Network Programming in a Cisco Open Network Environment

Bruno Klauser
Consulting Engineer
BN EMEAR CTO Team
bklauser@cisco.com
Announcing: Cisco Open Network Environment

Industry’s Most Comprehensive Networking Portfolio
- Hardware + Software
- Physical + Virtual
- Network + Compute

Applications

OPEN NETWORK ENVIRONMENT

1. Platform APIs
   One Platform Kit (onePK)
   - Programmatic APIs for Network HW (IOS, IOS-XR, NX-OS)

2. Controllers And Agents
   SDN:
   - Controller SW (OpenFlow, onePK)
   - OpenFlow 1.x support

3. Virtual Overlays
   Open Clouds with Nexus 1000V
   - Multi-hypervisor
   - Multi-service
   - Multi-cloud
   - Openstack support
**What Is 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…”

Note: SDN is not mandatory for network programmability nor automation

Source: www.opennetworking.org

---

**What Is OpenFlow?**

Open protocol that specifies **interactions between de-coupled control and data planes**

Note: OF is not mandatory for SDN

Note: North-bound Controller APIs are vendor-specific

---

**What is OpenStack?**

**Opensource software** for building public and private Clouds; includes Compute (Nova), Networking (Quantum) and Storage (Swift) services.

Note: Applicable to SDN and non-SDN networks

Source: www.openstack.org

---

**What is Overlay Network?**

Overlay network is created on existing network infrastructure (physical and/or virtual) using a network protocol. Examples of overlay network protocol are: GRE, VPLS, OTV, LISP and VXLAN

Note: Applicable to SDN and non-SDN networks
“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”)
A Brief History of Network Automation & Programming
Self-Service for Human Users

Wireless Controller(s) (WLC, etc, …)

ICT Operations

Business Operations

Operations (FCAPS)

Survivability → Manageability → Automation

Virtual / Overlay Networks

Services (Location, Guestnet, Onboarding, …)

Resource Allocation

1 3 5 6 7

Self-Service for Human Users

Cisco Connect

6
Inflection: Network Programming

Business Operations

What if the ‘User’ is a Software App?

Virtual / Overlay Networks

Network
Survivability → Manageability → Automation → Autonomy

ICT Operations

APIs and Agents
Domain Controllers

1 2 3 4 5 6 7
Operational Network Automation

Business Operations

Value to Business

Type I
Automate Existing Task

Benefits:
- OPEX ↓
- Quality ↑

Type II
Automate New Task

Benefits:
- OPEX ↓
- CAPEX ↓
- Quality ↑
- Reactive → Proactive

Maturity and Innovation

Network
Survivability → Manageability → Automation

ICT Operations

Domain Controllers
APIs and Agents

1

2

3

a

b
c
Operational Network Automation – I

- **Business Operations**
  - Type I Example: Embedded Automation Systems
  - Custom Notifications
  - Custom Failover
  - Custom Smart Ports

**Network**
Survivability ➔ Manageability ➔ Automation

Operational Network Automation – II

Type II Example: Preventive and Proactive Automations

Real-Life → Network Workflows

Smart Call Home
Proactive Maintenance

see: [www.cisco.com/go/smartcall](http://www.cisco.com/go/smartcall)

Network
Survivability → Manageability → Automation

From
- Late Surprises
- Multiple Manual Escalation Steps
- Iterative Problem Isolation
- Phone, Email based Data Exchange

To
- Early Warnings
- Automated Flow
- Pinpoint Detailed Events
- Reporting and Exports
## Network Automation and Manageability

**Fault**
- IP OAM—Ping, Trace, BFD, ISG per session
- 802.3ah—Link monitoring and remote fault indication
- 802.1ag—Continuity check, L2 ping, trace, AIS
- MPLS OAM—LSP ping, LSP trace, VCCV
- EEM—Embedded Event Manager
- EVENT-MIB—OID-based triggers, events, or SNMP Set, IETF DISMON
- EXPRESSION-MIB—OID expression-based triggers, IETF DISMON
- ...

**Configuration**
- Config CLI—diff, logging, lock, replace, rollback
- E-LMI—parameter and status signaling
- E-DI—Enhanced Device Interface, CLI, Perl, IETF Netconf
- EMM—Embedded Menu Manager
- NETCONF—IETF NETCONF XML PI
- CNS and WSMA
- TR-069
- KRON—command scheduler
- AutoInstall—bootstrapping
- IOS.sh—IOS Shell
- SmartInstall
- Auto SmartPorts

**Performance**
- Auto IP SLA—delay, jitter, loss probability
- CBQoS MIB—class-based QoS
- NBAR
- RMON
- EPC—Embedded Packet Capture
- ERM—Embedded Resource Manager
- GOLD—Generic Online Diagnosis
- Smart Call Home—preventive maintenance
- VidMon—Video Monitoring
- ...

