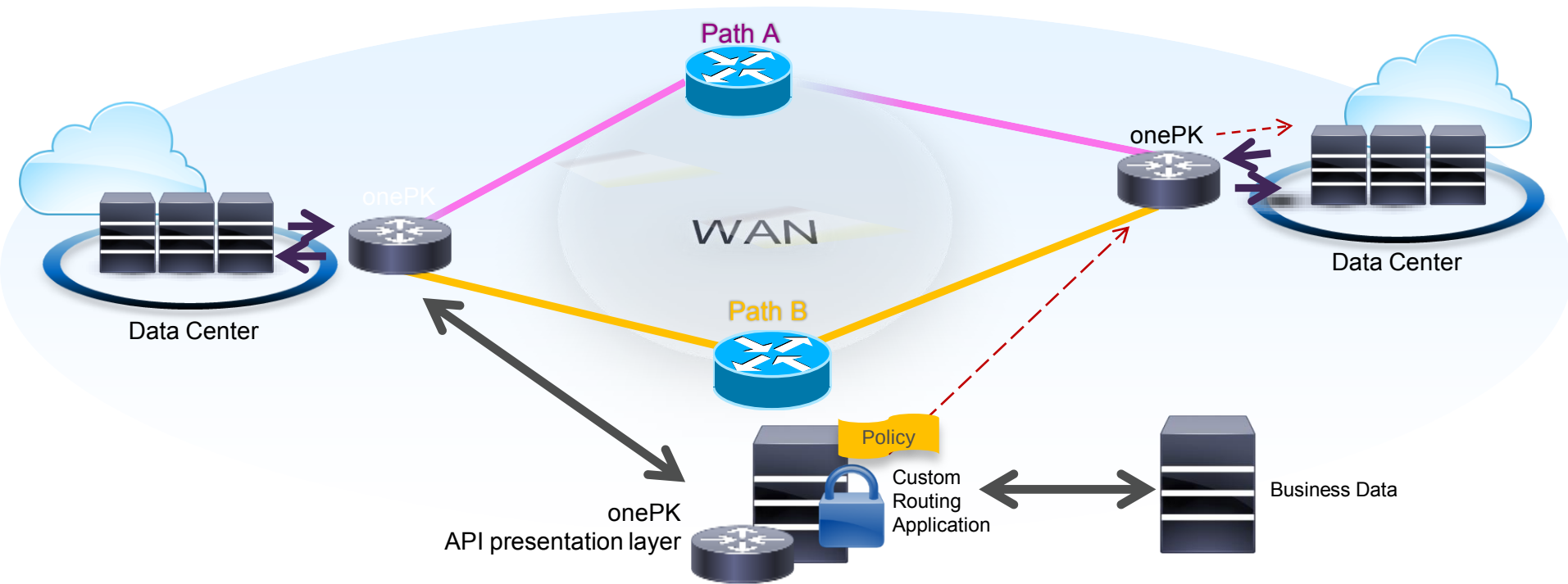
Get Information from the RIB.

Set Routes into the Application Routing Table,  
with settable Administrative Distances.

Applications can receive notifications when routes change, both in the RIB, and in the Application Routing Table.



# Example: Routing for Dollars / CO2 / Tulips /...

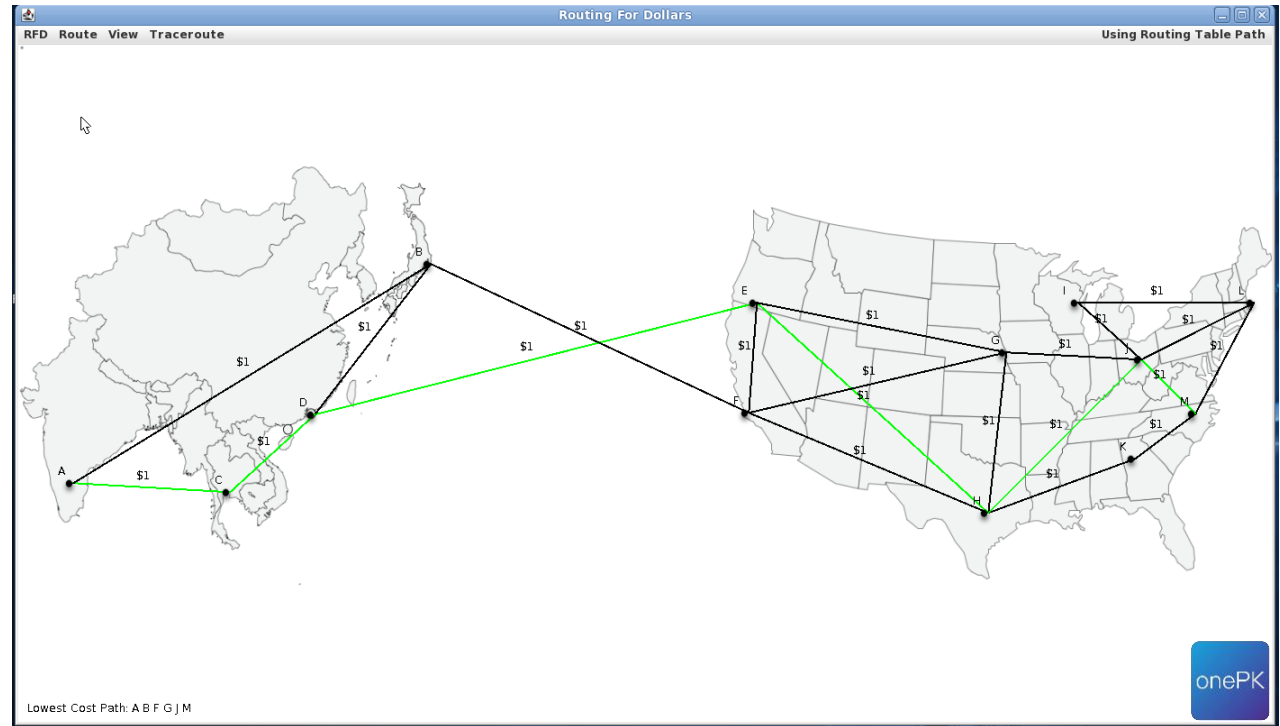


Network Extrinsic Metrics Influencing the Routing Topology

# Example: Routing for Dollars / CO2 / Tulips /...

## Setup

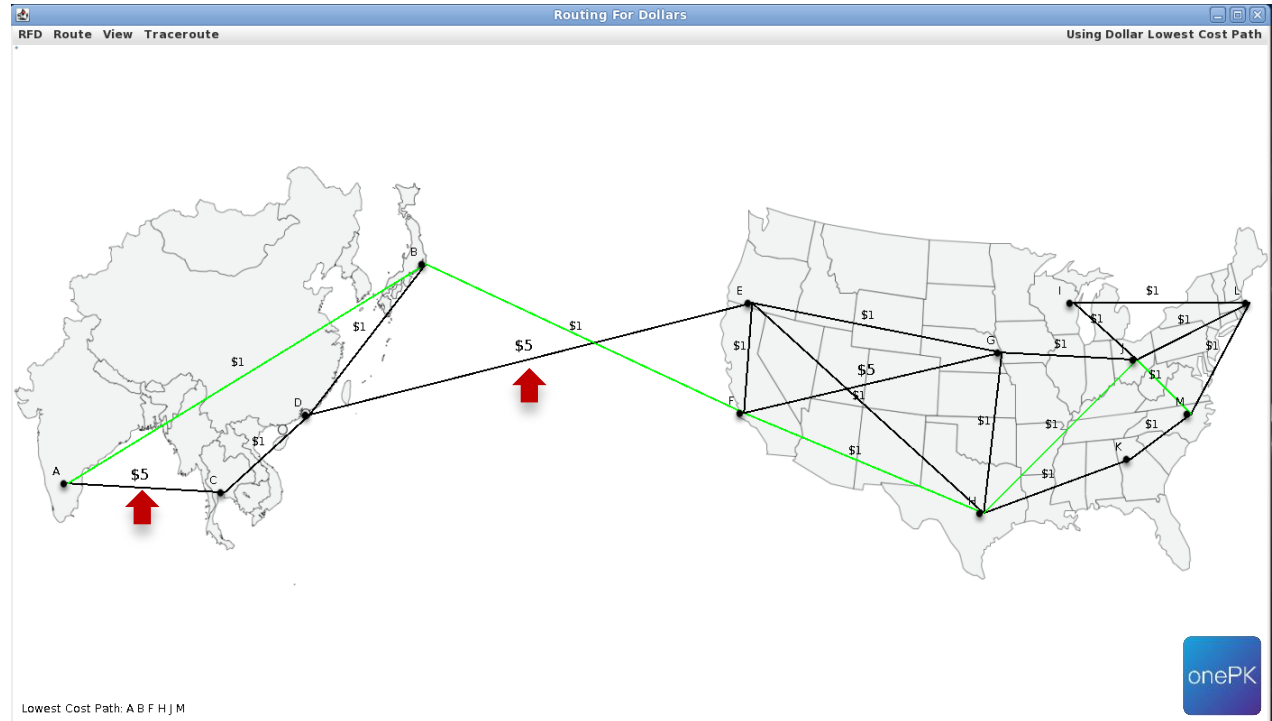
- EIGRP
- Routing Topology
- No External Metrics
- No External Algorithm



# Example: Routing for Dollars / CO2 / Tulips /...

## Application Routes

- EIGRP
- onePK
- External Metrics
- External Algorithm



# Example: Routing for Dollars / CO2 / Tulips /...

Routing For Dollars

RFD Route View Traceroute

Using Dollar Lowest Cost Path

Lowest Cost Path: A B F H J M

Type escape sequence to abort.  
Tracing the route to 100.1.1.1  
VRF info: (vrf)

Type escape sequence to abort.  
Tracing the route to 100.1.1.1  
VRF info: (vrf in name/id, vrf out name/id)  
1 40.20.1.2 28 msec 8 msec 9 msec  
2 10.60.1.2 17 msec 16 msec 17 msec  
3 20.50.1.2 22 msec 26 msec 22 msec  
4 20.80.1.2 35 msec 35 msec 34 msec  
5 30.30.1.2 139 msec \* 45 msec  
bangalore#

OK

```
bangalore#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
A - application route
+ - replicated route, % - next hop override

Gateway of last resort is not set

10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
C 10.1.1.0/24 is directly connected, Ethernet0/0
L 10.1.1.4/32 is directly connected, Ethernet0/0
D 10.40.1.0/24 [90/2681856] via 40.10.1.2, 2w1d, Serial2/0
D 10.50.1.0/24 [90/3193856] via 40.10.1.2, 2w1d, Serial2/0
D 10.60.1.0/24 [90/3705856] via 40.10.1.2, 2w1d, Serial2/0
D 10.70.1.0/24 [90/3193856] via 40.10.1.2, 2w1d, Serial2/0
D 20.0.0.0/24 is subnetted, 10 subnets
D 20.10.1.0 [90/3705856] via 40.10.1.2, 2w1d, Serial2/0
D 20.20.1.0 [90/4729856] via 40.10.1.2, 2w1d, Serial2/0
D 20.30.1.0 [90/3705856] via 40.10.1.2, 2w1d, Serial2/0
D 20.40.1.0 [90/4217856] via 40.10.1.2, 2w1d, Serial2/0
D 20.50.1.0 [90/4217856] via 40.10.1.2, 2w1d, Serial2/0
D 20.60.1.0 [90/4217856] via 40.10.1.2, 2w1d, Serial2/0
D 20.70.1.0 [90/4729856] via 40.10.1.2, 2w1d, Serial2/0
D 20.80.1.0 [90/4217856] via 40.10.1.2, 2w1d, Serial2/0
D 20.90.1.0 [90/6265856] via 40.10.1.2, 2w1d, Serial2/0
D 20.100.1.0 [90/4729856] via 40.10.1.2, 2w1d, Serial2/0
D 30.0.0.0/24 is subnetted, 5 subnets
D 30.10.1.0 [90/5241856] via 40.10.1.2, 2w1d, Serial2/0
D 30.20.1.0 [90/4729856] via 40.10.1.2, 2w1d, Serial2/0
D 30.30.1.0 [90/4729856] via 40.10.1.2, 2w1d, Serial2/0
D 30.40.1.0 [90/5241856] via 40.10.1.2, 2w1d, Serial2/0
D 30.50.1.0 [90/5241856] via 40.10.1.2, 2w1d, Serial2/0
D 40.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C 40.10.1.0/24 is directly connected, Serial2/0
L 40.10.1.1/32 is directly connected, Serial2/0
C 40.20.1.0/24 is directly connected, Serial2/3
L 40.20.1.1/32 is directly connected, Serial2/3
D 100.0.0.0/24 is subnetted, 1 subnets
D 100.1.1.0 is directly connected, 00:01:56, Serial2/3
bangalore#
```

router ospf 1  
redistribute application <app name> ...

# Example: Routing for Dollars / CO2 / Tulips /...

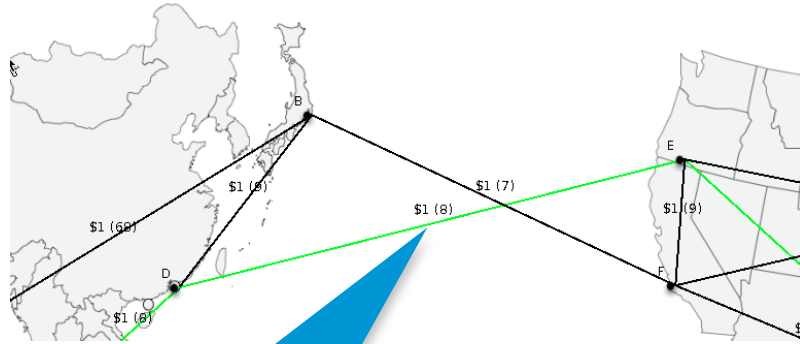
- Code Metrics
  - Total lines of code: 4700 (JAVA)
  - 40% SWING GUI
  - 20% Dijkstra's algorithm, lowest cost path determination
  - 25% Housekeeping: Node and link database
  - 15% Calls to onePK infrastructure + error checking
- Code increase to add “Latency based routing” on top of “Routing for Dollars”
  - 100 lines of code
- Modular code base written in Java has allowed us to port this to mobility client.

Framework makes it easy to modify code and change business logic.

Modular java code makes it easy to deploy on multiple clients.

# Example: Routing for Dollars / CO2 / Tulips /...

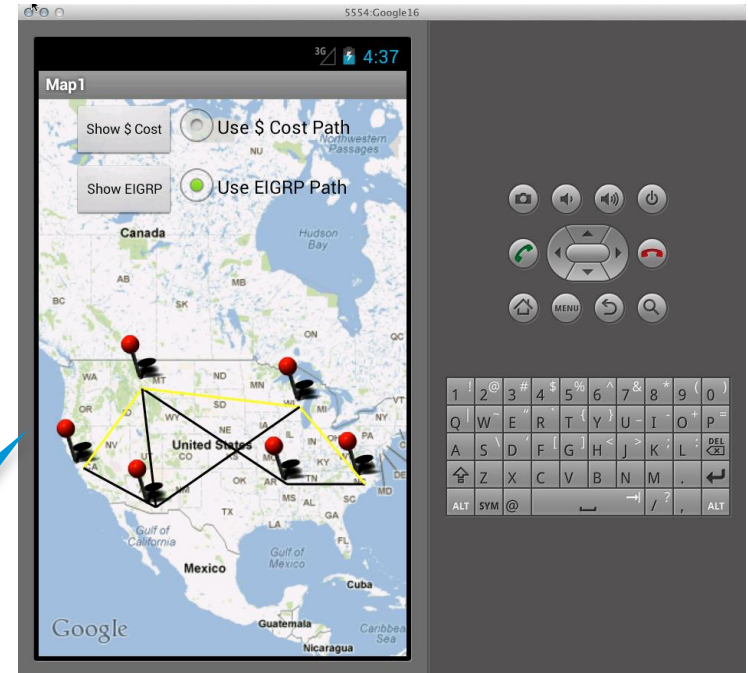
## Recent Extensions



Path determination  
based on lowest  
latency

Latency information  
fed into app through  
IPSLA

Port to  
mobility client



# Custom Routing: Algorithm

- 1) Connect to all Network Elements in network (Static list).
- 2) Get interface information about each Network Element and determine links between them.
- 1) Gather EIGRP information.
- 2) Display nodes, link information, and EIGRP best path in SWING based GUI.
- 3) Use Dijkstra's algorithm to find lowest cost path (All links start with \$1 path).
- 4) As costs are adjusted in GUI recalculate lowest cost path.
- 5) If "Use Lowest Dollar Cost Route" is selected, apply route through Routing Service to each NE.
- 6) If "Show lowest cost path" is selected, display information on SWING updates.
- 7) Remove route on each NE if "Use Routing Table Route" is selected.

# Custom Routing: Code

## Connect to Node, Get Address:

```
B = TopoNode.getNode('B', InetAddress.getByName("40.20.1.2"), 431 + x_adjust, 194 + y_adjust, 150, 330, 260, 320, 3);  
...  
ne = NetworkApplication.getInstance().getNetworkElement(address);  
...  
ne.connect(userName, password);
```

Assign address of NE to connect to.

Get / create NE object.

Connect to NE.

```
// Find all addresses by which the specified NetworkElement is known  
private static List<InetAddress> getAddressList(NetworkElement ne) {  
    HashMap<InetAddress,String> addrHash = new HashMap<InetAddress,String>();  
    try {  
        for (NetworkInterface i : ne.getInterfaceList(new InterfaceFilter())) {  
            List<InetAddress> addrs = i.getAddressList();  
            for (InetAddress a : addrs) {  
                addrHash.put(a, "");  
            }  
        }  
    } catch (OneException e) {  
        e.printStackTrace();  
    }  
    return new ArrayList<InetAddress>(addrHash.keySet());  
}
```

Get Interface List with default filter (all).

Get list of addresses.

# Deep Dive into Core Features Policy

# Policy Service Set

The Policy Service Set provides API's into Cisco's Common Classification Policy Language (C3PL)

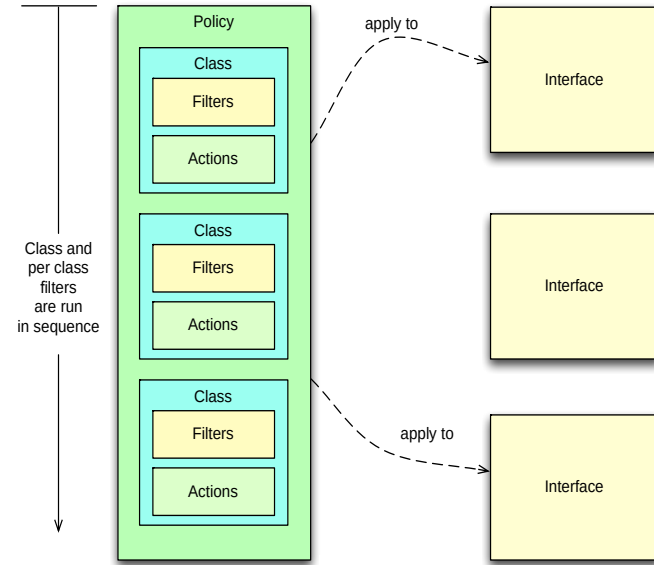
## Objects provided by the Service set:

**Policy object** - This is a top level object in a hierarchy of objects that define class, filter and action objects for applying QoS, Datapath, etc. policy to interface targets. The Policy object is associated with a target (or interface)

**Class object** - Multiple Class objects can be defined within a Policy object. Each Class object can have one or more filter classifier and action objects.

**Filter object** - The Filter object specifies the events to be matched and classifies a packet stream into flows.

**Action object.** The Action object specifies the actions to be performed when an event specified in the Class object is matched.