**Accounting**
- Flexible NetFlow—IETF IPFIX
- BGP policy accounting includes AS information
- Periodic MIB bulk data collection and transfer
- ...

**Security**
- Auto Secure—one-touch device hardening
- LDP Auth—message authentication
- Routing Auth—MD5 authentication, BGP, OSPF
- ...

---

Cisco IOS® Device Manageability Instrumentation Has Evolved Significantly

[Diagram showing network automation and manageability with various elements like DC, Headquarters, and other network components.]

---
Network Automation Life-Cycle Span

- **Pre-Operational Era**
  - Is it built to Specification?
  - Does it meet Requirements?
  - Is there room for yet another service?

- **Operational Era**
  - How to configure?
  - Is it working as specified?
  - What if something goes wrong?
  - Are we meeting SLA?

- **Post-Operational Era**
  - How to take out of service?

Network Automation spans across full Life Cycle
Trigger a Workflow in the Network

**Problem:** Sometimes we want to receive remote information on a Router / Switch and be able to react to it locally – for example a notification from a UPS System.

**Solution:** Use Network Automation based on Cisco IOS Embedded Event Manager leveraging the EEM SNMP Notification Event Detector

- Router / Switch can receive SNMP Notifications
- Execute (trigger) EEM Policy to take local action
- Policy can query varbind info
- Supports Incoming or Outgoing Notifications
- Outgoing only for locally generated Notifications

```
Router(config)# event manager applet catch-a-trap
router(config-applet)# description test snmp notification unmanaged service
router(config-applet)# event snmp-notification oid 1.3.6.1.6.3.1.1.4.1.0
               oid-val "1.3.6.1.6.3.1.1.5.3" op eq src-ip-address 10.51.89.176
               direction incoming
router(config-applet)# action 010 ...
router(config-applet)# action 020 ...
```
**Trigger a Workflow from the Network**

**Problem:** A new rogue WLAN device in sensitive areas should be detected by Cisco CleanAir and automatically focus/pan/zoom a security camera.

**Solution:** Use Network Automation based on Cisco IOS Embedded Event Manager to receive an SNMP Notification from WLC and trigger the Video Operations Manager via HTTP.

1. Rogue WLAN Device added
2. Rogue Device detected by CleanAir AP
3. WLC sends SNMP Notification
4. EEM triggers upon SNMP Notification
5. EEM notifies VSOM via HTTP
6. Security Camera Focus/Pan/Zoom
Collect and Share Remote Information – 1/2

**Problem:** How to actively gather and share information from a router and from a few devices behind the router – across organizational and technical borders?

**Solution 1:** Initiate a project to make use of SNMP, Syslog, Event Management Software, Reporting, Provisioning and CRM Systems ...

**Solution 2:** Use Cisco IOS Network Automation to collect and post the information

Using Cisco IOS Embedded Event Manager and Tcl:

1. Import the http package into EEM policy
   ```tcl
   namespace import ::http::*
   ```

2. Collect the information required

3. Build a query for the http POST operation
   ```tcl
   set my_query [::http::formatQuery "status" $my_info]
   ```

4. POST the information to a website
   ```tcl
   set my_reply [::http::geturl $my_server_url -query $my_query]
   ```
Collect and Share Remote Information – 2/2

See: http://twitter.com/EASyDMI

Note: it is NOT recommended to use a public site or feed other than for demo purpose
Embedded Event Manager (EEM)

**Event Detectors**
- Syslog ED
- SNMP EDs
- Timer EDs
- none ED
- HW EDs
- Watchdog ED
- Interface ED
- XML RPC ED
- CLI ED
- OIR ED
- ERM ED
- EOT ED
- RF ED
- GOLD ED
- NetFlow ED
- IPSLA ED
- Route ED
- CDP LLDP ED
- 802.1x ED
- MAC ED

**Actions**
- EEM Applets
  - multi-event-correlation

**Embedded Event Manager**
# Embedded Event Manager – Applet Evolutions

<table>
<thead>
<tr>
<th>EEM Version</th>
<th>Release</th>
<th>Applet Modifications</th>
<th>Peanut Gallery Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>12.0(26)S</td>
<td>2 Events: Syslog, SNMP</td>
<td>Initial Version</td>
</tr>
<tr>
<td></td>
<td>12.3(4)T</td>
<td>Actions: Log, CNS event, Reload, Switchover</td>
<td>Limited benefits</td>
</tr>
<tr>
<td>2.0</td>
<td>12.3(14)T1</td>
<td>No structure changes</td>
<td>Many new actions</td>
</tr>
<tr>
<td>2.1</td>
<td>12.2(18)SXF5</td>
<td>Various New event detectors</td>
<td>Popular CLI / mail</td>
</tr>
<tr>
<td>2.2</td>
<td>12.4M</td>
<td>New actions: cli, info, mail, policy, SNMP trap, Modify counters, Publish application events, Read/set tracked objects</td>
<td>Actions run linear</td>
</tr>
<tr>
<td>2.3</td>
<td>12.4(11)T</td>
<td>maxrun support</td>
<td>Maxrun support == security</td>
</tr>
<tr>
<td></td>
<td></td>
<td>pattern parameter for CLI actions</td>
<td>Can handle CLI prompts</td>
</tr>
<tr>
<td>2.4</td>
<td>12.4(20)T</td>
<td>multi-event support</td>
<td>Boolean correlation of events within applet.</td>
</tr>
<tr>
<td>3.0</td>
<td>12.4(22)T</td>
<td>Redesign of action mechanism</td>
<td>Applets now rock!</td>
</tr>
<tr>
<td></td>
<td>12.2(33)SE</td>
<td>Program counter added</td>
<td>Programming language feel.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Loops, conditionals, regexps, context save, error handling</td>
<td></td>
</tr>
<tr>
<td>3.1/3.2</td>
<td>-</td>
<td>No changes</td>
<td>-</td>
</tr>
<tr>
<td>4.0</td>
<td>15.2(2)T</td>
<td>Applet file actions</td>
<td>File manipulation handy!</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TLS/SSL support for SMTP actions</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Custom port for SMTP actions</td>
<td></td>
</tr>
</tbody>
</table>
Example: EEM Applets – Loops, Variables

**Problem:** None in Particular

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

```plaintext
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."
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."
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."
!
alias exec sing event manager run 99-bob
```

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.

![Diagram showing packaging process]

- EASy Installer
- 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](http://www.cisco.com/go/easy)
Embedded Automation Systems (EASy)

1. Browse and Download EASy Packages
   www.cisco.com/go/easy

2. Make Sure to also download EASy Installer

3. Browse Other Embedded Automations
   www.cisco.com/go/ciscobeyond

4. Learn About The Technology Under The Hood
   www.cisco.com/go/instrumentation
   www.cisco.com/go/eem
   www.cisco.com/go/pec

5. Discuss, Ask Questions, Suggest Answers
   supportforums.cisco.com
   supportforums.cisco.mobi

6. Upload your own Examples to CiscoBeyond
   www.cisco.com/go/ciscobeyond

7. Engage via ask-easy@cisco.com
Inflection: Business-Driven Network Automations …

Value to Business

Type I
Automate Existing Task
Benefits:
- OPEX ↓
- Quality ↑

Type II
Automate New Task
Benefits:
- OPEX ↓
- CAPEX ↓
- Quality ↑
- Reactive → Proactive

Type III – Automation as Integral Part of Solution Design
Benefits:
- Revenue enabler
- OPEX ↓
- CAPEX ↓
- Quality ↑
- Reactive → Proactive
- Corporate Learning enabler

Network Survivability → Manageability → Automation
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
What are Network Programming and SDN?

SDN Enables
Business Applications to program (parts of) the Network

SDN is Relevant when (in any combination)
The Business is Software Centric
The Business Demands Automation
Network Operations is Highly Mature

SDN Adoption across Markets and Technology Domains
2007 : Prosumer/SMB Market Space
2009 : ESMB Routing & Switching Customizations
2010 : Network Compliance and Interop Testing
2012 : Software-Defined Data-Center SDDC
2013 : SP Network Function Virtualization, Connected Industries, Cloud-Intelligent Networks, …
“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”)
Who is the Network Programmer – 1/2

Currently Three Main Groups:

- **Network “Ponytails”**
- **System “Square Eyes”**
- **New Breed “Generation Y”**
### Who is the Network Programmer – 2/2

<table>
<thead>
<tr>
<th>Works for</th>
<th>Network “Ponytails”</th>
<th>System “Square Eyes”</th>
<th>Generation Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typical Employers</td>
<td>Vendor, Partner, large Customers</td>
<td>ISV/SI/Dev Partner, large Customer</td>
<td>Dev/SI/ISV Partner, large Customer, Education, Academia, NREN</td>
</tr>
<tr>
<td>Atypical Employers</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>Network, IOS Skills</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>Scripting Skills</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>Programming Skills</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>ICT Architectural Comp</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>SW Engineering Comp</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>3rd/System Domain Comp</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
<td>[names omitted]</td>
</tr>
<tr>
<td>Typical Tools</td>
<td>PuTTY, perl, expect, Visio, mySQL, LAMP, email, wiki</td>
<td>Eclypse, JMS, TiBCO, make, perl, C, Java, Oracle, TOAD, email, wiki</td>
<td>gedit, REST, xmpp, rss, XML/XSD, IM, social, blogs, play, redmine</td>
</tr>
</tbody>
</table>
Cisco Open Network Environment
Describing Software Architectures