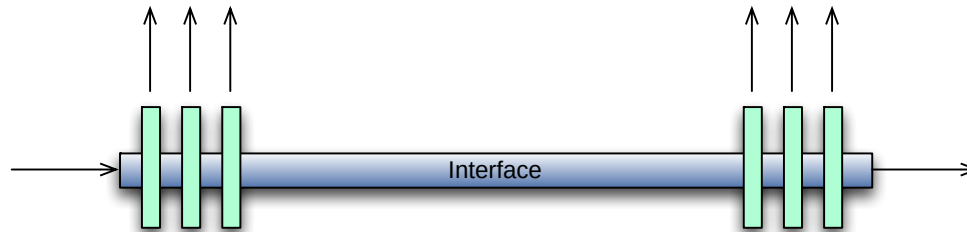


# Policy Objects

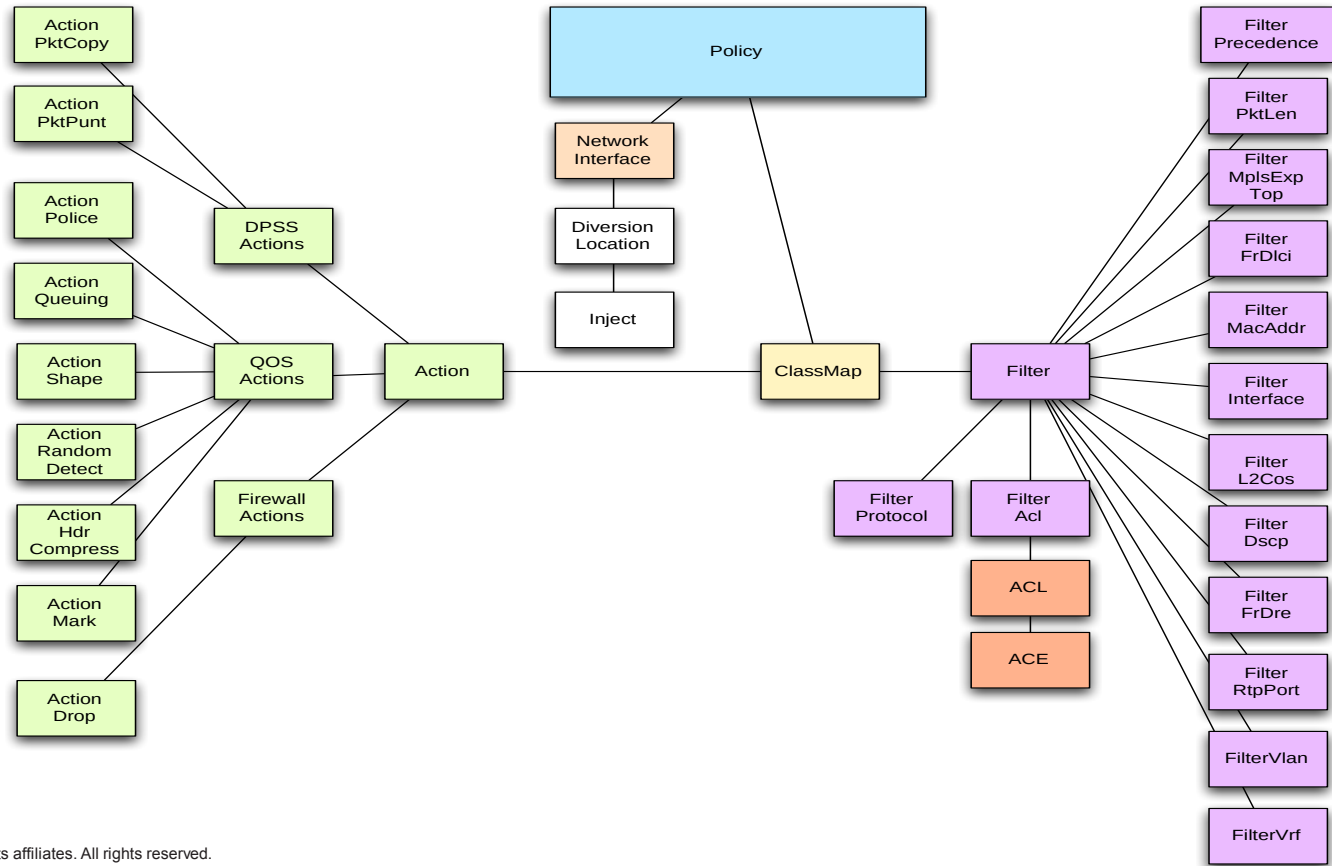
The policy objects can be used to apply QOS, Datapath / Flow, NAT, etc. policy to an interface.

For flow policy, actions can include Punt, Copy, Shape, Next-Hop, Change source, Change destination.

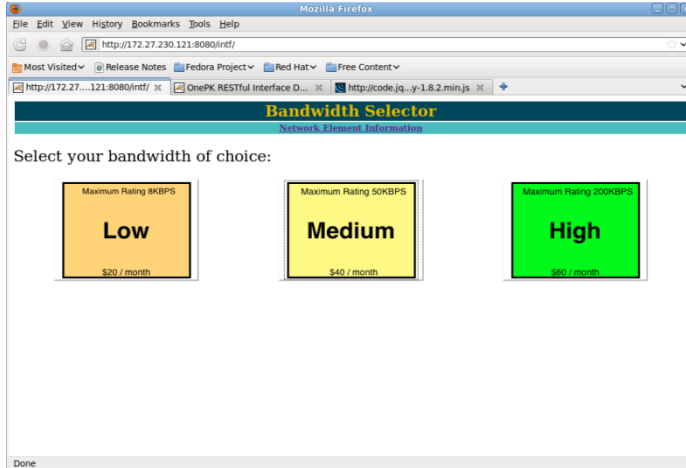
Policy can be applied to both incoming and outgoing traffic.



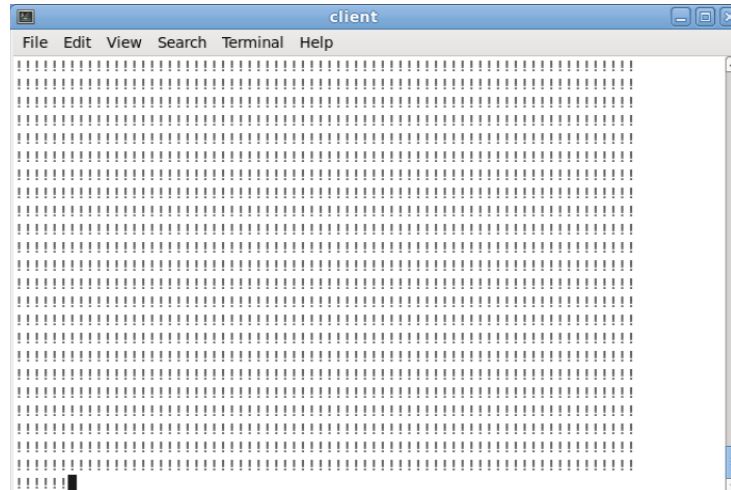
# Policy Object Model



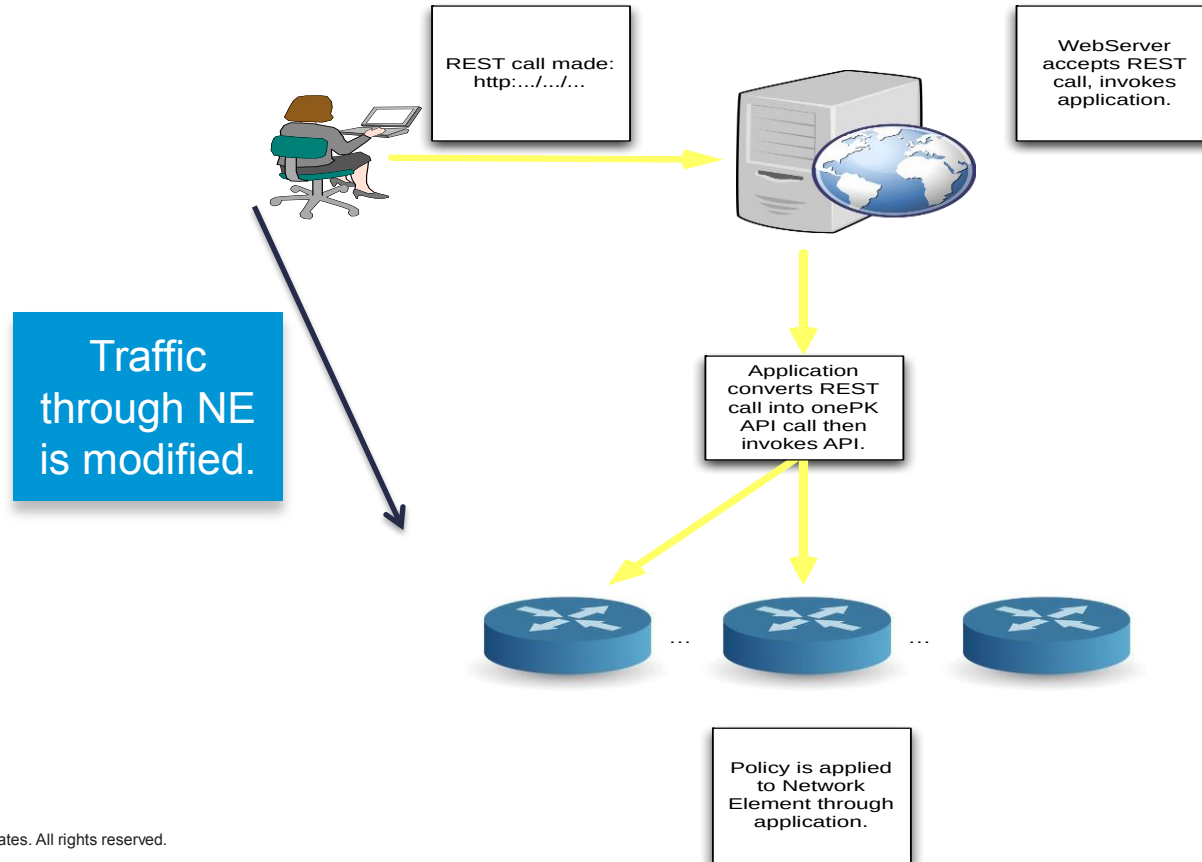
# Policy Demo – Bandwidth Choice through REST



Agent / Controller Model  
+ Web Programming



# Policy Demo – Bandwidth Choice through REST



# Policy Demo: Statistics

- Code Metrics
  - Total lines of code: 648 (Java)
  - HTML: 150 lines
- Uses JAX-RS – Java API for RESTful Web Services.
- Call flow in .java is determined by URL path.



Open design of SDK allows for easy integration with third party tools.

# Policy Demo – Code

```
@GET
@Path("pss/policy/{policy-id}/class/{class-id}/action/shape")
@Produces(MediaType.TEXT_PLAIN)
public String getActionShape(@PathParam("address") String neAddr,
    @PathParam("policy-id") String policyId,
    @PathParam("class-id") String classId,
    @DefaultValue("-1")@QueryParam("rate") int rate,
    @Context UriInfo ui) {
    if (rate < 1) {
        String rtnStr = "Shape parameters: " + ui.getPath() + "?rate=<value => 1\n";
        return rtnStr;
    } else {
        Policy pol = getPolicyMap(policyId, neAddr);

        try {
            pol.addClass(getClassMap(classId, neAddr), 100);
            pol.addAction(getClassMap(classId, neAddr), createActionShape(rate, 1000));
        } catch (Exception e) {
            // ...
        }
    }
}

private Action createActionShape(int rate) {
    return new ActionShape(rate, 1500, 5001, ShapeType.ONEP_SHAPE_AVG,
        ActionShape.RateUnits.ONEP_RATE_UNITS_BPS,
        ActionShape.BurstUnits.ONEP_BURST_UNITS_BITS);
}
```

Define function  
servicing URL.

Parse URL input.

Create new policy.

Add class to  
policy.

Add action to  
class and policy.

Create shape action  
with given rate.

# Deep Dive into Core Features Datapath

# Datapath Service

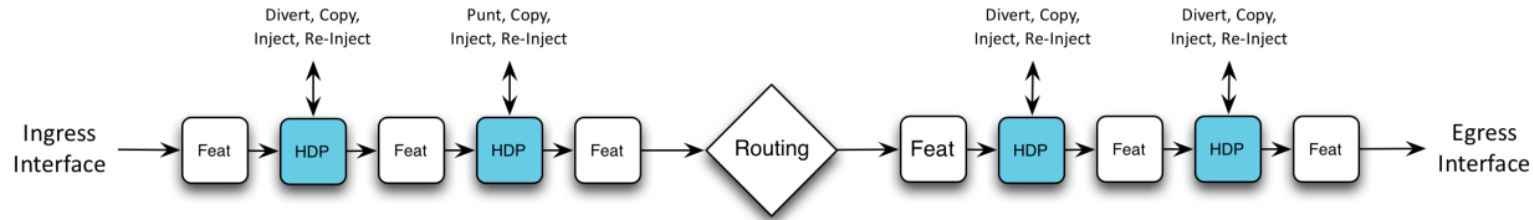
Packet delivery to application.

Packets can be punted to application from various points in forwarding path.

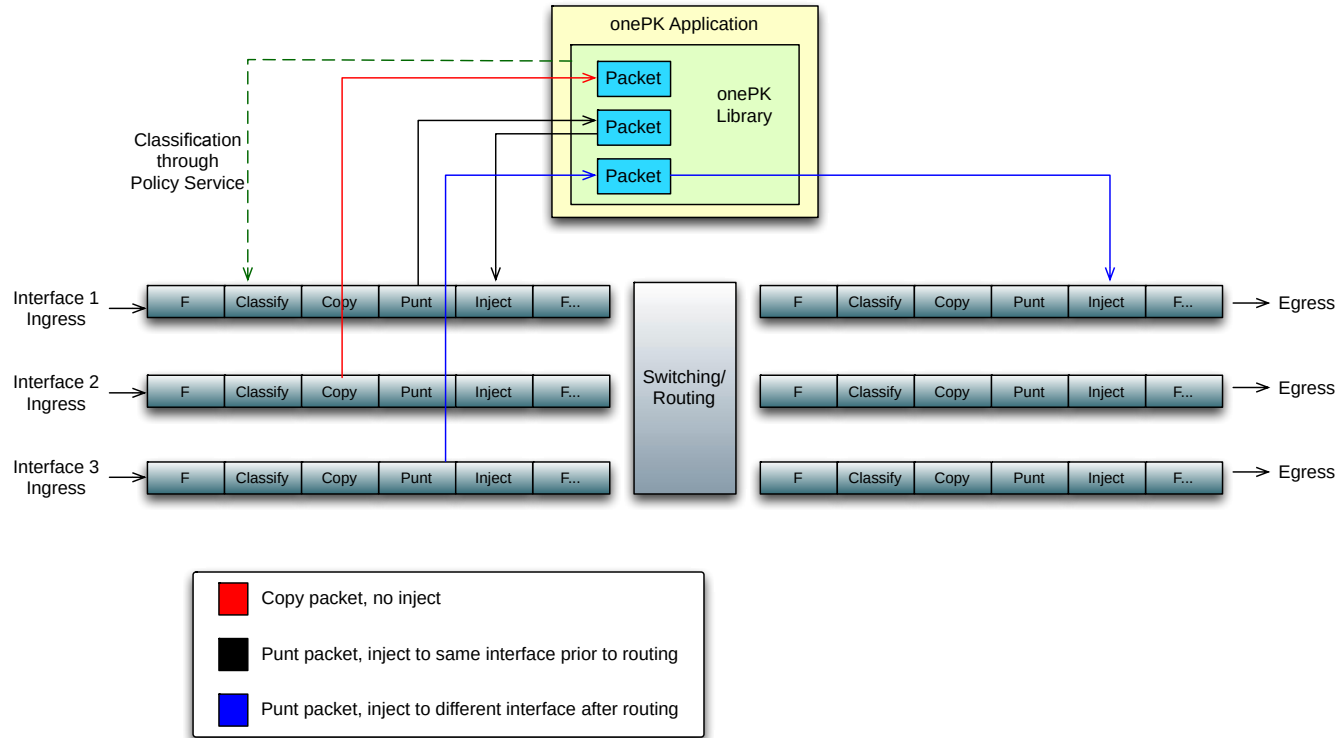
Packets can be copied from various points in forwarding path.

Packets can be injected back into forwarding path.

Works in conjunction with policy to determine what packets to manipulate.

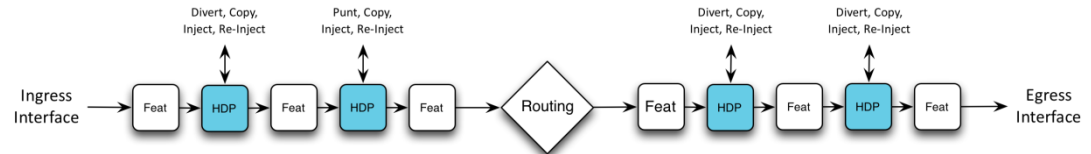


# Datapath Service



# Datapath Service – Sample Applications

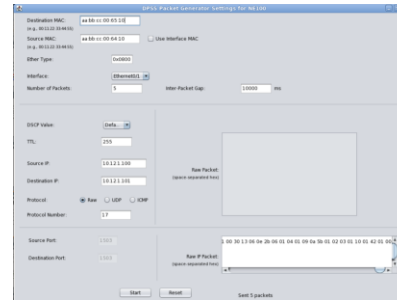
Firewall Applications / content filtering  
Network Security  
Load Balancers  
Packet and flow monitors  
Traffic capture and injection  
Quality of experience troubleshooting



# Example: Packet Generation

**Problem:** Customers want a GUI interface to generate traffic  
**Value proposition:** Use DPSS to inject raw packets.

1. User enters packet type they want injected.
2. App connects to Network Element A.
3. App uses dpss inject routine to inject packet into forwarding path.
4. Packet is received on Network Element B.



1

```
onep_dpss_inject_raw_packet(dpss_handle,  
pktBuf, len, 0, out_intf,  
ONEP_TARGET_LOCATION_HARDWARE_DEFINED_OUTPUT);
```

onePK application

2

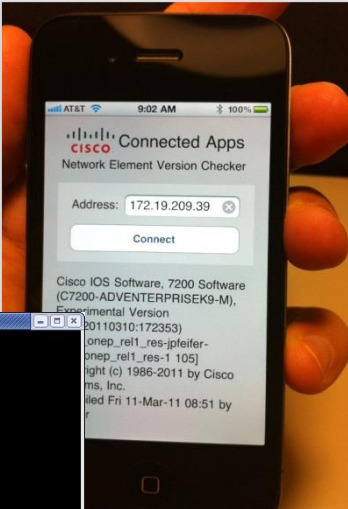
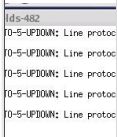


3

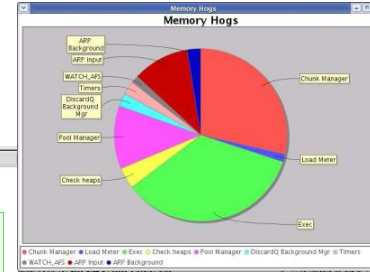


4

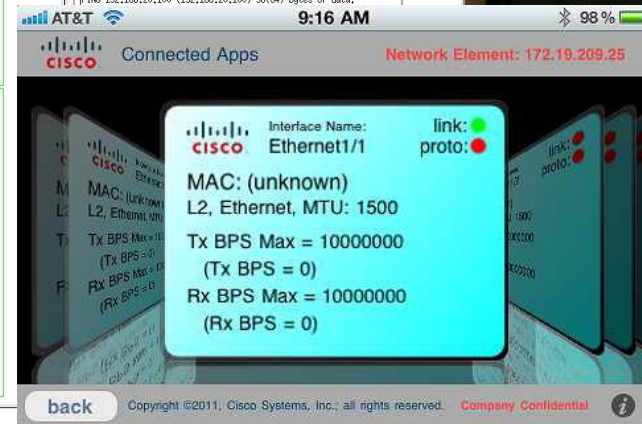
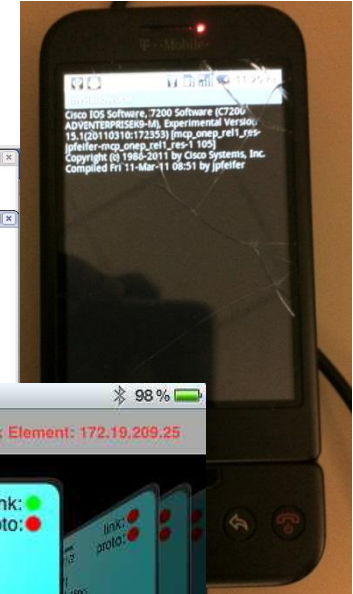




-none-specific  
 -none-specific  
 -none-specific  
 none-specific  
 -none-specific  
 -none-specific



```
root@lisp-ubuntu4:~#  
root@lisp-ubuntu4:~#* ping 192.168.20.100  
PING 192.168.20.100 (192.168.20.100) 56(84) bytes of data,  
^C  
--- 192.168.20.100 ping statistics ---  
13 packets transmitted, 0 received, 100% packet loss, time 12019ms  
root@lisp-ubuntu4:~#  
root@lisp-ubuntu4:~#* ping 192.168.20.100  
PING 192.168.20.100 (192.168.20.100) 56(84) bytes of data,
```



Gray=Not Connected      Green=Happy      Yellow=Warnings      Red=Errors

# Agenda

- Welcome
- Industry Observations
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- Autonomic Networking
- Resources and References



# onePK AiO Virtual Machine

v1.1.0.99

# Prerequisites

- onePK AiO VM Image (.ova file)
- VMWare Player installed on host OS
- ~10 GB disk space available on host OS
- ~4 GB RAM available on host OS



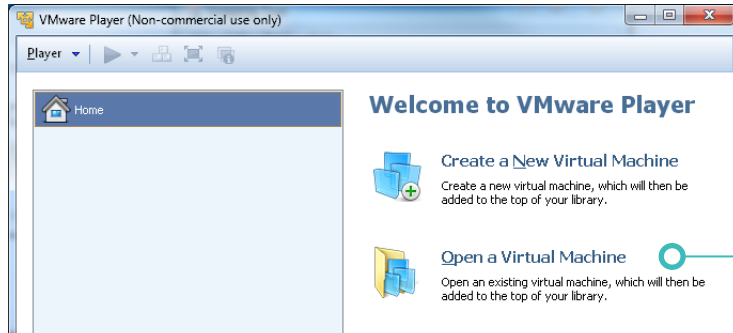
# onePK All in ONE Virtual Machine (onePK AiO VM)