4+1 View Model

Scenarios: Use Case, Who, What and Why
Examples: Routing-for-(Dollars), Application-Flow-Manipulation, Network Slicing, SDDC Provisioning, CI/CD, …

Development View: SDKs, Packages, Libraries, Tools
Examples: onePK SDK, binaries, .so, Eclipse, Debugger, IOL, …

Logical View: Features, Functions, Classes, Abstractions
Examples: onePK APIs in C, Java, Application APIs, …

Process View: Architectures, Processes, Interactions, Objects
Examples: Agents, Controllers, Thrift IDL, Cloud Connectors, Sentinels, …

Physical View: Deployment, Hosting, Topology, Connectivity
Examples: Virtual Containers, Blades, Endpoints, L1-2-3, Overlay

See: https://en.wikipedia.org/wiki/4+1_Architectural_View_Model
Cisco Open Network Environment – ONE

Preserve What is Working
- Resilience, Scale, Security
- Functionality and Rich Features
- Instrumentation

Evolve for New Requirements
- Operational Simplicity and Automations
- Programmability and Network-Awareness
- Upcoming Innovations

Open and Integrated Framework
- Software Defined Network concepts are a component of the Open Network Environment
- Existing APIs, Agents, Controllers and Infrastructure contribute

Open Network Environment

Development View
- Logical View

Process View
- Network Programming
- (Software) Architectures and Patterns

Physical View
- Deployment and Virtualization

Scenarios and Motivations
## Cisco Open Network Environment – ONE

### Preserve What is Working
- Resilience, Scale, Security
- Functionality and Rich Features
- Instrumentation

### Evolve for New Requirements
- Operational Simplicity and Automations
- Programmability and Network-Awareness
- Upcoming Innovations

### Open and Integrated Framework
- Software Defined Network concepts are a component of the Open Network Environment
- Existing APIs, Agents, Controllers and Infrastructure contribute

### Open Network Environment

<table>
<thead>
<tr>
<th>Network Programming</th>
<th>(Software) Architectures and Patterns</th>
<th>Deployment and Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>onePK</td>
<td>Controllers (ONE/Openflow PoC)</td>
<td>Nexus 1000v</td>
</tr>
<tr>
<td></td>
<td>(SBC, WLC, +++)</td>
<td>CSR 1000v</td>
</tr>
<tr>
<td>developer.cisco.com</td>
<td>CDN, Training, Certification, Partners, EEM, EASY</td>
<td>VSG and vFW/ASA, vWAAS, vNAM, ...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cisco Openstack Ed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blade Hosting (UCS-E, ...), Virtual Containers (AirVision, Cat, ISR, ASR, ...)</td>
</tr>
</tbody>
</table>

### Scenarios and Motivations
Preserve What is Working

- Resilience, Scale, Security
- Functionality and Rich Features
- Instrumentation

Evolve for New Requirements

- Operational Simplicity and Automations
- Programmability and Network-Awareness
- Upcoming Innovations

Open and Integrated Framework

- Software Defined Network concepts are a component of the Open Network Environment
- Existing APIs, Agents, Controllers and Infrastructure contribute

---

**Cisco Open Network Environment – ONE**

### Open Network Environment

<table>
<thead>
<tr>
<th>Network Programming</th>
<th>(Software) Architectures and Patterns</th>
<th>Deployment and Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>onePK</td>
<td>Controllers (ONE/Openflow PoC) (SBC, WLC, +++)</td>
<td>Nexus 1000v CSR 1000v VSG and vFW/ASA, vWAAS, vNAM, …</td>
</tr>
<tr>
<td>developer.cisco.com, CDN, Training, Certification, Partners, EEM, EASy</td>
<td>CIN, CloudConnect, Sentinels, Agents</td>
<td>Cisco Openstack Ed Blade Hosting (UCS-E, …), Virtual Containers (AirVision, Cat, ISR, ASR, …)</td>
</tr>
</tbody>
</table>

---

**Scenarios and Motivations**
Cisco ONE – Virtual Networking / Cloud Networking

- Nexus 1000V
  - Distributed switch
  - NX-OS consistency
- VSG
  - VM-level controls
  - Zone-based FW
- ASA 1000V
  - Edge firewall, VPN
  - Protocol Inspection
- vWAAS
  - WAN optimization
  - Application traffic
- CSR 1000V (Cloud Router)
  - WAN L3 gateway
  - Routing and VPN
- Ecosystem Services
  - Citrix NetScaler VPX virtual ADC
  - Imperva Web App. Firewall

6000+ Customers

Shipping

Shipping

Shipping

Limited Availability: CQ4’12
Full Availability: CQ1’13

2013
## Cisco Open Network Environment – ONE

### Preserve What is Working
- Resilience, Scale, Security
- Functionality and Rich Features
- Instrumentation

### Evolve for New Requirements
- Operational Simplicity and Automations
- Programmability and Network-Awareness
- Upcoming Innovations

### Open and Integrated Framework
- Software Defined Network concepts are a component of the Open Network Environment
- Existing APIs, Agents, Controllers and Infrastructure contribute

---

### Open Network Environment

<table>
<thead>
<tr>
<th>Network Programming</th>
<th>(Software) Architectures and Patterns</th>
<th>Deployment and Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>onePK</td>
<td>Controllers (ONE/Openflow PoC) (SBC, WLC, +++)</td>
<td>Nexus 1000v CSR 1000v VSG and vFW/ASA, vWAAS, vNAM, …</td>
</tr>
<tr>
<td>developer.cisco.com, CDN, Training, Certification, Partners, EEM, EASy</td>
<td>CIN, CloudConnect, Sentinels, Agents</td>
<td>Cisco Openstack Ed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenarios and Motivations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blade Hosting (UCS-E, …), Virtual Containers (AirVision, Cat, ISR, ASR, …)</td>
</tr>
</tbody>
</table>
Example: Controller Architectures

Open Daylight Controller Provides

Open-Source, Community-Driven Architecture and Platform for SDN Applications

Cisco are

Co-Founder
Platinum Member
Active Steering and Technical Member

Initial Controller Download available from 2013-04-08

http://www.opendaylight.org
http://wiki.opendaylight.org
Example: Controller Architectures

Cisco ONE Controller Provides

Platform for generic control functions – state consolidation across multiple entities

Current Use Cases

Flexible Network Partitioning and Provisioning (“Campus Slicing”)
Network Troubleshooting
Custom Routing

Software Product, Java, Beta: 1H CY13

Cisco ONE Controller Anatomy

onePK and Openflow support
Modular Architectures
Extensibility and Manageability (UI and API)
Example: Flexible Monitoring Matrix Solution – 1/2

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

**Challenge:** Traditional implementations tend to be inflexible, closed, expensive

- **Specialized high-density, high-throughput Switch**
- **Static rules and configurations**
- **Closed systems, limited adaptability, tool compatibility**
- **Expensive, not scalable**

Diagram:
- **Static Configuration**
- **Matrix Switch**
- **Analyzer**
- **Tool Ports**
- **IDS-Green**
- **Network Ports**
- **Production Network**
- **Public Internet**

**Example:**

- Public Internet
- Tool Ports
- Analyzer
- IDS-Green
- Network Ports
- Matrix Switch
- Static Configuration
- Production Network
- SPAN

Note: The diagram illustrates the components and connections of the monitoring matrix solution, emphasizing the static configuration and its limitations compared to a flexible, programmable solution.
Example: Flexible Monitoring Matrix Solution – 2/2

Problem: How to build a flexible, programmable and cost-effective monitoring matrix solution?

Solution: Use Network Programming based on Cisco ONE Controller:

- Agent enabled Nexus 3000 Switches (Openflow)
- Cisco ONE Controller and Matrix Application
- Open UI (Controller and Application) and API (Agent and Controller)
- Cost-effective, scalable, extensible
Example: Cloud Connectors for ISR G2 and beyond

Cloud Connectors Provide
- Network-Awareness to Cloud Services
- Cloud Service-Awareness to Network
- Improved Quality and Experience
- Simplified Deployment and Operations

Cloud Connector Anatomy
- Deployed into Branch on ASR/ISR
- Native (in Network OS) or Hosted (on SRE, UCS-E Blade)
- Abstractions on top of Network OS

Some Examples

Available Now
- Scansafe Connector
- HCS Connector
- Webex Cloud Connect Audio

Future
- Backup/Storage Connector
- Identity Services Connector
- Securelogix / UC Services Connector
- VXI Connector
Cisco Cloud Intelligent Network
Delivering Optimal Experience, Pervasive Security, and Simplified Operations

Management and Policy

Cloud Connectors
- Collaboration Survivability
- Web Security
- Cloud Storage
- 3rd Party

Network Services for Cloud
- Visibility
- Optimization
- Security
- Collaboration
- App Hosting

Cloud Intelligent Platforms
- ISR G2
- ASR 1K
- CSR 1KV

Branch Office
Campus / Data Center
Cloud

Private/Public/Hybrid

Users

Cloud Services
**Cloud Connectors**

Bringing Network Intelligence to the Cloud

Cloud Connector – a network service that improves the performance, security or availability of cloud applications. Cisco Cloud Connectors provide Optimal Experience, Pervasive Security, and Simplified Operations when utilizing Private, Public or Hybrid Clouds over the WAN or Internet.