## Preconfigured Development Environment

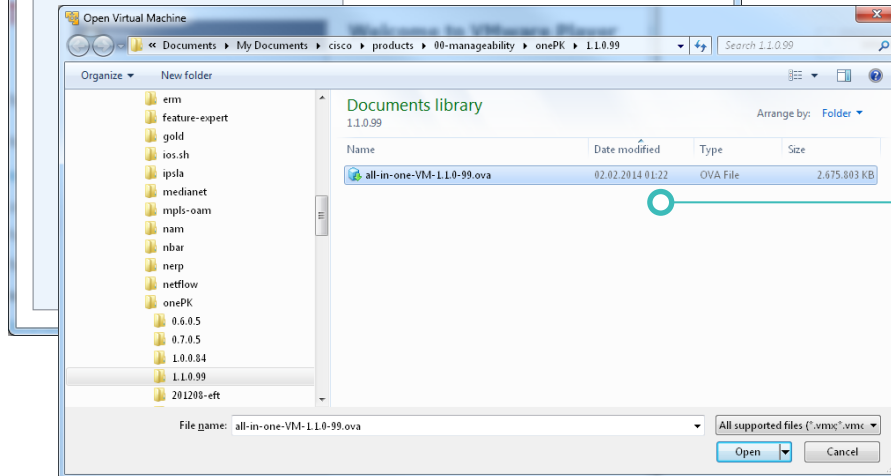
- onePK SDK for C, Java and Python
- Sample and Tutorial Applications
- Network Simulation based on vIOS
- Eclipse IDE, Tools and Documentation
- Developer Community at <https://developer.cisco.com/web/onePK>

Download YOUR VM via Devnet  
<http://tinyurl.com/DevNet-ONE>

# Getting Started with the onePK AiO VM – 1/x

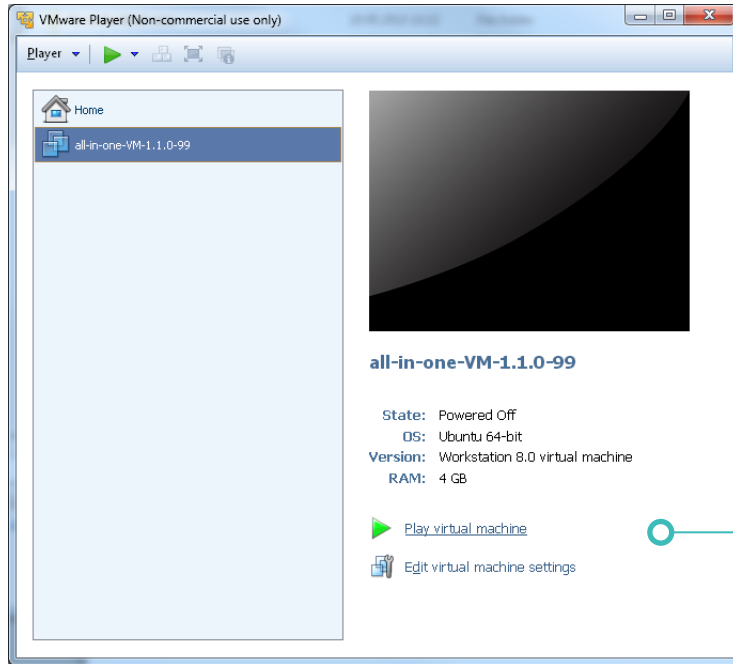


Open a Virtual Machine



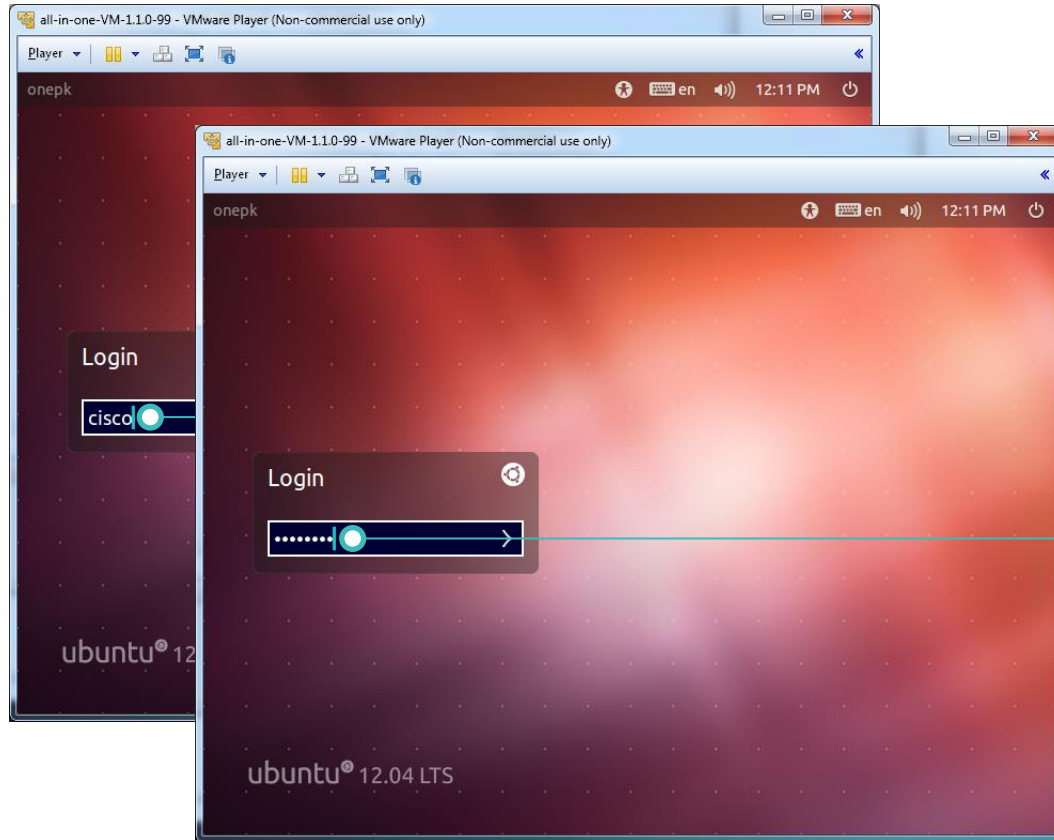
Import onePK AiO VM .ova File

# Getting Started with the onePK AiO VM – 2/x



Select then Play the onePK AiO VM

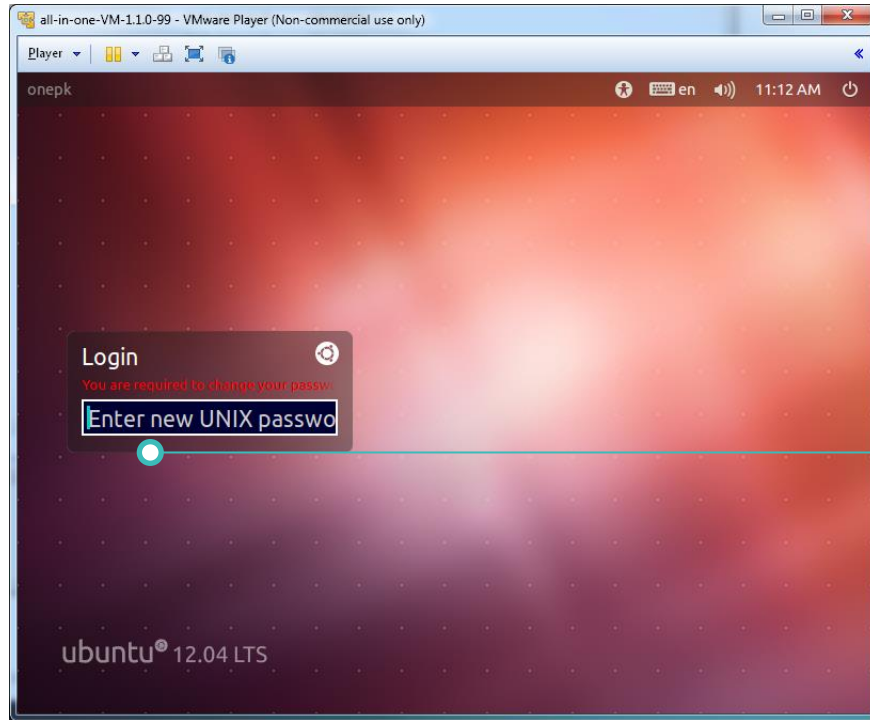
# Getting Started with the onePK AiO VM – 3/x



User: cisco

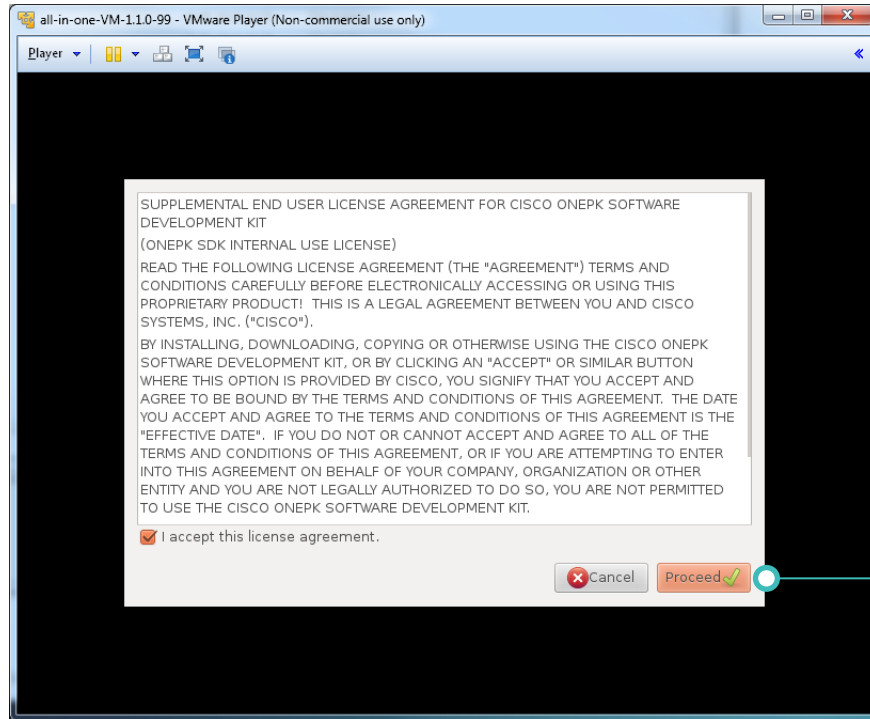
Password: cisco123

# Getting Started with the onePK AiO VM – 4/x



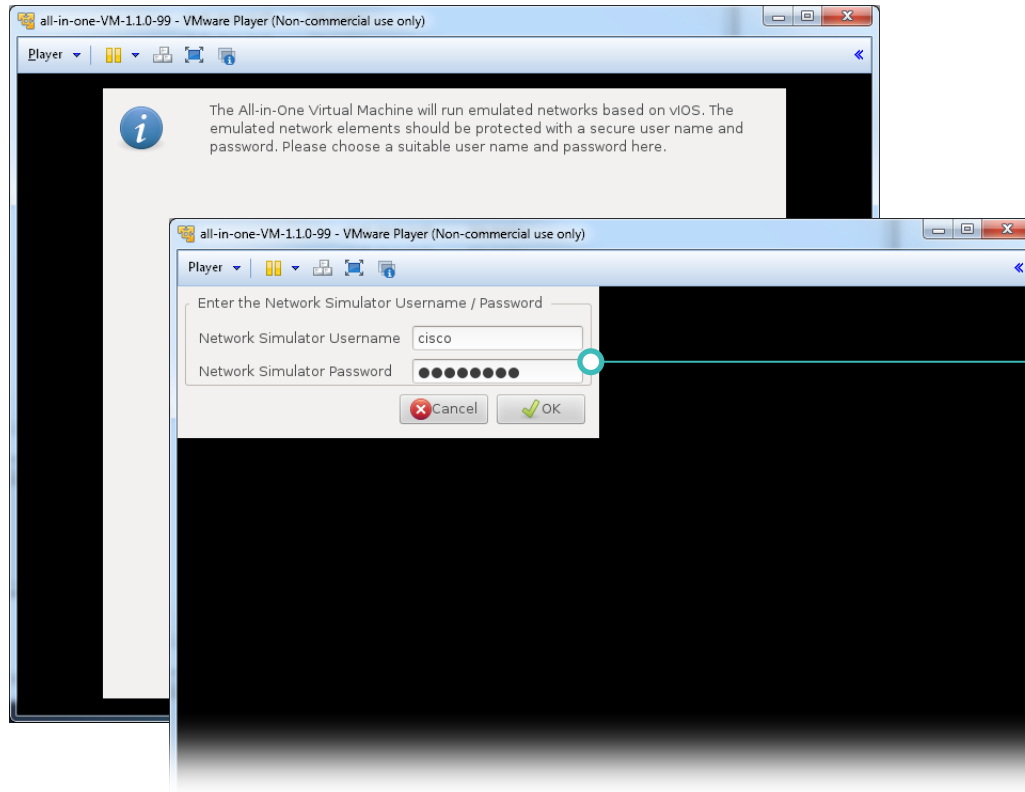
**Set New Password**  
(use cisco123 )

# Getting Started with the onePK AiO VM – 5/x



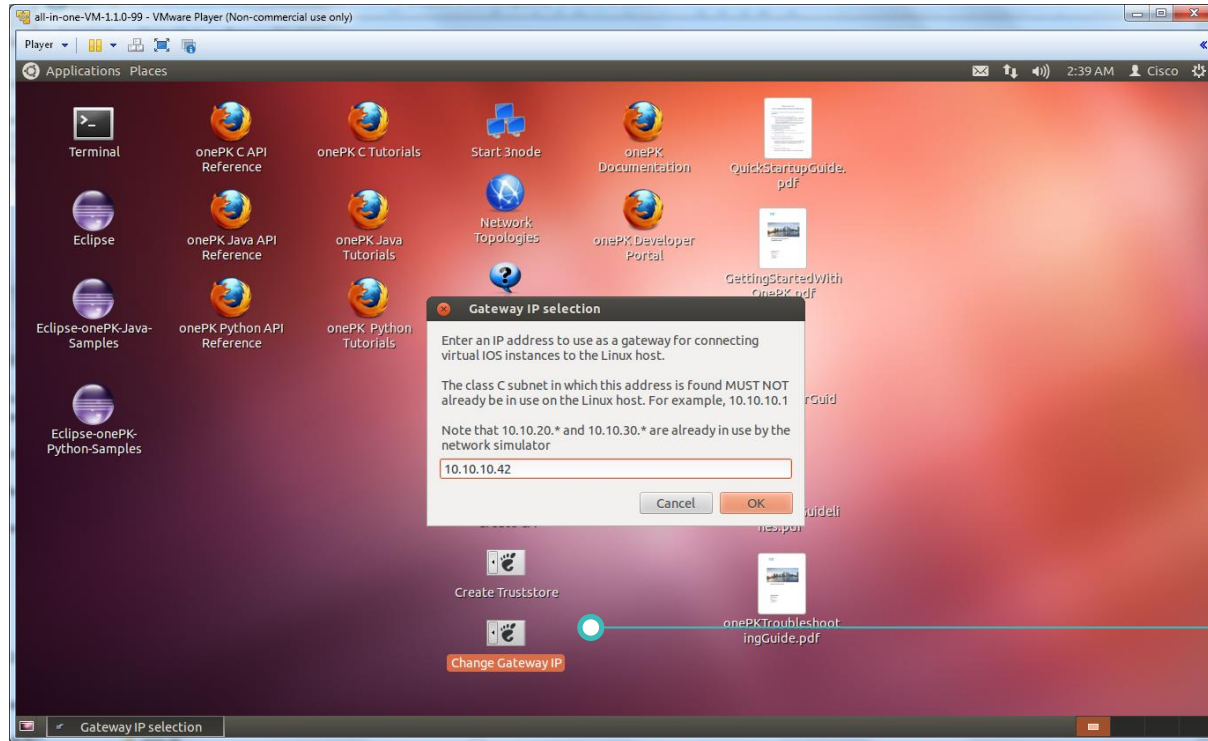
Accept License Agreement

# Getting Started with the onePK AiO VM – 6/x

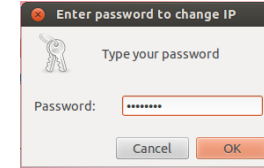


**Set vIOS User and Password**  
(use cisco / cisco123 )

# Getting Started with the onePK AiO VM – 7/x

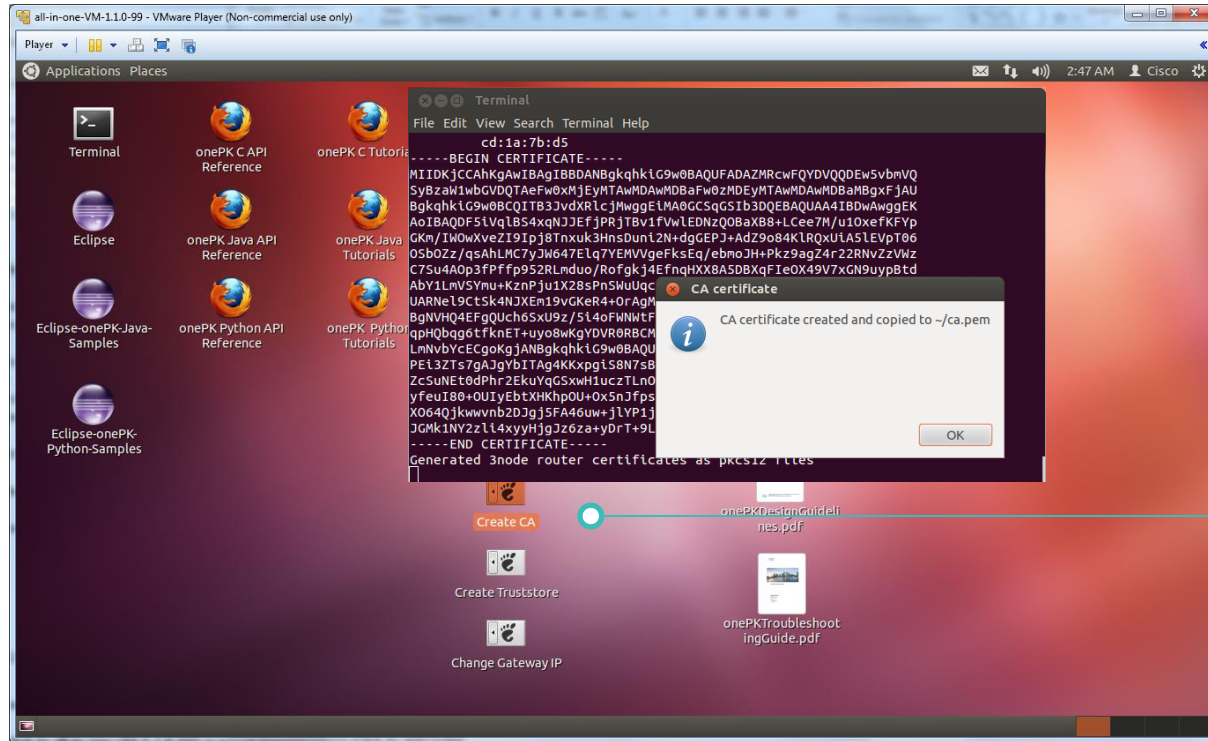


- Provide your Linux Password



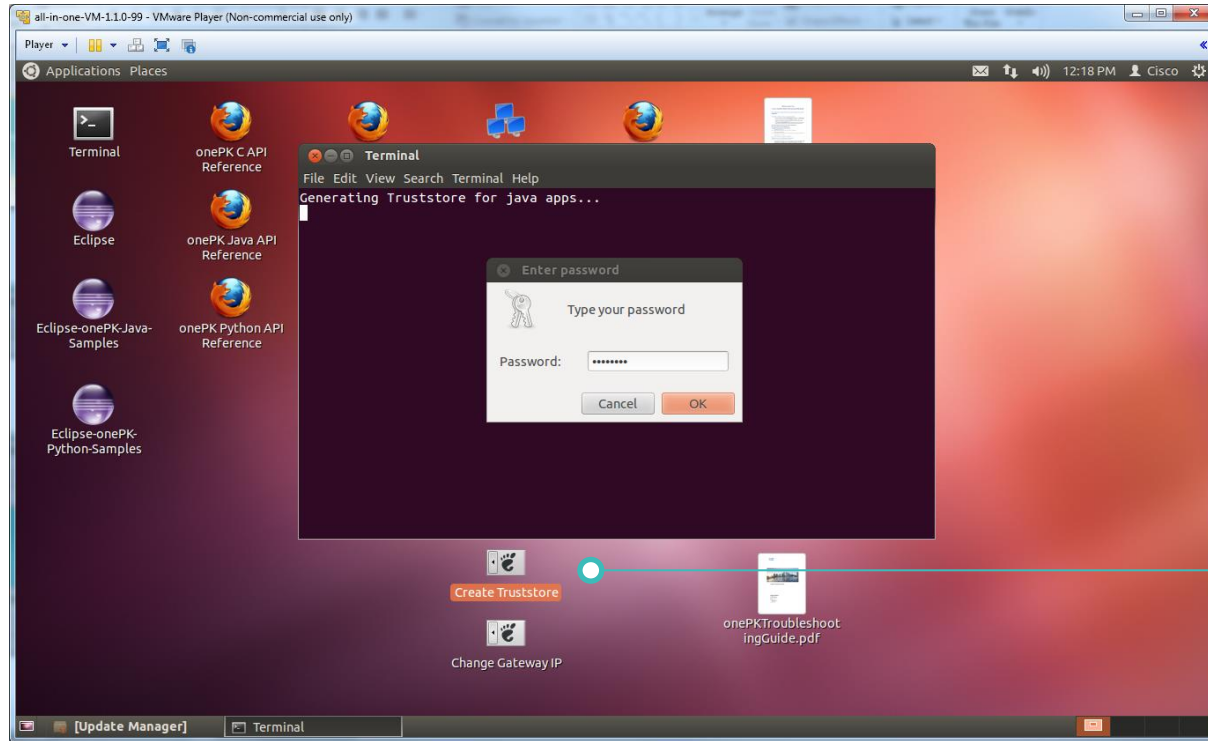
Set Gateway IP Address  
(use 10.10.10.42)