Cloud Intelligent Platforms

- Visibility
- Optimization
- Security
- Collaborations
- App Hosting

Users

Cloud Services

Private/Public/Hybrid
Types of Cloud Connectors

• **Native** Connectors
  - Hosted Collaboration Solution (ESRST)
  - WebEx Cloud Connected Audio (CUBE)
  - ScanSafe Connector
  
  Available Now

• **Hosted** Connectors
  - Run on UCS-E and use OnePK, UC Services or No API
  - May not always run everything on UCS-E

  Available Now
  - OnePK and UC Services (Pi20 July 2012)

• **Scripted** Connectors
  - Uses OnePK API, Python scripting environment
  - Rapid Development and Deployment
  - Takes advantage of Cisco infrastructure (hosting environment to run)

  Development environment by Pi22 (H2 CY13)

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
Cisco Open Network Environment – ONE

Preserve What is Working
- Resilience, Scale, Security
- Functionality and Rich Features
- Instrumentation

Evolve for New Requirements
- Operational Simplicity and Automations
- Programmability and Network-Awareness
- Upcoming Innovations

Open and Integrated Framework
- Software Defined Network concepts are a component of the Open Network Environment
- Existing APIs, Agents, Controllers and Infrastructure contribute

Open Network Environment

<table>
<thead>
<tr>
<th>Network Programming</th>
<th>(Software) Architectures and Patterns</th>
<th>Deployment and Virtualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>onePK</td>
<td>Controllers (ONE/Openflow PoC)</td>
<td>Nexus 1000v</td>
</tr>
<tr>
<td>developer.cisco.com, CDN, Training, Certification, Partners, EEM, EASY</td>
<td>(SBC, WLC, +++)</td>
<td>CSR 1000v</td>
</tr>
<tr>
<td></td>
<td>CIN, CloudConnect, Sentinels, Agents</td>
<td>VSG and vFW/ASA, vWAAS, vNAM, …</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blade Hosting (UCS-E, …), Virtual Containers (AirVision, Cat, ISR, ASR, …)</td>
</tr>
</tbody>
</table>

Scenarios and Motivations
Cisco ONE Platform Kit (onePK)

Network Programming Environment to:
- Innovate
- Extend
- Automate
- Customize
- Enhance
- Modify

YOUR Applications

Any Cisco Router or Switch

C, JAVA Program

API Presentation

API Infrastructure

Catalyst  Nexus  ASR ISR

onePK IPC Channel
Cisco ONE Platform Kit (onePK)

onePK Provides

- Abstractions (Service Sets)
- Programmatic Interfaces (C, Java, (REST) …)
- Software Development Kit (SDK)

Anatomy of a onePK Application

- Software Application (currently C and Java)
- Interfaces and Abstractions (Service Sets)
- Communication Bus (Thrift IDL)
- Connected-Apps Agent in Network OS
- Network OS Features and Embedded Automations

---

<table>
<thead>
<tr>
<th>Service Set</th>
<th>Description</th>
<th>Version v.6.0.5 EFT 2012/2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Path</td>
<td>Provides packet delivery service to application: Copy, Punt, Inject</td>
<td></td>
</tr>
<tr>
<td>Policy</td>
<td>Provides filtering (NBAR, ACL), classification (Class-maps, Policy-maps), actions (Marking, Policing, Queuing, Copy, Punt) and applying policies to interfaces on network elements</td>
<td></td>
</tr>
<tr>
<td>Routing</td>
<td>Read RIB routes, add/remove routes, receive RIB notifications</td>
<td></td>
</tr>
<tr>
<td>Element</td>
<td>Get element properties, CPU/memory statistics, network interfaces, element and interface events</td>
<td></td>
</tr>
<tr>
<td>Discovery</td>
<td>L3 topology and local service discovery</td>
<td></td>
</tr>
<tr>
<td>Utility</td>
<td>Syslog events notification, Path tracing capabilities (ingress/egress and interface stats, next-hop info, etc.)</td>
<td></td>
</tr>
<tr>
<td>Developer</td>
<td>Debug capability, CLI extension which allows application to extend/integrate application’s CLIs with network element</td>
<td></td>
</tr>
</tbody>
</table>
Network Programming
Cisco ONE Platform Kit (onePK)
YOUR Applications

- **System**
  - CPU, Memory, Platform, Serial #, Versions, Uptime, Location, OIR, CLI Changes

- **Interfaces**
  - Port, Slot, BW, MTU, TX/RX, BPS, PPS, Errors, Other Stats, Config, Link Changes

- **Discovery**
  - CDP, Topology Graph, Edges, Nodes, Topology Changes
YOUR Applications

Element
- System
- Interfaces
- Discovery

Location

IP address, MTU, Clear Stats, Shut/No Shut

Filters
Example: Connect and Get Properties via onePK (C)

```c
char *str = NULL;
onep_element_connect(elemA, user, pwd, NULL, &sh);
onep_element_get_property(elemA, &property);
if (property) {
    onep_element_to_string(elemA, &str);
    if (str) {
        fprintf(stderr, "%Element Info: %s\n", str);
        free(str);
    }
}
```
Monitor Memory Usage