# Getting Started with the onePK AiO VM – 8/x



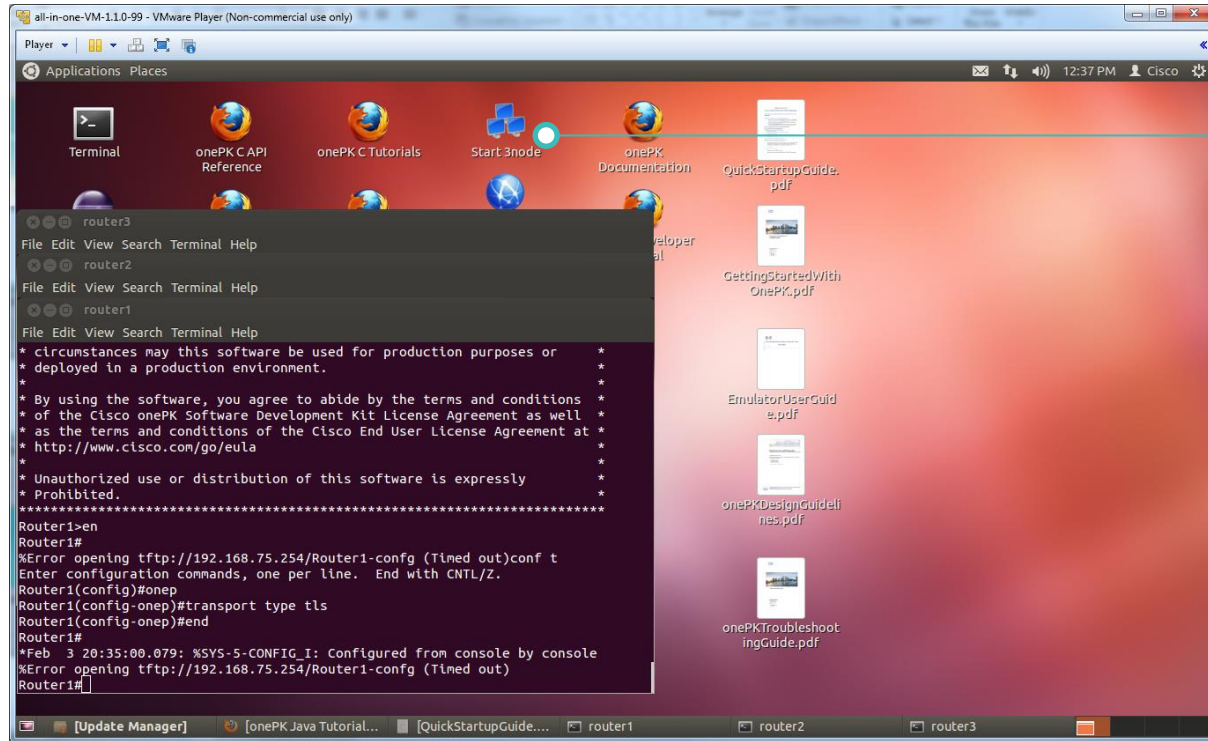
Create  
Certificate Authority

# Getting Started with the onePK AiO VM – 9/x



- onePK can use TLS to connect to the Network
- TLS uses Certificates
- Certificates for use with C and Python are pre-generated at `/home/cisco/ca.pem`
- For Java ...  
Run Create Truststore

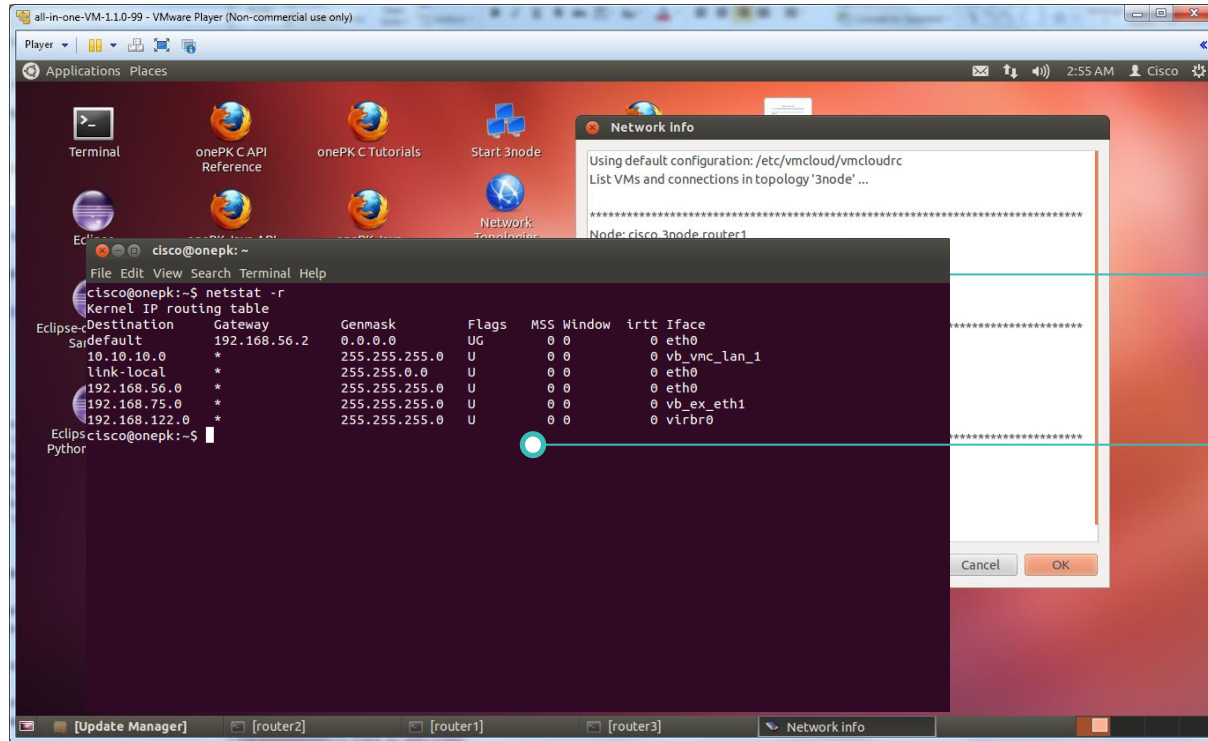
# Getting Started with the onePK AiO VM – 10/x



Run Start 3node

This may take time !

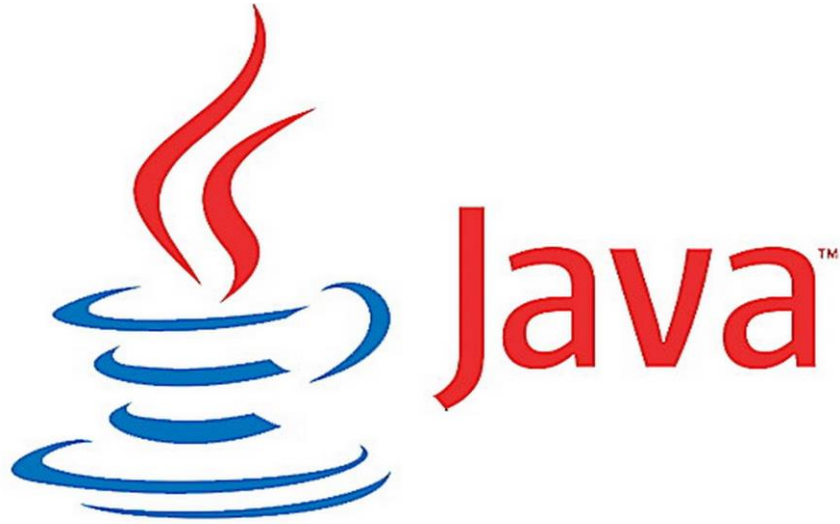
# Getting Started with the onePK AiO VM – 11/x



Confirm 3node Status

Confirm 3node  
Router Reachability  
(check for 10.10.10.0 route  
using netstat -r)

# Java @ onePK AiO VM

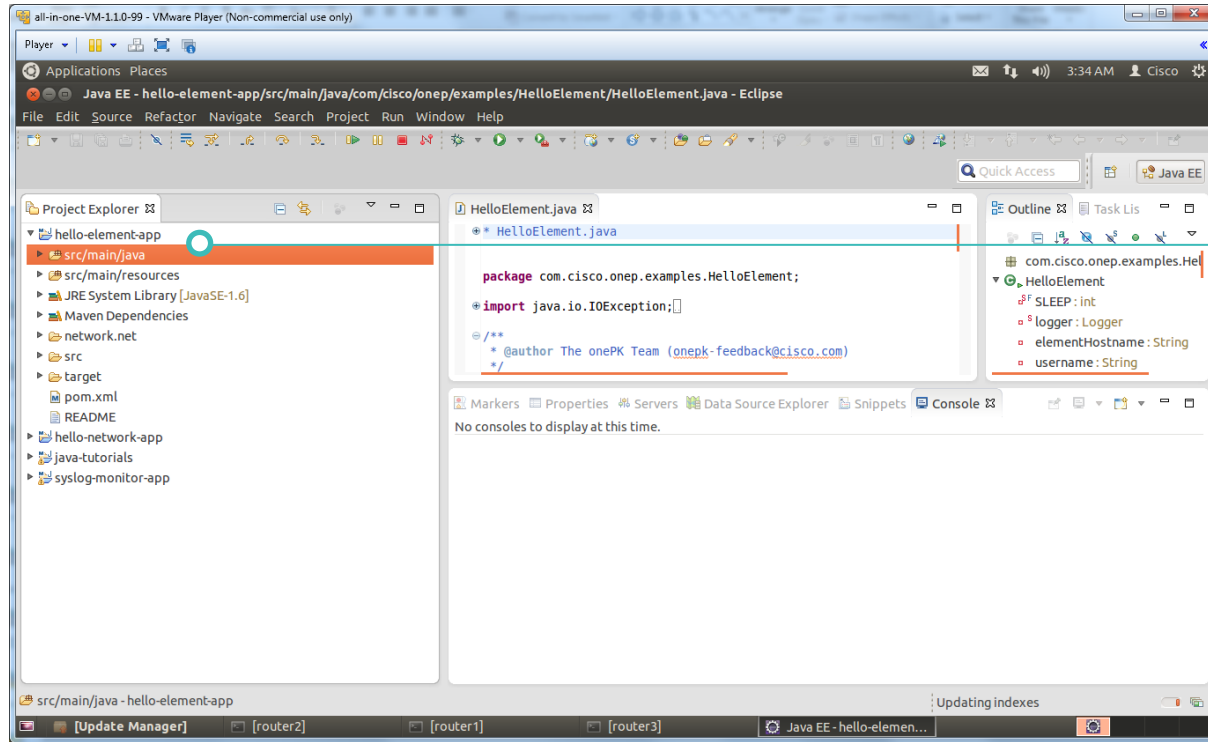


# Java @ onePK AiO VM – Hello Element Basics 1/3



Launch Eclipse  
Java Samples

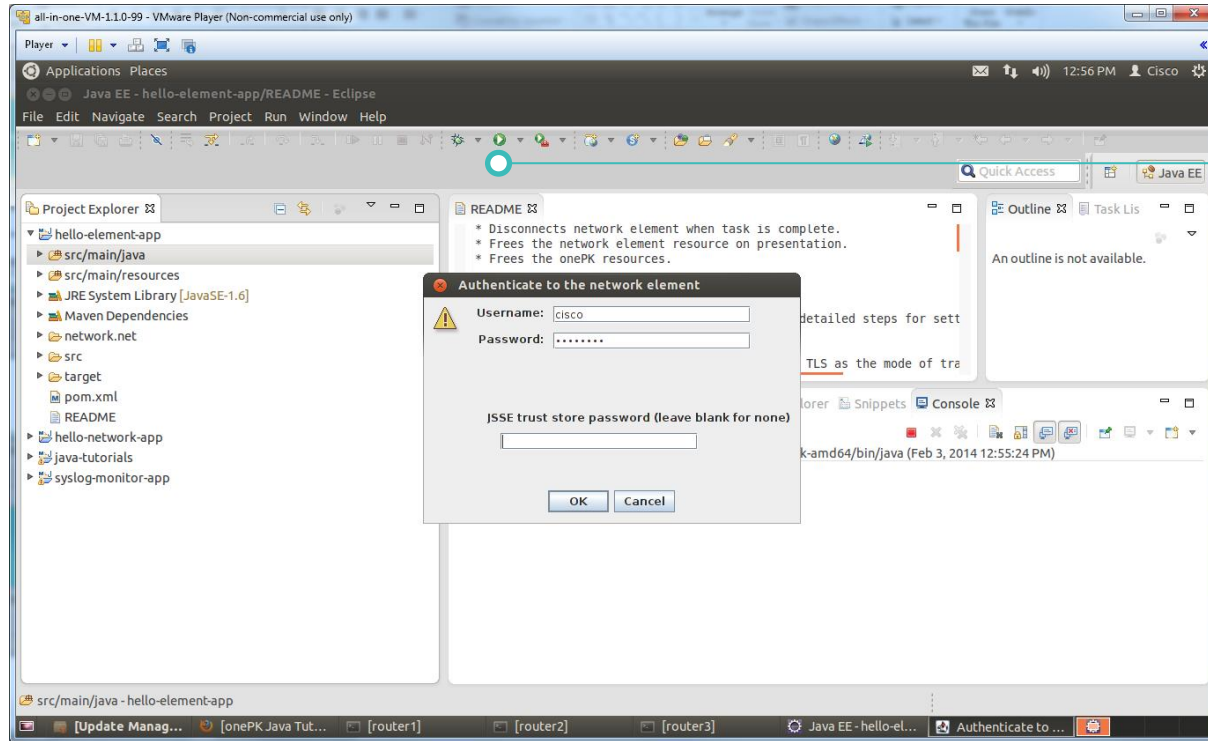
# Java @ onePK AiO VM – Hello Element Basics 2/3



Open Project  
hello-element-app

Select src/main/java

# Java @ onePK AiO VM – Hello Element Basics 3/3



## Run as Java Application

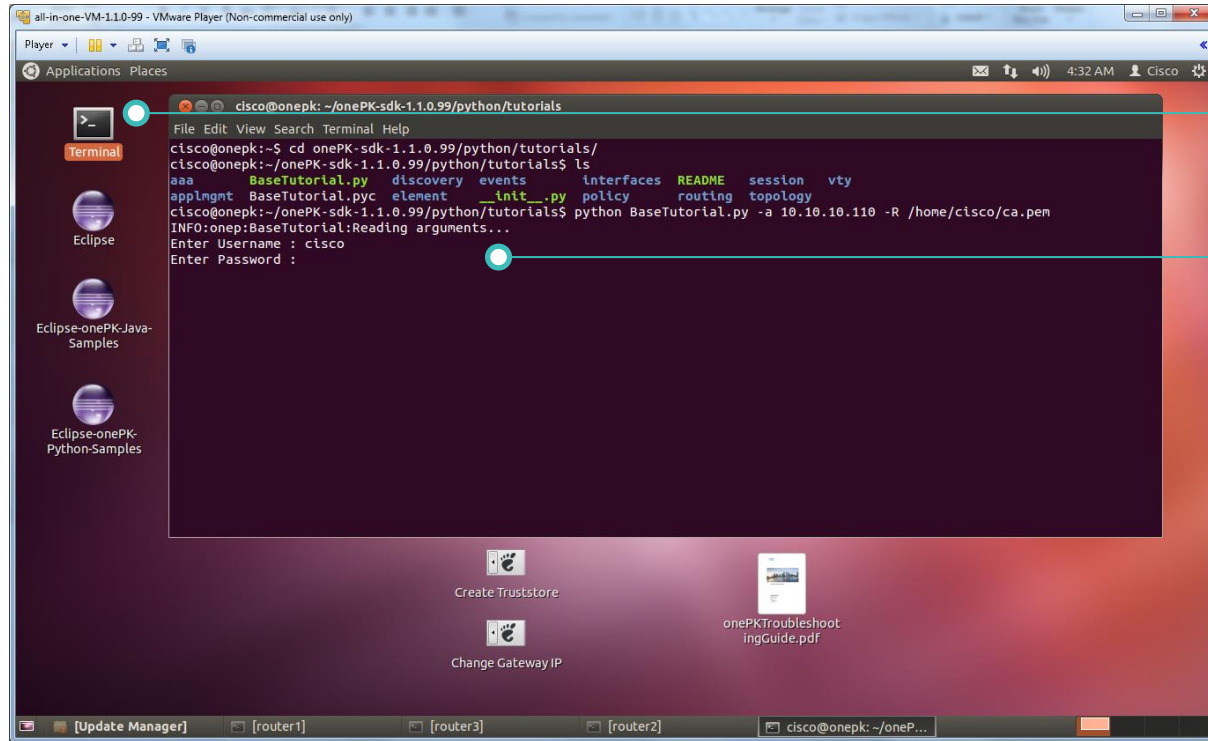
Compare the output with:

- The README
- Router 1# sh ver
- Router 1# sh onep history all
- The source code in HelloElement.java

# Python @ onePK AiO VM



# Python @ onePK AiO VM – x/x

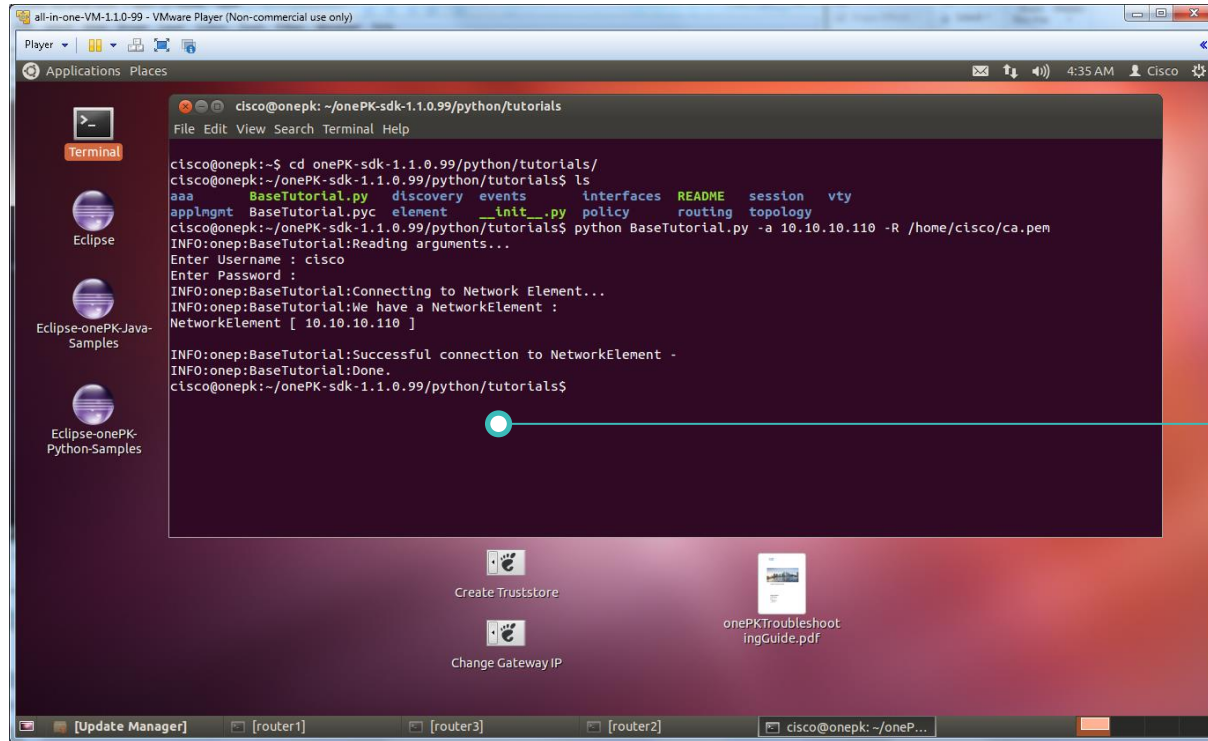


Open a Terminal

Navigate to  
Python Tutorials

Run BaseTutorial.py

# Python @ onePK AiO VM – x/x



```
cisco@onepk: ~/onePK-sdk-1.1.0.99/python/tutorials
File Edit View Search Terminal Help

cisco@onepk:~$ cd onePK-sdk-1.1.0.99/python/tutorials/
cisco@onepk:~/onePK-sdk-1.1.0.99/python/tutorials$ ls
aaa      BaseTutorial.py  discovery  events      interfaces  README      session  vty
applngmt BaseTutorial.pyc element     __init__.py policy      routing    topology
cisco@onepk:~/onePK-sdk-1.1.0.99/python/tutorials$ python BaseTutorial.py -a 10.10.110 -R /home/cisco/ca.pem
INFO:onep:BaseTutorial:Reading arguments...
Enter Username : cisco
Enter Password :
INFO:onep:BaseTutorial:Connecting to Network Element...
INFO:onep:BaseTutorial:We have a NetworkElement :
NetworkElement [ 10.10.110 ]

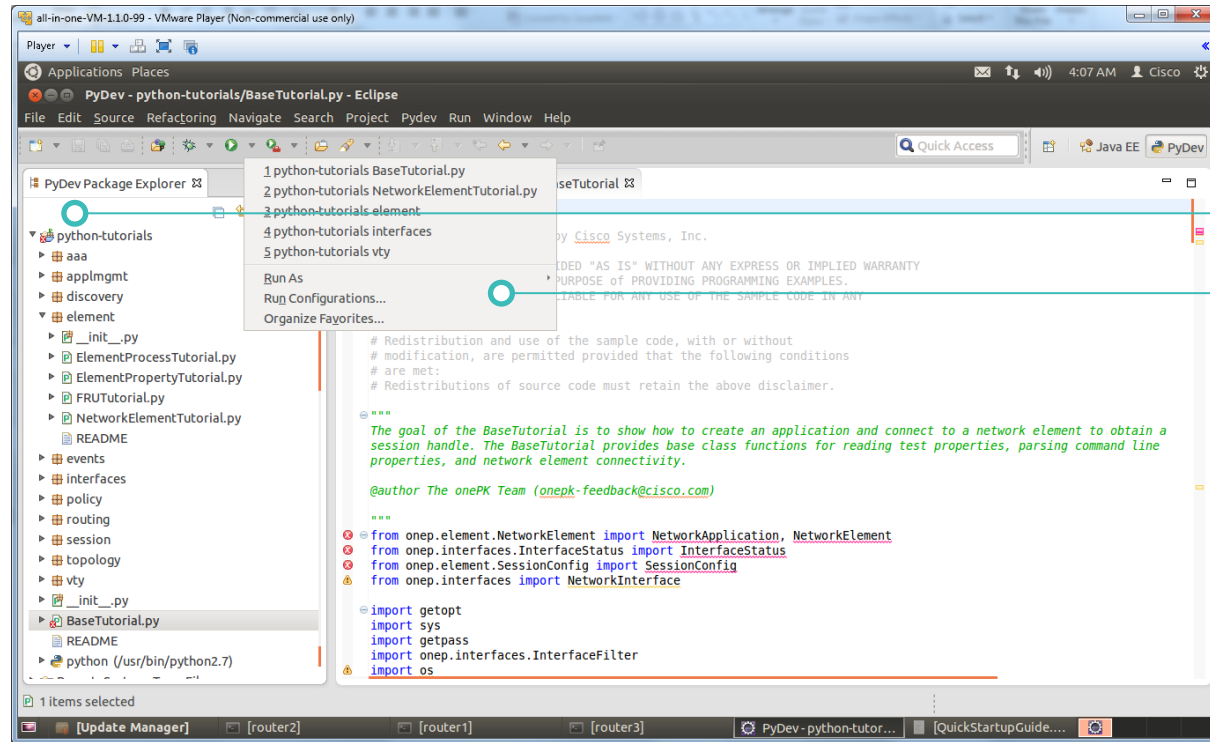
INFO:onep:BaseTutorial:Successful connection to NetworkElement -
INFO:onep:BaseTutorial:Done.
cisco@onepk:~/onePK-sdk-1.1.0.99/python/tutorials$
```

BaseTutorial.py will  
connect and disconnect

Compare with:

- Router 1# sh onep history all
- The script in BaseTutorial.py

# Python @ onePK AiO VM – x/x

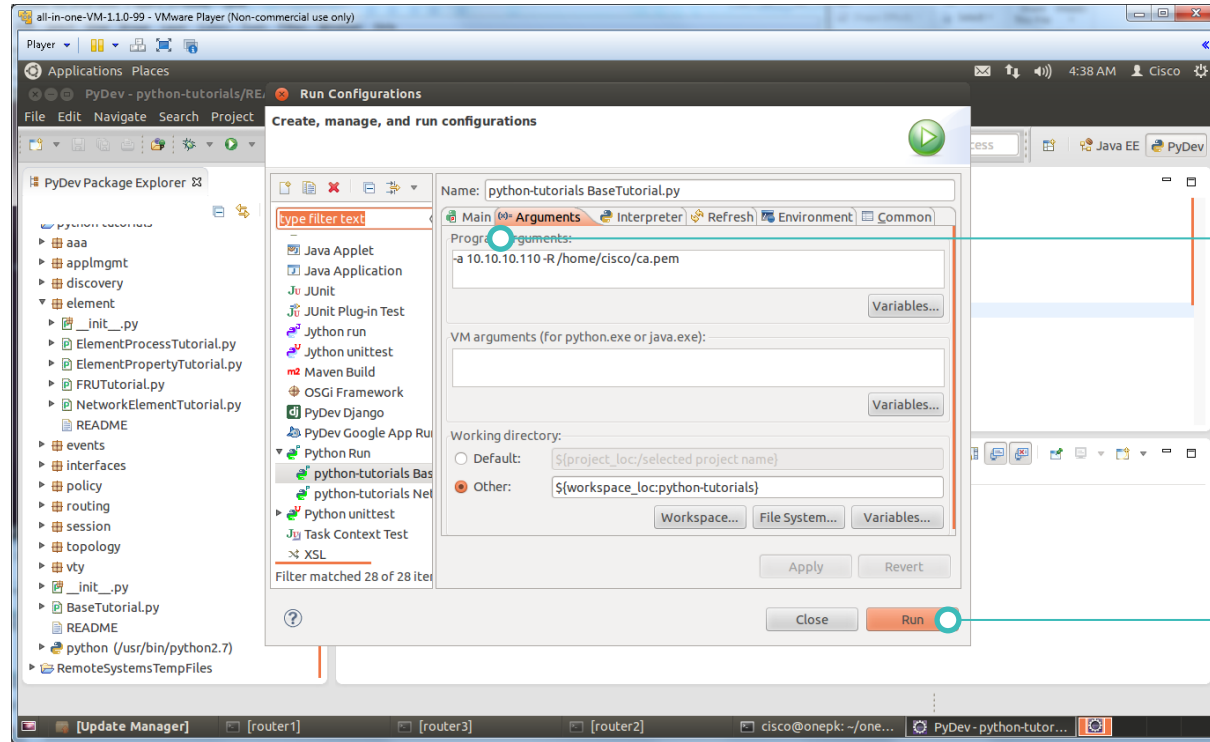


Open Python  
Tutorials in Eclipse

Select BaseTutorial.py

Navigate to  
Run Configurations ...

# Python @ onePK AiO VM – x/x

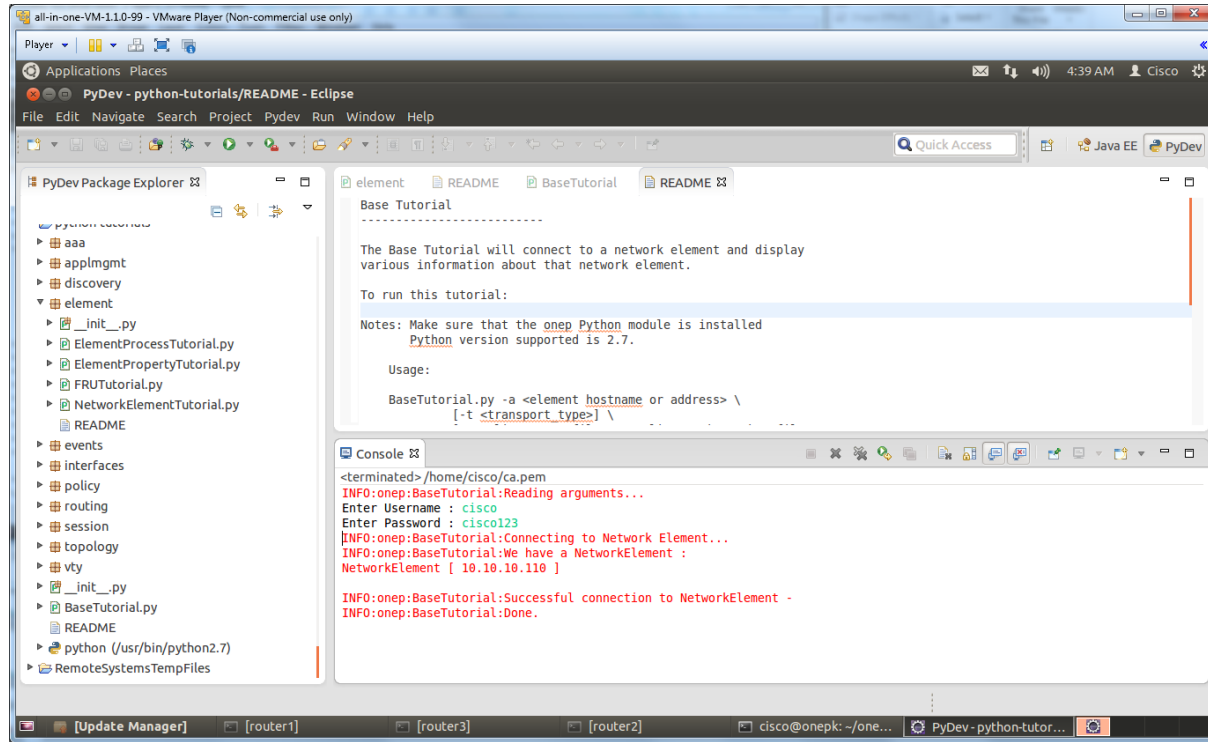


Switch to  
[Arguments] Tab

Add arguments for  
router IP (-a)  
Root CA file (-R)

Run from Eclipse

# Python @ onePK AiO VM – x/x



Compare with:

- Router 1# sh onep history all
- The script in BaseTutorial.py

From the Desktop, open onePK Tutorials in Firefox and explore ...



# Agenda

- Welcome
- Industry Observations
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- **Recent Industry Adoption**
- Autonomic Networking
- Resources and References

# Has This Ever Happened To You?



Have you ever been woken up for a VPN tunnel hit?



Have you ever been staring at a console for a memory leak to reoccur?



Have you ever been alerted to a problem by 1000 users calling the NOC?

# Things Start Small

# What Can Automation Do for Me?

**Challenge 1:** Every few weeks a router is running low on memory around 2 am, and I want to find out what's happening

- **Solution:** EEM policy could be triggered based on the memory utilization, capture the memory information and send the output with Syslog or Email

**Challenge 2:** My devices are running into a bug where “show ip ospf database” causes them to crash. I want to prevent the command from being run until I can upgrade

- **Solution:** EEM policy can trigger when “show ip ospf database” is executed and stop the command from running and the device from crashing.

**Challenge 3:** I want to devices to run an automated set of diagnostics that are periodically updated in a central database.

- **Solution:** OnePK application can be used to connect to the central database, extract the commands given the device's place in the network, run the diagnostics, and then report the results



# Automate Diagnostics

```
event manager applet LOW_IO_MEM
  event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val "4000000" poll-interval 60
  action 0.0 syslog msg "LOW MEMORY DETECTED. Please wait - logging information to flash:low_mem.txt"
  action 0.1 cli command "enable"
  action 0.2 cli command "term exec prompt timestamp"
  action 1.2 cli command "show memory statistics | append flash:low_mem.txt"
  action 1.3 cli command "show process mem sorted | append flash:low_mem.txt"
  action 2.3 cli command "show mem all total | append flash:low_mem.txt"
  action 3.2 cli command "show log | append flash:low_mem.txt"
  action 3.3 cli command "show tech | append flash:low_mem.txt"
  action 3.4 cli command "show mem debug leaks summ | append flash:low_mem.txt"
```

- Capture the required diagnostic information *at the time* a low I/O memory event occurs
- Save the data for future analysis and alert the operators that the problem has occurred
- This simple applet is extremely popular in TAC that use this every time they are diagnosing a low I/O memory case

# Automate Bug Workarounds

```
event manager applet workaround_CSCso53115
  event timer cron cron-entry @reboot
  action 1.0 cli command "enable"
  action 2.0 cli command "config t"
  action 3.0 cli command "no mgcp"
  action 4.0 cli command "mgcp"
  action 5.0 cli command "end"
```

- When bugs like CSCso53115 occur (and MGCP fails to reinitialize on reboot), EEM comes to the rescue
- Automate the workaround described in the bug
  - *“Once the router comes up, perform a “no mgcp” / “mgcp” to force the MGCP to reinitialize. This will cause MGCP to come up and work correctly.”*
- When the router reboots, EEM automatically reconfigures MGCP thus ensuring no user intervention is required

# More Devices, More Versions, More Automation

# Alert on a Route Change

```
event manager applet route_table_monitor
  event routing network 0.0.0.0/0 type all ge 1
  action 1.0 syslog msg "Route changed: Type: $_routing_type,
Network: $_routing_network, Mask/Prefix: $_routing_mask,
Protocol: $_routing_protocol, GW: $_routing_lastgateway,
Intf: $_routing_lastinterface"
```



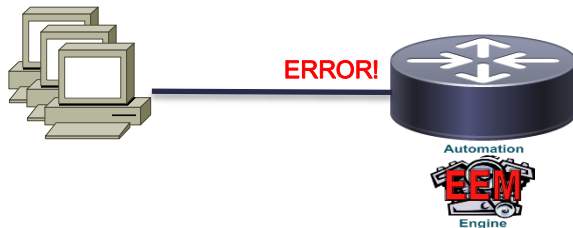
```
Jan  2 02:34:45.381: %HA_EM-6-LOG: route_table_monitor: Route changed: Type:
remove, Network: 10.14.1.0, Mask/Prefix: 255.255.255.0, Protocol: OSPF, GW:
10.14.1.1, Intf: GigabitEthernet0/0
```

- Do you know when a critical route goes away?
- Unfortunately, there are no built-in notifications when the routing table changes
- Use EEM to proactively notify operations when any change occurs

# Monitor an Interface for Errors

- Interface errors can be the silent killer
- As the interface takes errors, performance suffers but no alerts are seen
- Using EEM, we can monitor for increases in errors, send notifications, or trigger a fail-over

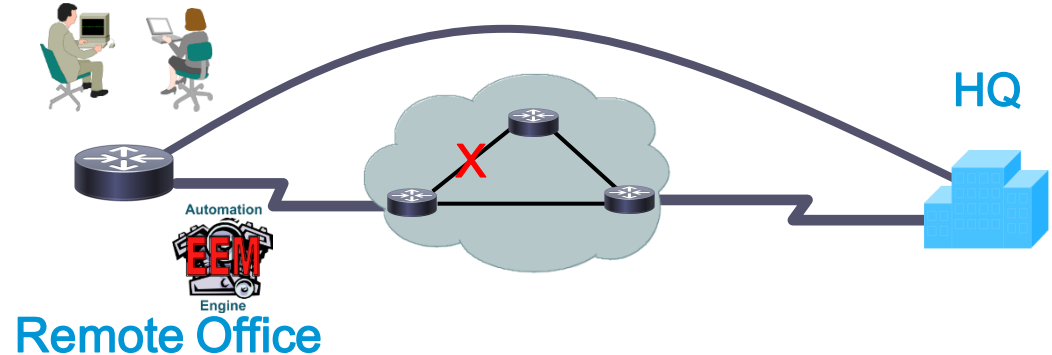
```
event manager applet error_monitor
  event interface name GigabitEthernet0/1 parameter input_errors
  entry-op ge entry-val 5 entry-type increment poll-interval 10
  action 1.0 syslog priority errors msg "Interface
GigabitEthernet0/1 has seen $_interface_delta_value input errors
in the past 10 seconds; failing over HSRP"
  action 2.0 cli command "enable"
  action 2.1 cli command "config t"
  action 2.2 cli command "int gi0/1"
  action 2.3 cli command "shut"
  action 2.4 cli command "end"
```



```
Jan  2 02:34:45.381: %HA_EM-3-LOG: error_monitor: Interface
GigabitEthernet0/1 has seen 7 input errors in the past 10
seconds; failing over HSRP
```

# Path Failover

- EEM, IPSLA, and enhanced object tracking can help fail over a path if an intermediate hop fails
- EEM can augment standard failover to make sure the existing path is stable enough before restoring main-path traffic flow



# Failover With Dampening

## IPSLA + Object Tracking

```
ip sla 1
  icmp-echo 10.1.1.1 source-interface Serial0/0
ip sla schedule 1 life forever start-time now
track 1 ip sla 1 reachability
```

# Failover With Dampening

## IPSLA + Object Tracking

```
ip sla  
icmp-  
ip sla  
track 1
```

### EEM “Down” Applet

```
event manager applet track_down  
  event track 1 state down  
  action 1.0 cli command "enable"  
  action 2.0 cli command "config t"  
  action 3.0 cli command "int tun0"  
  action 4.0 cli command "no shut"  
  action 5.0 cli command "no event manager applet track_timer"  
  action 6.0 cli command "end"
```

# Failover With Dampening

## IPSLA + Object Tracking

### EEM “Down” Applet

### EEM “Up” Applet

```
event manager environment q "  
event manager applet track_up  
  event track 1 state down  
  action 1.0 cli command "enable"  
  action 2.0 cli command "config t"  
  action 3.0 cli command "event manager applet track_timer"  
  action 3.1 cli command "event timer countdown time 300"  
  action 3.2 cli command "action 1.0 cli command enable"  
  action 3.3 cli command "action 2.0 cli command $q config t$q"  
  action 3.4 cli command "action 3.0 cli command $q int tun0$q"  
  action 3.5 cli command "action 4.0 cli command shut"  
  action 3.6 cli command "action 5.0 cli command $q no event  
manager applet track_timer$q"  
  action 3.7 cli command "action 6.0 cli command end"  
  action 3.8 cli command "end"
```

This is needed to embed  
quotes in EEM applets

Make sure the path  
is stable for five  
minutes

# Failover With Dampening

## IPSLA + Object Tracking

### EEM “Down” Applet

**Embedded Applet (Runs if the path is stable for five minutes)**

```
event manager applet track_timer
  event timer countdown time 300
  action 1.0 cli command enable
  action 2.0 cli command " config t"
  action 3.0 cli command " int tun0"
  action 4.0 cli command shut
  action 5.0 cli command " no event manager applet track_timer"
  action 6.0 cli command end
```

```
action 3.4 cli command "action 3.0 cli command $q int tun0$q"
action 3.5 cli command "action 4.0 cli command shut"
action 3.6 cli command "action 5.0 cli command $q no event
manager applet track_timer$q"
action 3.7 cli command "action 6.0 cli command end"
action 3.8 cli command "end"
```

# Diagnosing Transient Problems

**Problem:** Periodically, your network encounters strange problems that cause connectivity issues or performance problems. You'd like to be able to look at the traffic on the network when the problem is occurring. Unfortunately, hindsight is always 20/20, but it doesn't allow you to go back in time to put a sniffer on the network ahead of the problem ☹.

# Diagnosing Transient Problems

## Doing Things the Long Way



**But the sniffer captures the traffic *AFTER* the problem occurred!**

# Diagnosing Transient Problems

**Solution:** Use Embedded Packet Capture together with Embedded Event Manager to create an always-running “Digital Packet Recorder.” Add logic to recognize the event you want, and then stop the capture. The resulting capture file can be emailed to you for analysis at a convenient time (think MythTV for your network 😊).