**Problem:** What if we need to dynamically investigate further upon a resource symptom?

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

```
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 "Warning: proc memory spike"
```
A Network “Top”

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
**Problem:** Check for known root causes and misconfigurations along a multi-platform path from a custom graphical application

**Solution:** Use onePK

1. Network begins with mismatched parameters on either side of link (e.g. MTU)
2. Application checks parameters on either side and identifies mismatches (red lines)
3. Application sets parameters to match (lines turn green)
4. Application registers for events related to parameters change.
5. Users logs into console and manually changes parameter. Topology indicates change.
MTU Checking in Action ...

Device Info
- Hostname: ASR9k
- Address: 172.20.165.43
- Processor: MPC8641D
- Version: ASR-9006 AC Chassis
- Serial Number: FOX1548GNC1
- Neighbors: 2
- MTU:
  - Gi0/0/0/1:1500 to Eth1/1:1500 (proctype)
  - Gi0/0/0/0:1024 to Gi0/0/0/0:1514 (CRS) *
onePK Service Sets – Policy and Routing – 1/2

Routing
- RIB, Next-Hop, metric, AD, scope (VRF), Changes

QoS
- Configured Classes

Security
- Configured ACLs

YOUR Applications
onePK Service Sets – Policy and Routing – 2/2

Routing
QoS
Security

Static routes
Service-Policies (Police, Mark, Shape, Queue)
ACLs

YOUR Applications
Example: Get and Set Routes via onePK (Java)

- **Getting Routes**

```java
L3UnicastScope scope = new L3UnicastScope("", AFIType.IPV4, SAFIType.UNICAST, "");
NetworkPrefix prefix = new NetworkPrefix(InetAddress.getByName("0.0.0.0"), 0);
L3UnicastRIBFilter ribFilter = new L3UnicastRIBFilter(OwnerType.NONE, "NONE", prefix);
L3UnicastRouteRange range = new L3UnicastRouteRange(prefix, RouteRange.RangeType.EQUAL_OR_LARGER, 100);
List<TopoNode> mynodes = TopoNode.getAllNodes();
for(TopoNode thisnode : mynodes) {
    Routing routing = Routing.getInstance(thisnode.ne);
    RIB rib = routing.getRib();
    List<Route> routeList = rib.getRouteList(scope, ribFilter, range);
    for (Route route : routeList) {
        // Process route
    }
}
```

- **Setting Routes**

```java
L3UnicastRoute aRoute = new L3UnicastRoute(prefix, nextHopList);
aRoute.setAdminDistance(1);
RouteOperation op = new L3UnicastRouteOperation(RouteOperationType.ADD, aRoute);
List<RouteOperation> opList = new ArrayList<RouteOperation>();
opList.add(op);
AppRouteTable art = routing.getAppRouteTable();
art.updateRoutes(scope, opList);
```
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 / …

```plaintext
router ospf 1
redistribute application <app name> ...
```
Example: Routing for Dollars / CO₂ / Tulips /…

Statistics and Metrics

• **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.
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
onePK Service Sets – Data Path – 1/2

Data Plane

Copy or Punt Packets

YOUR Applications
onePK Service Sets – Data Path – 2/2
Example: Punt and Inject Packets via onePK (C)

TRY(rc, onep_dpss_register_for_packets(
    nel,
    dpss,
    targ_left,
    interesting_class,
    ONEP_DPSS_ACTION_PUNT,
    encrypt_callback,
    (void *)intf_left,
    &reg_handle), "Register for packets");
Custom Application Traffic Flow Handling – 1/2

**Problem:** We need to custom encrypt packets of a specific application traffic flow

**Solution:** Use onePK to punt, encrypt and reinject the relevant packets

1. Policy APIs on ingress router are set to punt telnet and syslog to app
2. App encrypts punted traffic and re-injects into data path.
3. Policy APIs on egress router punt telnet and syslog to app
4. App decrypts punted traffic and re-injects into data path.
5. Traffic that does not match policy passes through unencrypted.
Custom Application Traffic Flow Handling – 2/2

What Client Sees

```
client#telnet 10.13.1.1
Trying 10.13.1.1 ... Open

User Access Verification
Username: user1
Password: 
server>en
Password: 
server#show clock
*10:02:12.131 PST Mon Jul 2 2012
server#!Starting Application Now
server#show clock
*10:02:42.169 PST Mon Jul 2 2012
server#
```