# Troubleshooting Transient Problems

## Configuring EPC

- Setup EPC to use a circular buffer, and run forever
- Capture on all interfaces so that nothing gets lost
  - *You can pick a specific interface if you want*
- Start the capture manually, and then EEM will take care of stopping the capture

```
Router#monitor capture point ip cef cappnt all both
Router#monitor capture buffer capbuf size 512 max-size 1518
circular
Router#monitor capture point associate cappnt capbuf
Router#monitor capture point start cappnt
```

# Troubleshooting Transient Problems

## Stopping the Capture With EEM

```
::cisco::eem::event_register_syslog pattern "%CRYPTO-4-RECVD_PKT_MAC_ERR: decrypt: mac  
verify failed"
```

```
namespace import ::cisco::eem::*  
namespace import ::cisco::lib::*
```

```
if { [catch {cli_open} result] } {  
    error "Failed to open CLI session: '$result'" $errorInfo  
}
```

```
array set cliarr $result
```

```
if { [catch {cli_exec $cliarr(fd) "enable"} result] } {  
    error "Failed to enable CLI session: '$result'" $errorInfo  
}
```

```
if { [catch {cli_exec $cliarr(fd) "monitor capture point stop cappnt"} result] } {  
    error "Failed to start packet capture: '$result'" $errorInfo  
}
```

```
...
```

React to a syslog  
indicating that the  
specific problem has  
occurred

Stop the capture

# Troubleshooting Transient Problems

## Send the Capture File As an Email

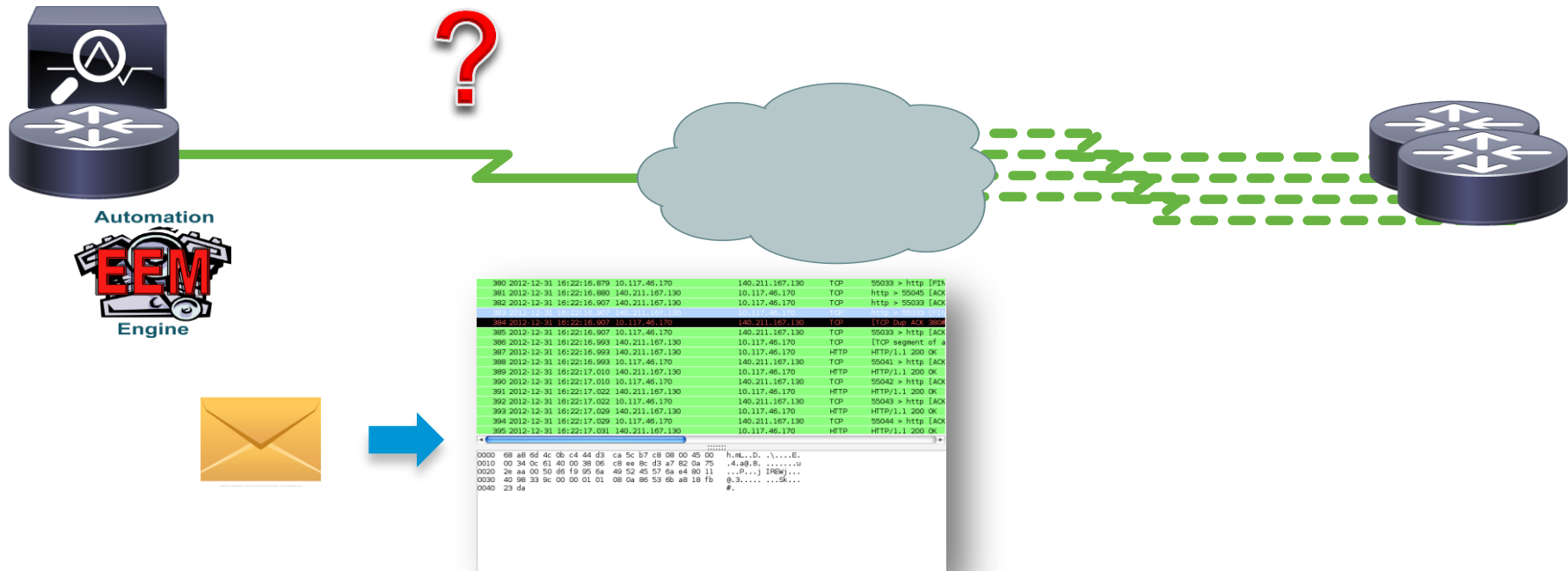
```
if { [catch {cli_exec $cliarr(fd) "monitor capture buffer capbuf export flash:dpr.pcap"} result] } {  
    error "Failed to export capture buffer: '$result'" $errorInfo  
}  
set email_body_pre "Mailservername: $_email_server  
From: $_email_from  
To: $_email_to  
Cc:  
Subject: Email from Router $_router_name  
MIME-Version: 1.0  
Content-type: multipart/mixed; boundary=\"EEM_email_boundary\"  
\n--EEM_email_boundary\n  
\n--EEM_email_boundary  
Content-Type: application/octet-stream  
Content-Transfer-Encoding: Base64  
Content-Disposition: attachment; filename=\"dpr.pcap\" \"\n\n  
if [catch {open "flash:vpn.pcap"} result] {  
    error "Failed to open flash:dpr.pcap: '$result'" $errorInfo  
}  
set fd $result  
fconfigure $fd -translation binary  
set email_b64 [::base64::encode [read $fd]]  
close $fd  
set email_body_mime "\n--EEM_email_boundary--"  
set email_body [format "%s%s%s" $email_body_pre $email_b64 $email_body_mime]  
if [catch {smtp_send_email $email_body} result] {  
    error "Failed to send email: '$result'" $errorInfo  
}
```

Export the capture  
to a file on flash

Create an email with  
a MIME header

Send the email

# Diagnosing Transient Problems With Your DPR



# Enabling Debug on a Remote Device

**Problem:** A device on the network is experiencing a connectivity loss or error condition. In order to fully diagnose the problem, debugging must be enabled on this device when the problem is occurring; *BUT* debugging must also be enabled on a peer device at the same time.

**Solution:** Using the Embedded Event Manager and SNMP traps, it is possible to link multiple devices together using a light-weight Remote Procedure Call (RPC) mechanism. The device experiencing the problem can enable debugging, then send an SNMP trap to its peer to do the same.

# Enabling Debug on a Remote Device

- It is possible to trigger an EEM on one device from another device.
- This is most easily done by sending an EEM trap from one device, and intercepting the trap using the SNMP-Notification ED on the second device
- Requires EEM 2.4 or higher on the second device, but only EEM 1.0 on the device sending the trap



Additional Reference: <http://tools.cisco.com/squish/7b686>

© 2013-2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public 143

## Enabling Debug on a Remote Device (cont.)

- When one device detects a condition, use SNMP traps to enable debugging on a peer device

### Source Router:

```
event manager applet ospf-change
  event syslog pattern "OSPF-5-ADJCHG"
  action 1.0 cli command "enable"
  action 2.0 cli command "debug ip ospf packet"
  action 3.0 snmp-trap intdata1 424242
!
snmp-server enable traps event-manager
snmp-server host 10.1.1.1 traps public event-manager
```

## Enabling Debug on a Remote Device (cont.)

- Receive the EEM trap on the peer device

Peer Router:

ceemHistoryPolicyIntData1

```
event manager applet ospf-change
  event snmp-notification oid 1.3.6.1.4.1.9.10.91.1.2.3.1.9.
oid-val "424242" op eq src-ip-address 10.1.1.2
  action 1.0 cli command "enable"
  action 2.0 cli command "debug ip ospf packet"
!
snmp-server manager
```

MIB Object Navigator : <http://tools.cisco.com/squish/7f7ef>

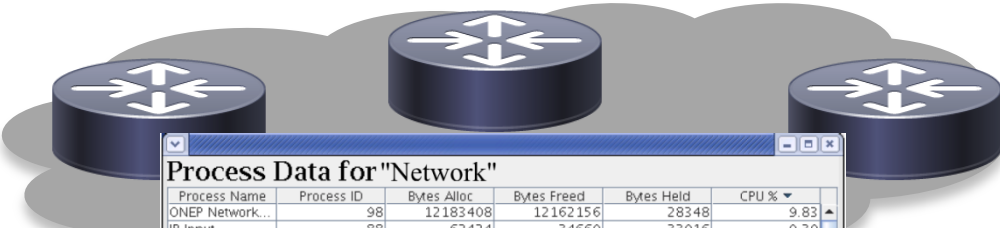
# Remote Command Execution

- If traps (and UDP) are not your thing, use telnet or SSH to do the synchronization the *EASy* way
- Automate commands on another (maybe non-IOS) device using EEM from a remote device
- EEM can automate connections across multiple protocols
  - Telnet
  - Reverse telnet
  - Session
  - Remote console
  - Remote commands
  - SSH (15.1(4)T and 15.2(2)T and higher)
- EASy package exists to help with this
  - EASy Command Shell
  - Download from <http://tools.cisco.com/squish/b35c5>



# The Network Programmability Ramp

# A Network “Top”

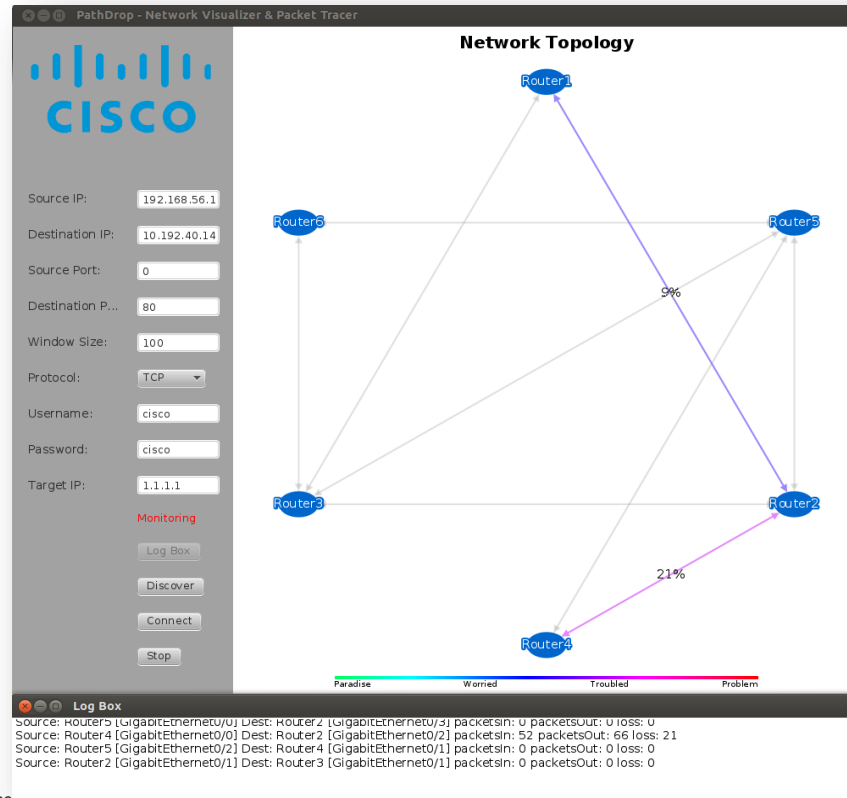


Process Data for "Network"

| Process Name      | Process ID | Bytes Alloc | Bytes Freed | Bytes Held | CPU % |
|-------------------|------------|-------------|-------------|------------|-------|
| ONEP Network...   | 98         | 12183408    | 12162156    | 28348      | 9.83  |
| IP Input          | 88         | 62424       | 34660       | 33016      | 0.39  |
| IP ARP Retry A... | 87         | 49444       | 0           | 56676      | 0.07  |
| PRR BR Learn      | 168        | 472         | 240         | 4464       | 0.07  |
| RMON Recycle ...  | 226        | 240         | 240         | 7232       | 0.07  |
| Per-Second Jobs   | 263        | 0           | 0           | 10232      | 0.07  |
| Chunk Manager     | 1          | 78824       | 0           | 86056      | 0     |
| Load Meter        | 2          | 240         | 240         | 4232       | 0     |
| Exec              | 3          | 111568      | 25000       | 111664     | 0     |
| Check heaps       | 4          | 4684        | 240         | 11676      | 0     |
| Pool Manager      | 5          | 457376      | 513008      | 34136      | 0     |
| DiscardQ Back...  | 6          | 0           | 0           | 7232       | 0     |
| Timers            | 7          | 240         | 240         | 7232       | 0     |
| WATCH_AFS         | 8          | 0           | 0           | 4232       | 0     |
| ARP Input         | 9          | 1145936     | 1125884     | 27284      | 0     |
| ARP Background    | 10         | 2224        | 2224        | 7232       | 0     |
| ATM Idle Timer    | 11         | 240         | 240         | 7232       | 0     |
| ATM_ASYNC PR...   | 12         | 0           | 0           | 7232       | 0     |
| AAA_SERVER....    | 13         | 0           | 0           | 7232       | 0     |
| Policy Manager    | 14         | 0           | 0           | 13232      | 0     |
| DDR Timers        | 15         | 240         | 240         | 7232       | 0     |
| Entity MIB API    | 16         | 21612       | 0           | 28844      | 0     |
| PrstVbl           | 17         | 240         | 240         | 7232       | 0     |
| RO Notify Timers  | 18         | 0           | 0           | 7232       | 0     |
| RMI RM Notify ... | 19         | 0           | 0           | 4232       | 0     |
| ATM_ASYNC B...    | 20         | 240         | 240         | 16232      | 0     |

- Use onePK to build a live process monitor similar to UNIX *top*
- The same app can connect to multiple devices to display the top processes across the entire network

# Troubleshooting Packet Loss Along a Path



- Use onePK to trace an application flow's path through the network
- Instruct the hops in the path to “be on the lookout” for that application's flow specification
- Perform a diagnostic session while the problem is reproduced
- The app then sees what nodes are dropping packets

# Agenda

- Welcome
- Industry Observations
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- **Autonomic Networking**
- Resources and References

# What Can Automation Do for Me?

**Challenge 1:** Every few weeks a router is running low on memory around 2 am, and I want to find out what's happening

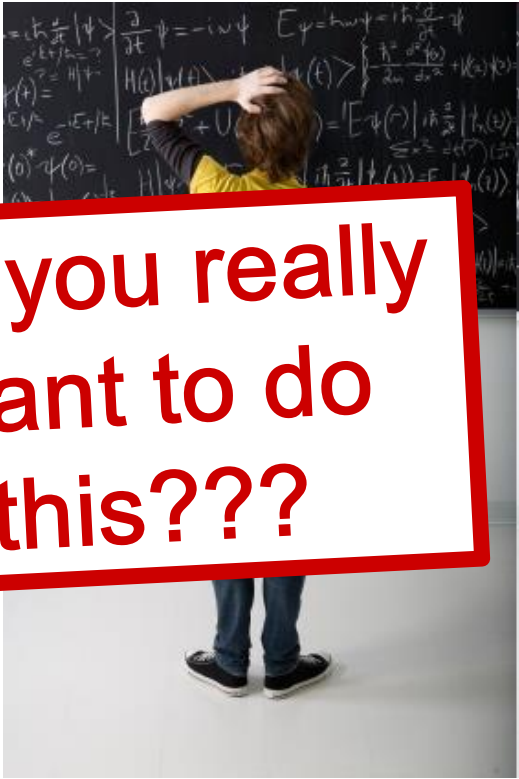
- **Solution:** EEM policy could be triggered based on the memory utilization, capture the memory information and send the output with Syslog or Email

**Challenge 2:** My devices are running into a bug where “show ip ospf database” causes them to crash. I want to prevent the command from being run until I can upgrade

- **Solution:** EEM policy can trigger when “show ip ospf database” is executed and stop the command from running and the device from crashing.

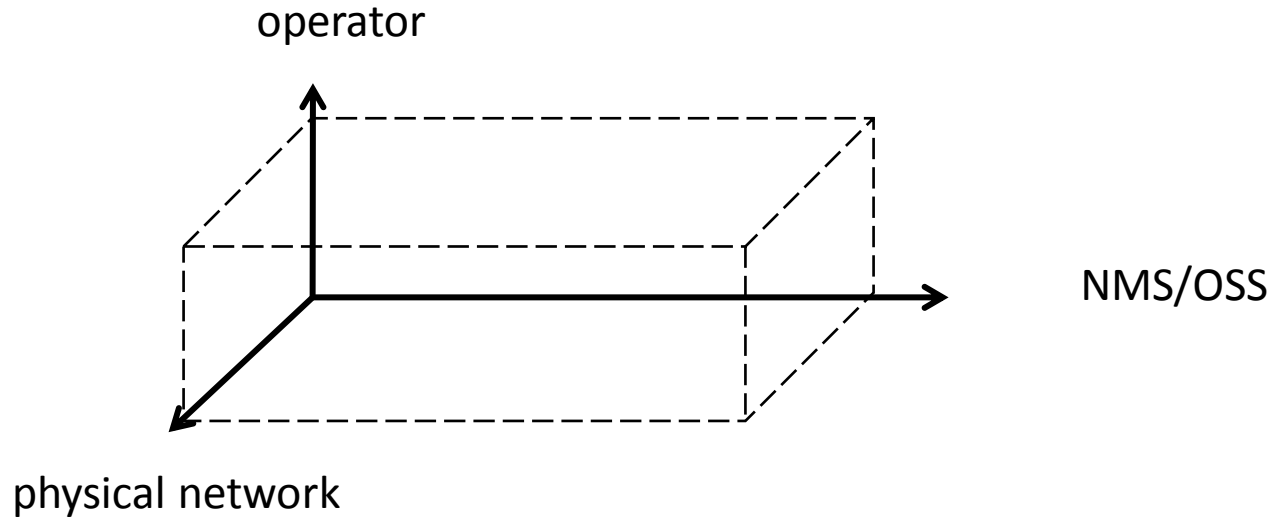
**Challenge 3:** I want to devices to run an automated set of diagnostics that are periodically updated in a central database.

- **Solution:** OnePK application can be used to connect to the central database, extract the commands given the device's place in the network, run the diagnostics, and then report the results



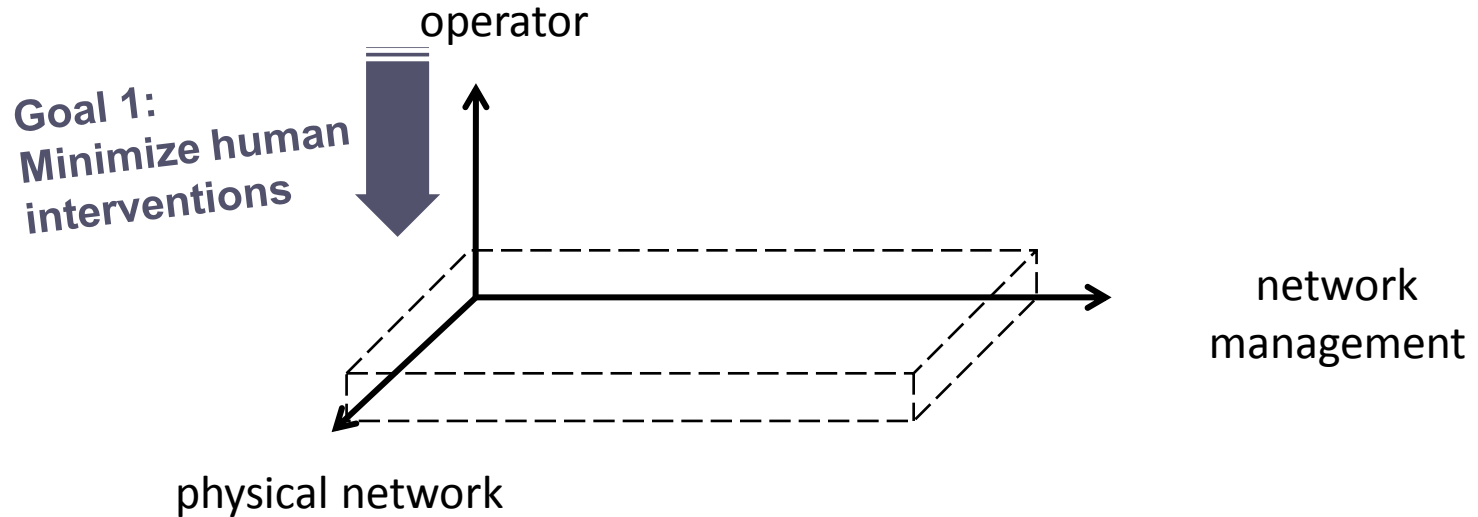
Do you really  
want to do  
this???

# Managing Complexity with Autonomics



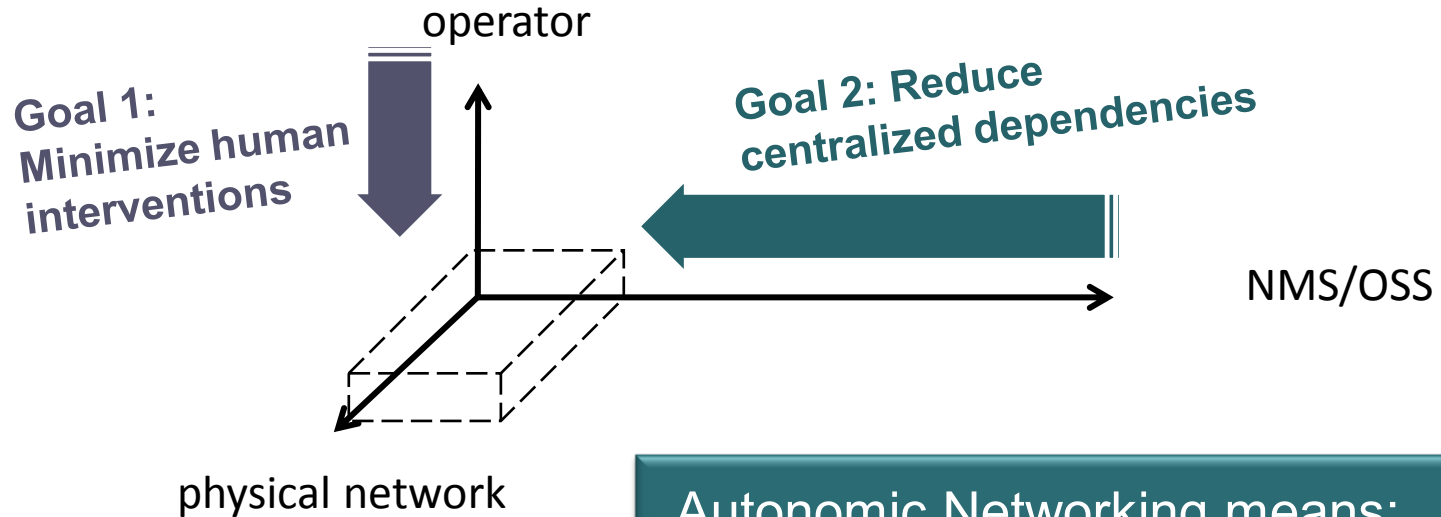
The “Complexity Cube”

# Managing Complexity with Autonomics



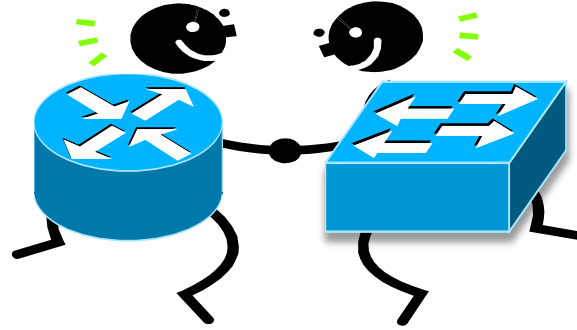
The “Complexity Cube”

# Managing Complexity with Autonomics

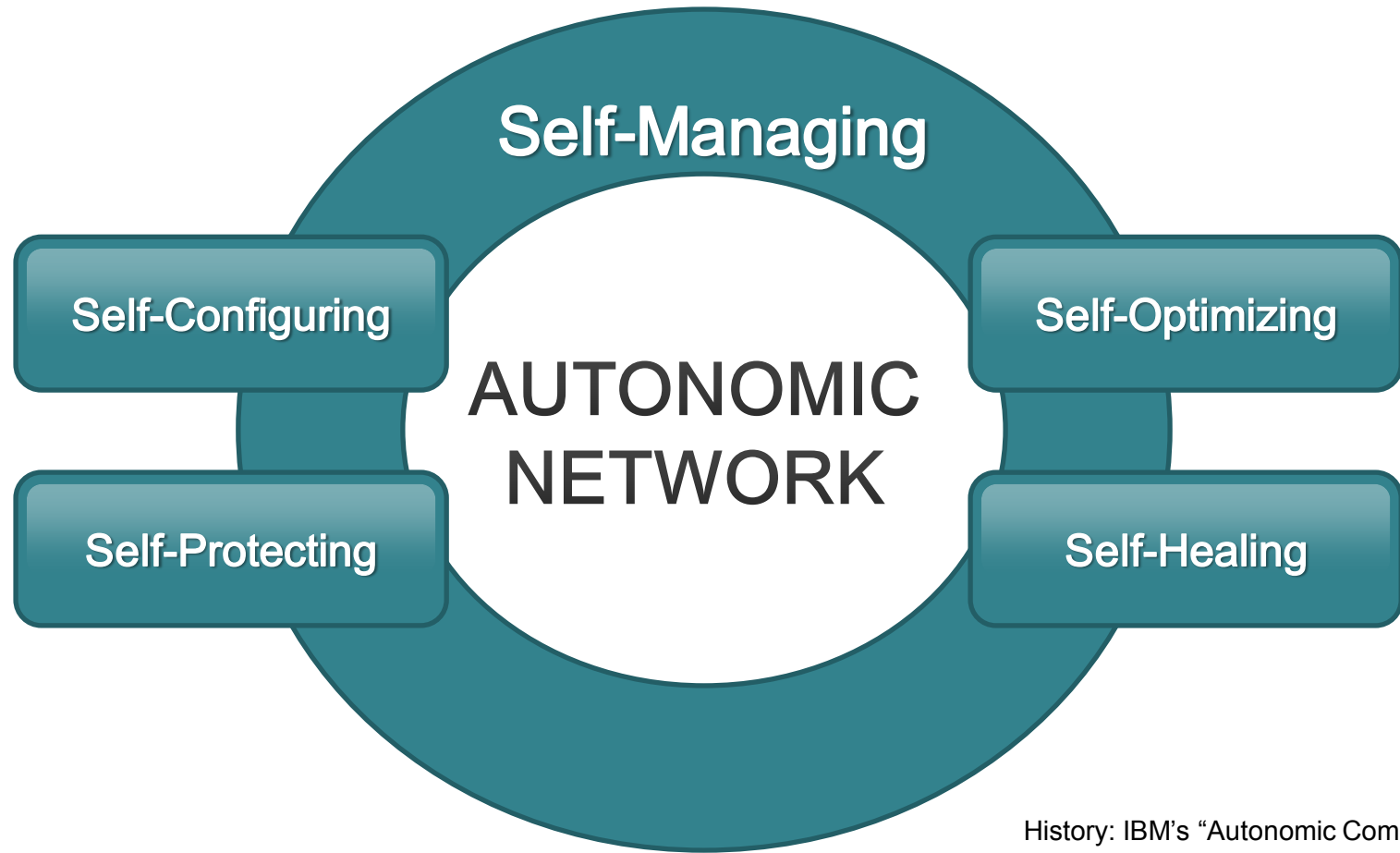


Autonomic Networking means:

- Minimize operator interventions
- Minimize NMS dependencies



Autonomics = Automation inside the network



History: IBM's "Autonomic Computing" (2001)  
<http://www.research.ibm.com/autonomic/>

## Google Cars Drive Themselves, in Traffic



## Amazon testing drones for deliveries

COMMENTS (1023)



Drone delivery - fact or fiction?

Amazon, the world's largest online retailer, is testing unmanned drones to deliver goods to customers, Chief Executive Jeff Bezos says.

### Related Stories

## NEWS TECHNOLOGY

Home UK Africa Asia Europe Latin America Mid-East US & Canada Business

13 May 2013 Last updated at 12:15 GMT

3.5K Share

## Pilotless flight trialled in UK shared airspace



Peter Marshall speaks to Astraea director Lambert Dopping-Hepenstal

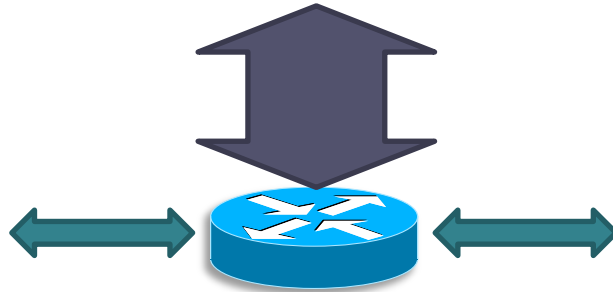
A Jetstream aircraft became the first to fly "unmanned" across UK shared airspace last month.

### Related

**DO NOT  
PANIC**

## Traditional

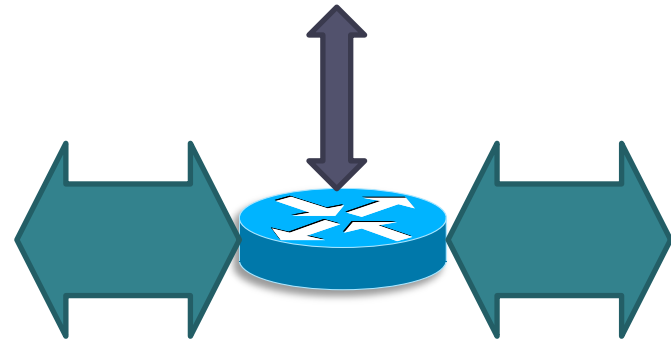
- Configuration
- Monitoring
- Reporting



- Routing

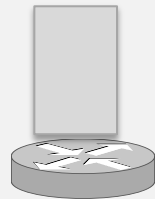
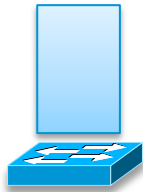
## Autonomic

- Policy and Service Orchestration
- Aggregated Reporting

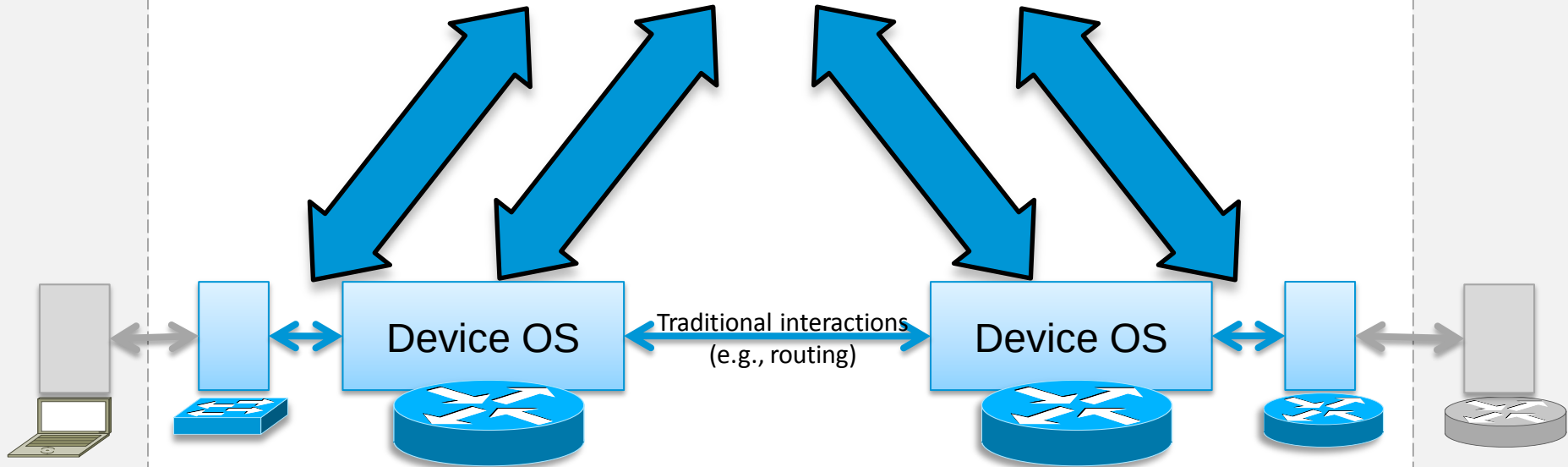
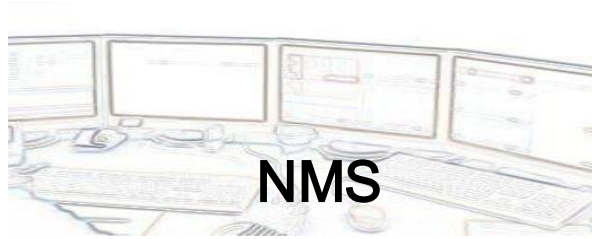


- Routing
- Discovery
- Autonomic interactions

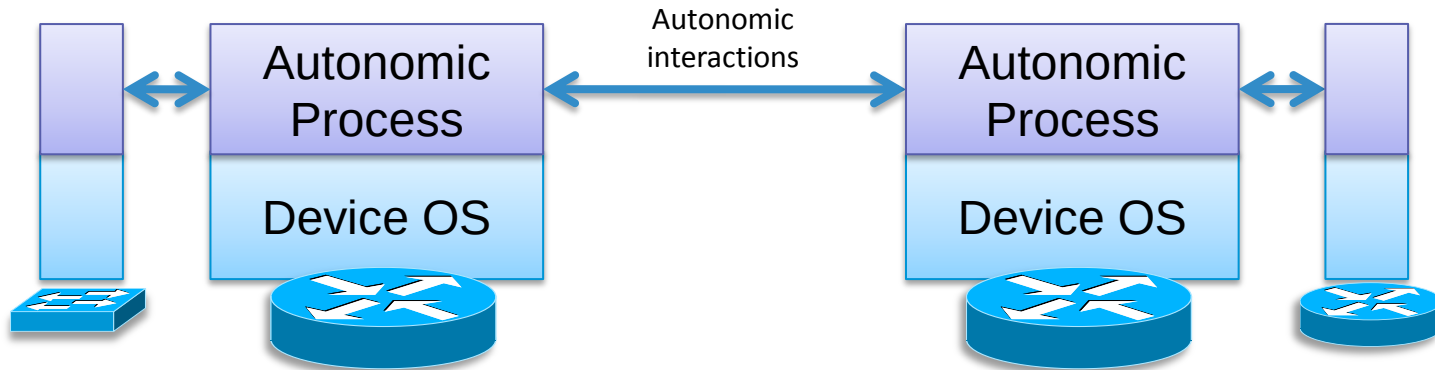
# Traditional Networks



# Traditional Networks

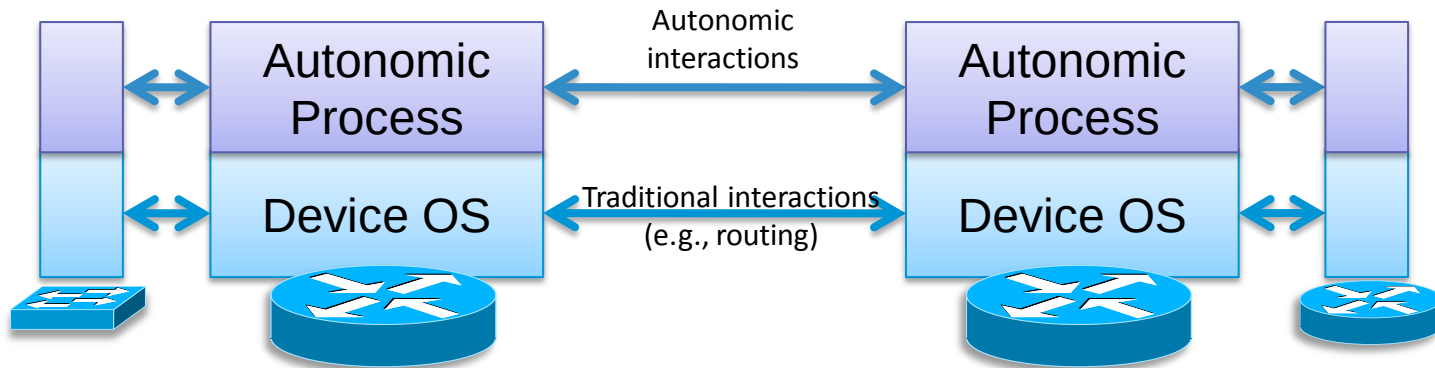


# The Autonomic Network



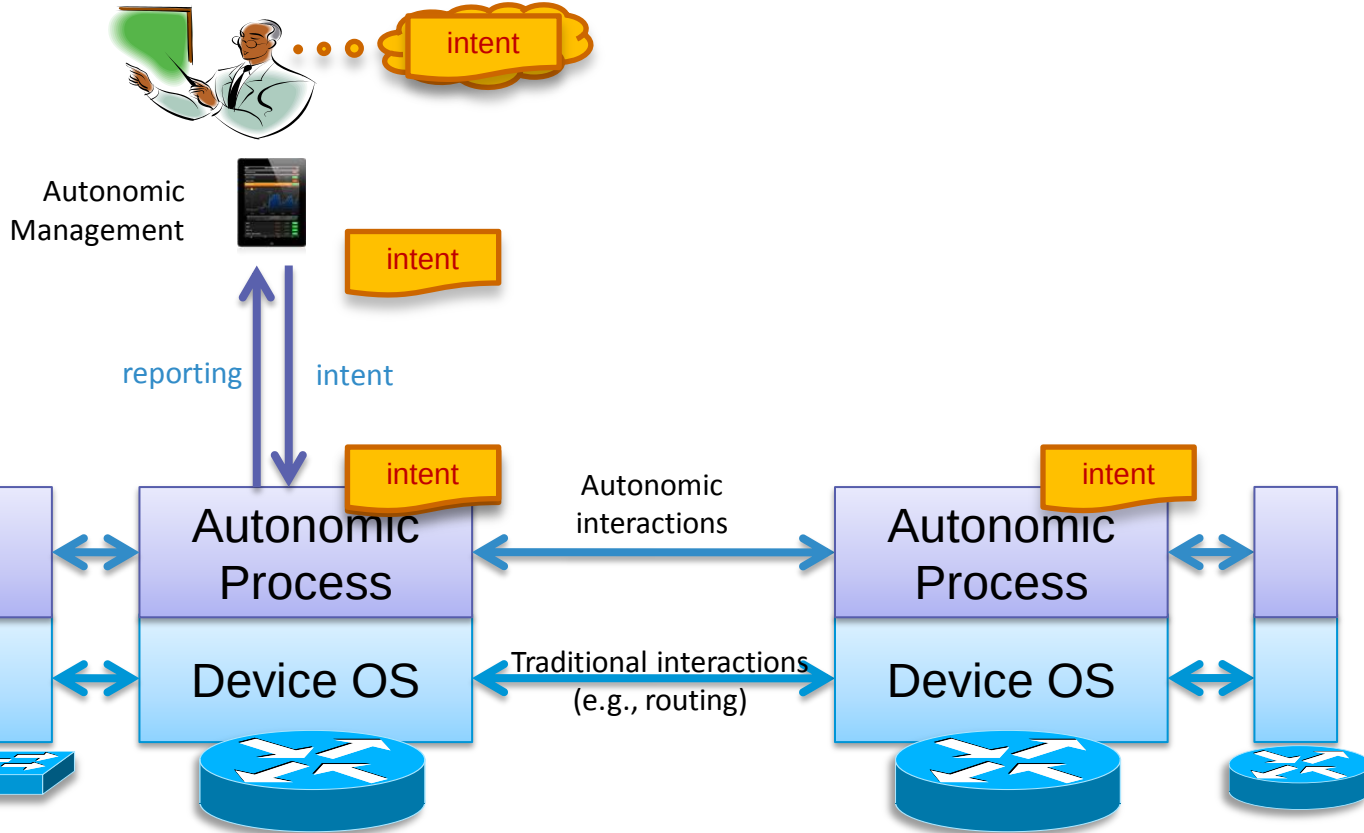
o and/or its affiliates. All rights reserved.

# The Autonomic Network

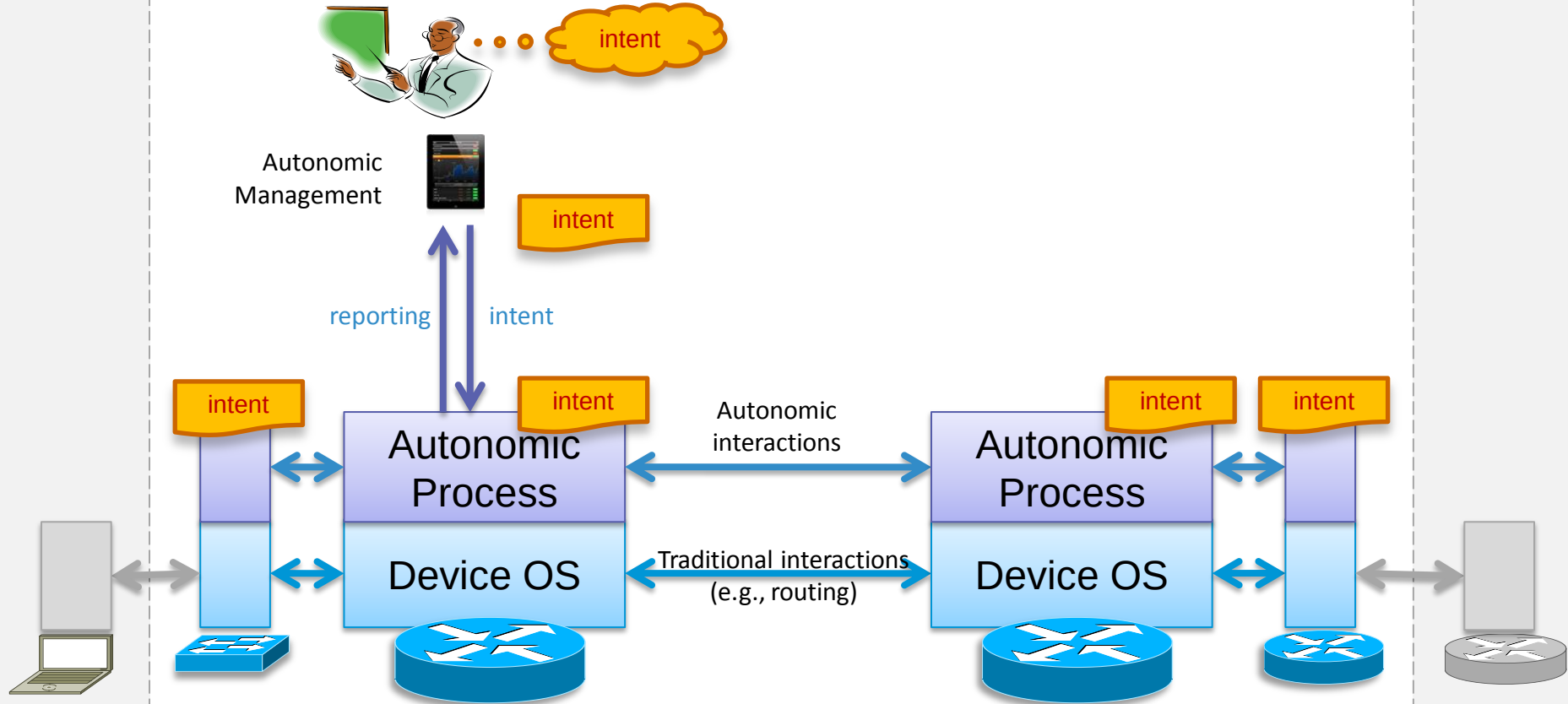


o and/or its affiliates. All rights reserved.

# The Autonomic Network



# The Autonomic Network



# SDN/ONE ...

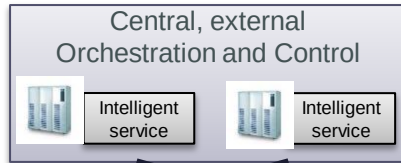
Business/Customer  
Controller  
(higher level orchestration)



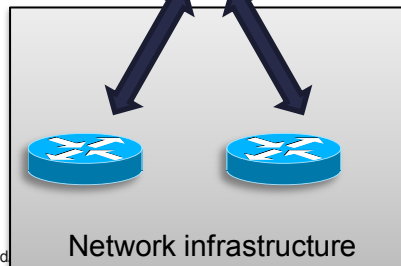
Designer/ Operator



Business level abstraction



Per-device-config, Eg: onePK



*Eliminate per-device operator/designer  
Configuration/operations of network  
Infrastructure devices*

## Openflow (class 0):

- No network embedded intelligence.
- Orchestration limited to narrow set of network services ("forwarding")

## ONE:

- Rich set of network services via onePK
- Optional network embedding of orchestration and control.
- This is what Autonomic Networking intends to leverage and expand upon...