What Wireshark Sees

```
Stream Content

User Access Verification
Username: ...!...!.....user1
Password: 
session
server#show clock
*10:02:12.131 PST Mon Jul 2 2012
server#!Starting Application Now
server#show clock
*10:02:42.169 PST Mon Jul 2 2012
server#%9^v^>9!v5:95=|gflfdlbdxg`ov...v.98v.#:vdvdfgd%3$ 3$u
```
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 Ka Band
5. Reliable, secure emergency response network saving ~4M€ operating cost annually
onePK Agent ↔ Application Interactions

Examples:
- Getting 50'000 ACLs from an Element
- Syslog Messages
- RIB Changes
- Setting 50'000 ACLs on an Element
- Get Element Version
- Set Interface Address

Examples:
- Getting 50'000 ACLs from an Element
- Syslog Messages
- RIB Changes
- Setting 50'000 ACLs on an Element
- Get Element Version
- Set Interface Address

Examples:
- Getting 50'000 ACLs from an Element
- Syslog Messages
- RIB Changes
- Setting 50'000 ACLs on an Element
- Get Element Version
- Set Interface Address

Examples:
- Getting 50'000 ACLs from an Element
- Syslog Messages
- RIB Changes
- Setting 50'000 ACLs on an Element
- Get Element Version
- Set Interface Address
What will YOU program?
Portfolio of API, Languages and Abstractions

Network Programming with onePK and Embedded Network Automation

- **Native Network OS Embedded Automation**
  - Event-/Expression- MIB, PfR, IPSLA Thresholds, Embedded Event Manager Applets, …

- **Advanced Network OS Embedded Scripting**
  - Tcl, Python, Embedded Event Manager, EASy, …

- **Structured API**
  - onePK C

- **Object Oriented API**
  - onePK Java

- **Higher-Level Abstractions / Interfaces**
  - onePK Libraries, REST, XMPP, Design Patterns, OMNI Controllers, …

Choice and Flexibility of Implementation

Network Programming – SDN

Network Automation – Embedded Automations
Cisco ONE – Current Key Components …

Business Operations

APIs and Agents

ONE Controller

Domain Controllers

Open Daylight Controller

Virtual / Overlay Networks

APIs and Agents

onePK

Virtual / Overlay Networks

Nexus 1000v
CSR 1000v
ASA 1000v
VSG
vWAAS
++

CloudConnectors

onePK

UCS-E

Virtual Containers

Network

Survivability → Manageability → Automation → Autonomy

For Your Reference

Openflow

Cisco ONE – Current Key Components …
Resources and References
References – Programmable and Cloud-Intelligent


Cisco Scripting Community: [www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)


Network Programming Hands-On Lab

At CiscoLive: Join us for TECNMS-3601 Advanced Network Automation or

1. Walk up to the WISP Labs
2. Book a seat for:
   - Network Automation – The Basics
   - Network Automation – Intermediate
   - Network Automation – Advanced
   - Network Automation – Smart Call Home
   - Network Programming – The Basics
   - Network Programming – Intermediate
   - Network Programming – Advanced

Partners via PEC (currently being updated):

2. Click on >Launch
3. Search for 'Network Programming'
4. Enjoy!
Master Class Network Programming and Automation
Sustainable Innovation and Differentiation with Cisco Embedded Automation and onePK

How can I successfully use Network Programming?
Step-by-step introduction and adoption strategies

What have other Customers and Partners done?
Real-Life Examples and Case Studies

Where do I start – can I do this?
Practical Examples and Hands-on Lab

What are Cisco ONE and onePK anyhow?
Illustration in the bigger SDN and Automation Context

Join us for the Network Programming Master Class Series
References – Instrumentation and Automation

**Device Manageability Instrumentation (DMI)** [www.cisco.com/go/instrumentation]
- Cisco Beyond – EEM Community: [www.cisco.com/go/ciscobeyond](http://www.cisco.com/go/ciscobeyond)
- 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)

- **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)
Embedded Automation Systems (EASy)

1. Browse and Download EASy Packages
   www.cisco.com/go/easy

2. Make Sure to also download EASy Installer

3. Browse Other Embedded Automations
   www.cisco.com/go/ciscobeyond

4. Learn About The Technology Under The Hood
   www.cisco.com/go/instrumentation
   www.cisco.com/go/eem
   www.cisco.com/go/pec

5. Discuss, Ask Questions, Suggest Answers
   supportforums.cisco.com
   supportforums.cisco.mobi

6. Upload your own Examples to CiscoBeyond
   www.cisco.com/go/ciscobeyond

7. Engage via ask-easy@cisco.com
Network Programming and Automation
Sustainable Innovation and Differentiation with Cisco Embedded Automation and onePK

Network Programming in a Cisco Open Networking Environment …
… provides Choice and Flexibility of
  - APIs and Abstractions
  - Architectures
  - Deployment Models
… closes the gap between Business Applications and Networks
… enables Operational Savings and New Opportunities
… puts YOU in control

What will YOU Program?
Thank you.