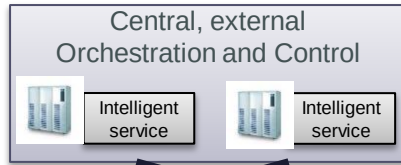
# SDN/ONE ...

Business/Customer  
Controller  
(higher level orchestration)

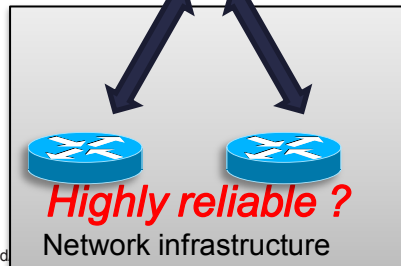


Designer/ Operator

Business level abstraction



Per-device-config, Eg: onePK



## Reliability:

- Network must be reliable
- (mis) Orchestration creates new sources for failures
  - Eg: orchestration interrupts connectivity between network devices and orchestrator)

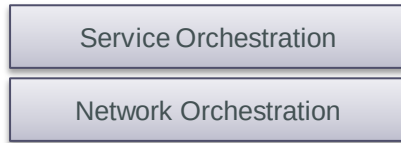
# SDN/ONE ...

Business/Customer  
Controller  
(higher level orchestration)

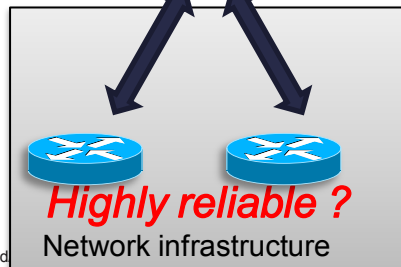


Designer/ Operator

Business level abstraction



Per-device-config, Eg: OnePK



## Service vs. Network orchestration:

- Service Orchestration is concerned about mapping Business logic into instance/policies of Network features
- Network orchestration configures network wide consistent network features.

# ONE + Autonomic Networking

Business/Customer  
Controller  
(higher level orchestration)

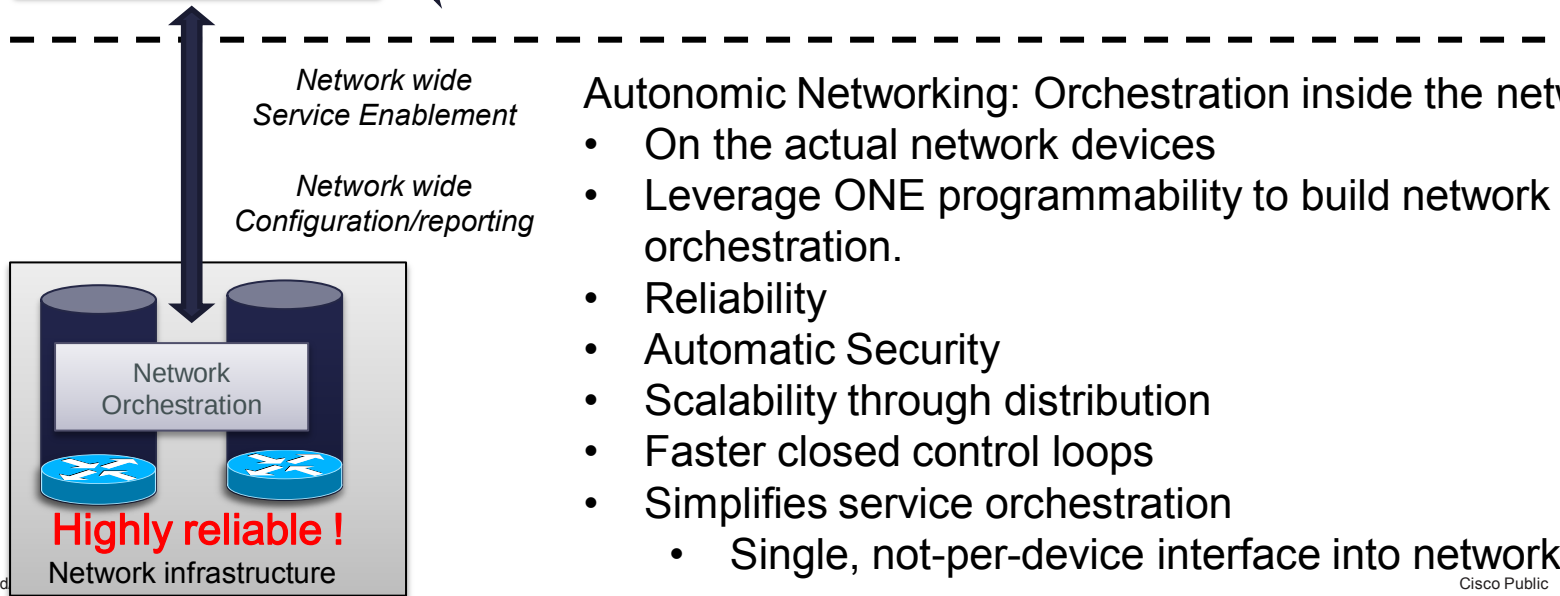


Business level abstraction

Service Orchestration



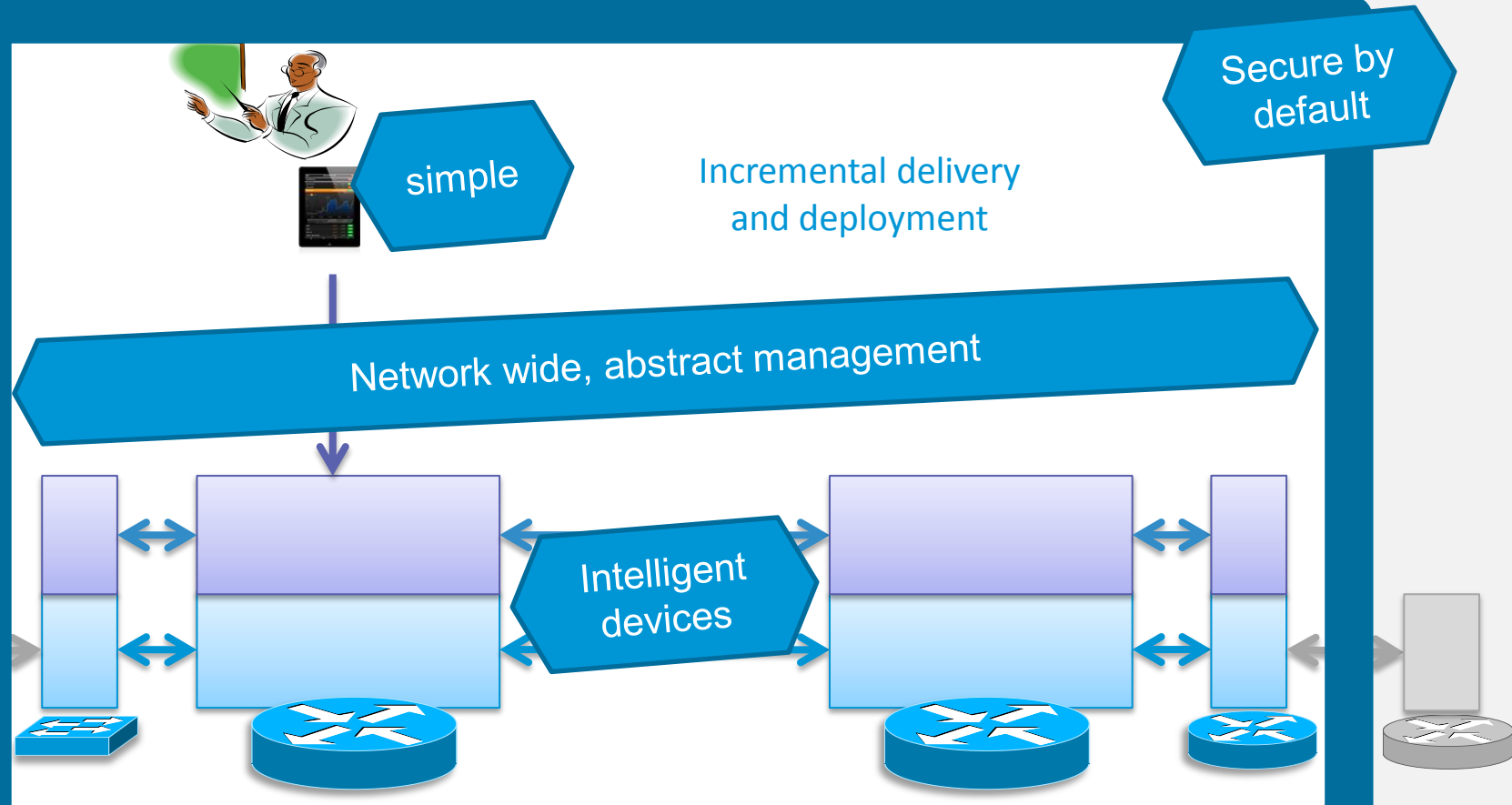
Designer/ Operator



Autonomic Networking: Orchestration inside the network

- On the actual network devices
- Leverage ONE programmability to build network orchestration.
- Reliability
- Automatic Security
- Scalability through distribution
- Faster closed control loops
- Simplifies service orchestration
  - Single, not-per-device interface into network

# Autonomic Networking – Summary



# Agenda

- Welcome
- Industry Observations
- Network Programmability

Hands-on Lab ...

Coffee Break

.... Hands-on Lab

- Recent Industry Adoption
- Autonomic Networking
- **Resources and References**

# Resources and References



# References – Instrumentation and Automation

Device Manageability Instrumentation (DMI) [www.cisco.com/go/instrumentation](http://www.cisco.com/go/instrumentation)

- Embedded Event Manager (EEM): [www.cisco.com/go/eem](http://www.cisco.com/go/eem)
- Cisco Beyond – EEM Community: [www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
- Embedded Menu Manager (EMM): <http://tinyurl.com/emm-in-124t>
- Embedded Packet Capture (EPC): [www.cisco.com/go/epc](http://www.cisco.com/go/epc)
- Flexible NetFlow: [www.cisco.com/go/netflow](http://www.cisco.com/go/netflow) and [www.cisco.com/go/fnf](http://www.cisco.com/go/fnf)
- IPSLA (formerly SAA, formerly RTR): [www.cisco.com/go/ipsla](http://www.cisco.com/go/ipsla)
- Network Analysis Module: <http://www.cisco.com/go/nam>
- Network Based Application Recognition (NBAR): [www.cisco.com/go/nbar](http://www.cisco.com/go/nbar)
- Smart Call Home: [www.cisco.com/go/smartcall](http://www.cisco.com/go/smartcall)
- Cisco Configuration Engine (CCE): [www.cisco.com/go/ciscoce](http://www.cisco.com/go/ciscoce)
- Feature Navigator: [www.cisco.com/go/fn](http://www.cisco.com/go/fn)
- MIB Locator: [www.cisco.com/go/mibs](http://www.cisco.com/go/mibs)

# Autonomic Networking References

- [www.cisco.com/go/autonomic/](http://www.cisco.com/go/autonomic/)
- A Framework for Autonomic Networking  
<http://tools.ietf.org/html/draft-behringer-autonomic-network-framework>
- Autonomic Networking: Definitions and Design Goals  
<http://tools.ietf.org/html/draft-irtf-nmrg-autonomic-network-definitions>
- Making the Internet Secure by Default  
<http://tools.ietf.org/html/draft-behringer-default-secure>
- Bootstrapping Key Infrastructures  
<http://tools.ietf.org/html/draft-pritikin-bootstrapping-keyinfrastructures>
- Autonomic Networking Configuration Guide, Cisco IOS Release 15S  
[www.cisco.com/en/US/partner/docs/ios-xml/ios/auto\\_net/configuration/15-s/an-auto-net-15-s-book.html](http://www.cisco.com/en/US/partner/docs/ios-xml/ios/auto_net/configuration/15-s/an-auto-net-15-s-book.html)
- Cisco IOS Autonomic Networking Command Reference  
[www.cisco.com/en/US/partner/docs/ios-xml/ios/auto\\_net/command/an-cr-book.html](http://www.cisco.com/en/US/partner/docs/ios-xml/ios/auto_net/command/an-cr-book.html)
- **autonomic-team@cisco.com**

# Embedded Automation Systems (EASy)

For Your  
Reference

1. Browse and Download EASy Packages  
[www.cisco.com/go/easy](http://www.cisco.com/go/easy)
2. Make Sure to also download EASy Installer
3. Browse Other Embedded Automations  
[www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
4. Learn About The Technology Under The Hood  
[www.cisco.com/go/instrumentation](http://www.cisco.com/go/instrumentation)  
[www.cisco.com/go/eem](http://www.cisco.com/go/eem)  
[www.cisco.com/go/pec](http://www.cisco.com/go/pec)
5. Discuss, Ask Questions, Suggest Answers  
[supportforums.cisco.com](http://supportforums.cisco.com)  
[supportforums.cisco.mobi](http://supportforums.cisco.mobi)
6. Upload your own Examples to CiscoBeyond  
[www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
7. Engage via [ask-easy@cisco.com](mailto:ask-easy@cisco.com)



# References – Network Programming

- Cisco ONE Enterprise Network Architecture Overview Video: <http://youtu.be/kyKZ-rrLIAY>
- Cisco ONE – Open Network Environment: [www.cisco.com/go/one](http://www.cisco.com/go/one)
- Cisco onePK – ONE Platform Kit: [www.cisco.com/go/onepk](http://www.cisco.com/go/onepk)
- Cisco Developer Network – onePK: [developer.cisco.com/web/onepk](http://developer.cisco.com/web/onepk)
- Cisco Developer Network – XNC: [developer.cisco.com/web/xnc](http://developer.cisco.com/web/xnc)
- CiscoLive 2013 Partner Showcases: [www.cisco.com/web/solutions/netsys/CiscoLive/](http://www.cisco.com/web/solutions/netsys/CiscoLive/)
- Cisco ONE on Slideshare: <http://www.slideshare.net/getyourbuildon>
- Network Programming Hands-on Lab Partner Education Connection: [www.cisco.com/go/pec](http://www.cisco.com/go/pec)
- Cisco Cloud Connectors – Blog: <http://blogs.cisco.com/borderless/the-network-is-the-path-to-accelerate-adoption-of-cloud-services/>
- Cisco Cloud Connectors – Marketplace: [https://marketplace.cisco.com/catalog/search?search\[technology\\_category\\_ids\]=938](https://marketplace.cisco.com/catalog/search?search[technology_category_ids]=938)
- Cisco EASy – Embedded Automation Solutions: <http://www.cisco.com/go/easy>
- Cisco Scripting Community: [www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)

# References – Cisco onePK @ DevNet

- Discover onePK
- Use Case Examples
- Getting Started Instructions
- Installation Guides
- API Documentation for C and Java
- Developer Community
- Media Library, News and FAQ

© 2013

Join the Community at:  
<https://developer.cisco.com/site/tech/networking/sdn/onepk-developer/overview/>

# References – Cisco XNC @ DevNet

- Review the Basics
- Technical Overview
- Getting Started Instructions
- Installation Guides
- API Documentation for REST and Java northbound Interface
- Developer Community
- Media Library, News and FAQ

Welcome! [Log In](#) [Register](#)

[DevNet](#) [Partner Network](#)

Home > [Networking](#) > [Cisco ONE](#) > [XNC Developer](#) > [XNC](#)

[XNC Developer Overview](#) >

[DevNet Map](#) >

[Community Forums](#) >

[Sandbox](#) >

## Cisco Extensible Network Controller (XNC) Developer Center

Welcome to the new look Cisco Extensible Network Controller (XNC) Developer Center website ...

Software Defined Networking (SDN) is an approach to programmable network that separates and abstracts the control-plane functions from the network devices and places them in a centralized controller. To accommodate adoption of SDN in both existing (greenfield) and new (greenfield) networks, Cisco has taken a hybrid approach. In this approach, some traditional control-plane functions will continue to exist in the network devices for conventional network operations, and some will be extracted to a centralized external controller to accommodate SDN solutions. Cisco Extensible Network Controller (XNC) addresses these requirements and emerging trends by providing much greater automation and orchestration of the network fabric, and by allowing dynamic, application-driven configuration of network and devices.

### 1. Learn about Cisco XNC

Read about Cisco XNC and see how it can empower your network.



[Get Started with one Controller](#)

### 2. Technical Overview of XNC

The technical overview describes the basics: how to integrate with Cisco XNC, where it fits in the architecture, and how to



[See the Technical Overview](#)

### 3. Get Started with Cisco XNC

Get the software and read how to get started with Cisco XNC



[Get Started NOW](#)

### Developer Resources

Latest [Latest Cisco XNC Info](#) >

[Watch: Cisco Extensible Network Controller \(XNC\) Demo Highlights Dynamic Network Configuration](#)

[Q&A: DemoFriday™ Questions Answered From The Extensible Network Controller \(XNC\) Demo](#)

[Sign up for the Introducing Network Programmability Fundamentals class](#)

### Community

Join the Community

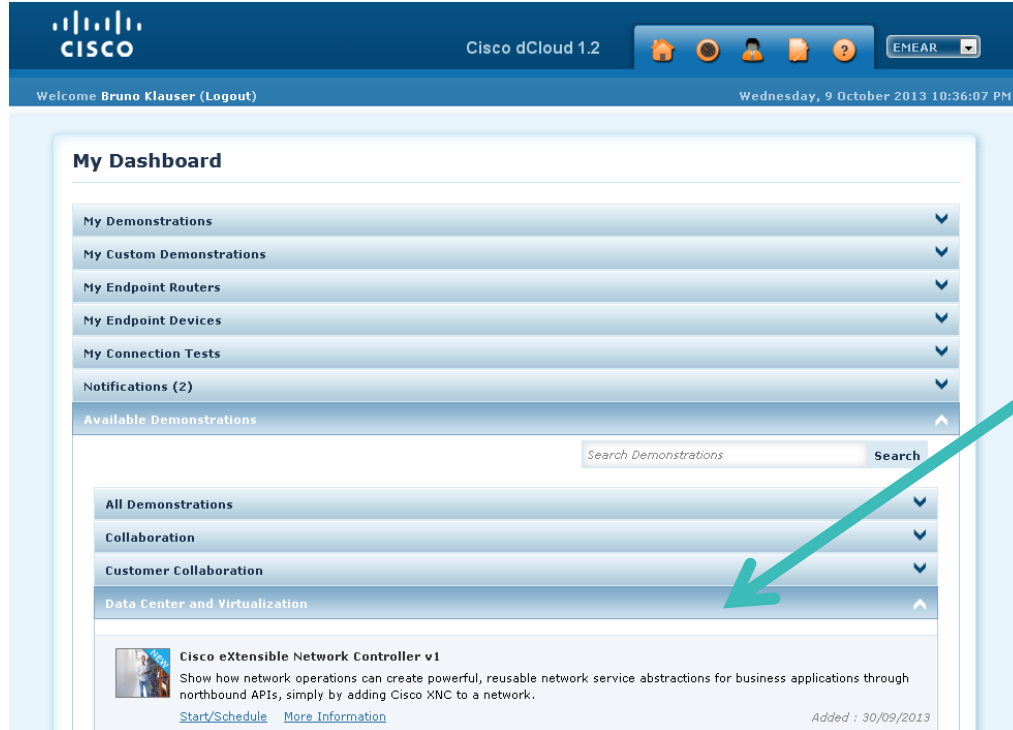
Join the Cisco XNC community to read forums and blogs. Connect with other Cisco XNC developers.

[Click to Join](#) >

Join the Community at:  
<https://developer.cisco.com/site/tech/networking/sdn/xnc/overview/>

# References – Cisco ONE @ dCloud

For Your  
Reference



- Cloud-based Demos
- Scheduled or on-demand
- <http://dcloud.cisco.com>

## XNC Demos available:

- Network Slicing
- Topology Independent Forwarding (TIF)
- Monitor Manager

# References – Cisco ONE @ dCloud

**Cisco eXtensible Network Controller v1** X

**Demo Use**

Please tell us how you're going to use the demonstration.

We also recommend testing connectivity and bandwidth before your demonstration. You can access the test from the "My Dashboard" page

Demonstration Use: **Briefing**

Delivery Location: **Cisco Office**

Delivery Country: **Netherlands**

Demo Theatre: **EMEAR**

Audience Level: **Other**

Audience Type: **TDM**

Number of Attendees: **1**

Customer Name: **PVT Lab bklauser**

Partner Name: **YOUR Company Name**

Account Manager ID:

Deal ID:

Revenue Impact:

**Cancel** **Back** **Schedule**

- Navigate to
  - Available Demos
  - Data Center and Virtualization
  - Cisco XNC
  - Make use of CTRL+/- and F11 in Firefox to maximize Interface

## Scheduling Info:

- Demo Use: Briefing
- Customer: MasterClass Lab
- Partner: Your Company Name
- Cisco Contact: bklauser

?





How will YOU  
■ Program The Network?

Thank you.



Authors: Michael Behringer, Patrick Charretour, Joe Clarke, Bruno Klauser, Jason Pfeifer,  
Cisco Systems {mbehring | pcharret | jclarke | bklauser | jpfeifer}@cisco.com