Device Manageability Instrumentation (DMI)

[20080616-20 – Nordics]

Bruno Klauser

Consulting Engineer NMS/OSS

European Markets

bklauser@cisco.com

wwwin-people.cisco.com/bklauser

Even Solberg

Product Manager

European Markets

esolberg@cisco.com

Introduction & Overview

Manageability and Self-* Networks

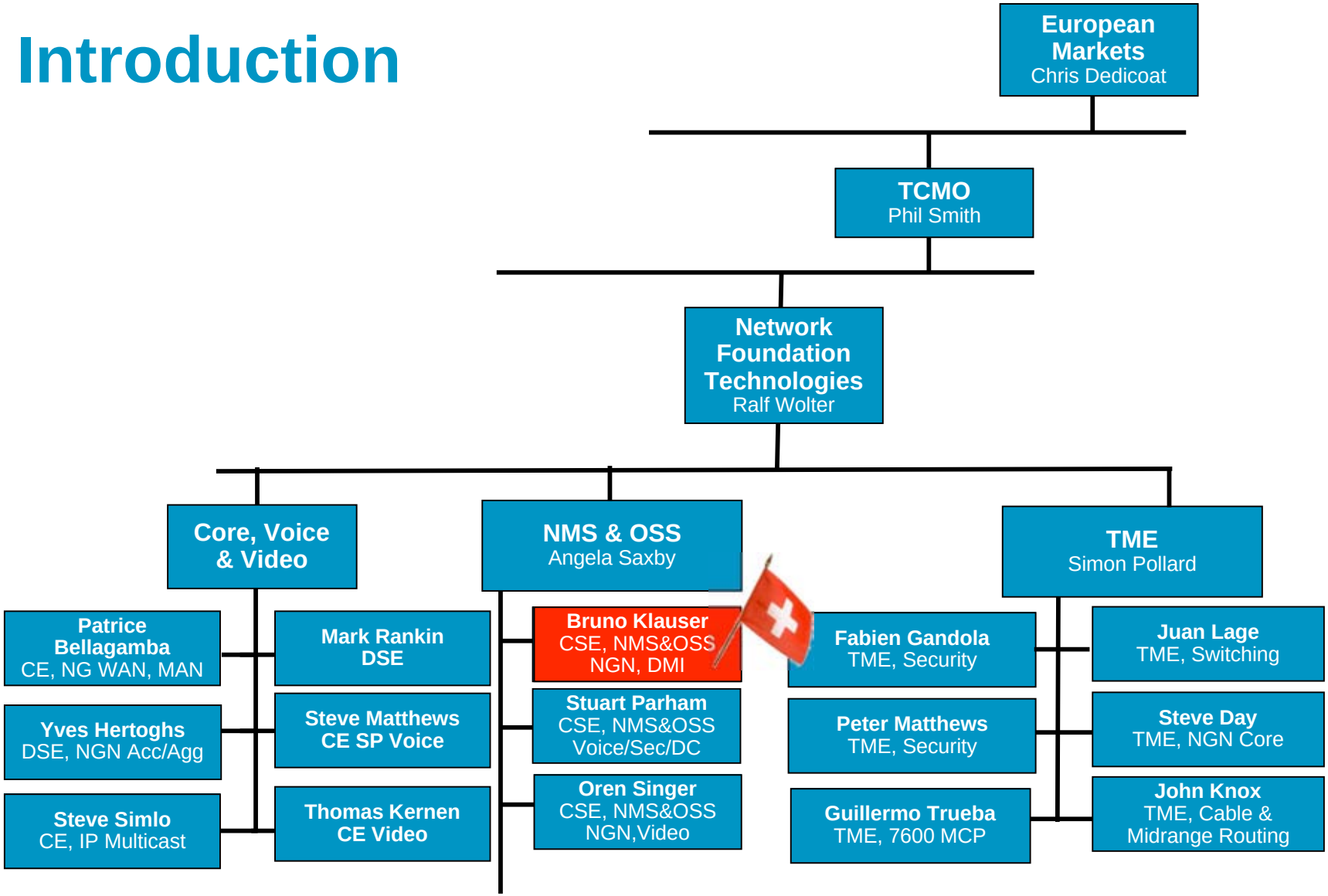
Airliner	Router	Network
8‘000 ‚instruments‘	MIB OIDs	Routers
21‘000 sensors		Links

- With increasing scale and complexity, things become hard to control entirely from the outside
  (hard = inaccurate, time- or resource-consuming, otherwise expensive)

From: Full control by a single central authority

To: Operating a system of self-managing components

Introduction



see: wwwin-people.cisco.com/bklauser/aboutme.html

see: http://zed.cisco.com/confluence/display/EUTMO/Network+Management+Team

Abstract

Does your network meet the expectations and requirements implied by business critical services? If so: can you prove it?

It has been said that device manageability instrumentation is one of Cisco's best kept secrets. Cisco IOS harbors an amazing wealth of functionality for measuring and managing network services.

How can you leverage the network's device manageability instrumentation features? How can you incorporate device manageability instrumentation upon service design?

During this seminar we will walk through the various stages in the live cycle of a network based service, and illustrate – based on technology updates and best practice examples – how the network's device manageability instrumentation can be used to achieve, verify and report on critical business objectives:

- service planning
- deployment and activation
- testing and verification
- ongoing service assurance
- troubleshooting and optimization

Introduction & Overview

Welcome aboard ...

This session is not

- An introduction to NMS concepts
- An in-depth session on 1 single feature
- About engineering details of SNMP
- About NMS applications

This Session is:

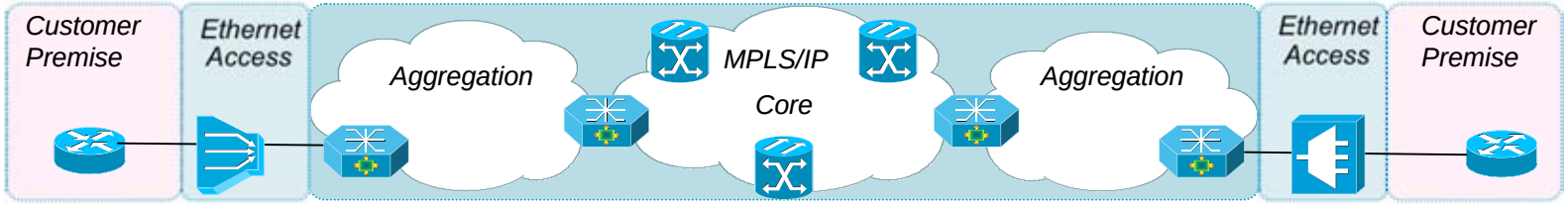
- About Device Manageability Instrumentation (DMI) embedded within the devices
- Organized along a service life cycle
- Full of practical examples

Agenda

- Introduction & Overview
- Service Planning
- [Coffee Break]
- Service Deployment & Activation
- [Lunch Break]
- Service Testing, Verification & Assurance
- [Coffee Break]
- Troubleshooting & Optimization
- Summary

Introduction & Overview

Manageability is a Prerequisite



Device Manageability Instrumentation (DMI)

Fault

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

Configuration

- **E-LMI**—(service parameter and status signaling)
- **E-DI**—(Enhanced Device Interface, CLI, Perl, IETF Netconf)
- **XML PI—**(IETF Netconf)
- **TR-069**
- **KRON**—command scheduler
- **Config change**—logging and notifications
- **Config replace and rollback**
- **Diff**—context diff utility
- **MIB persistence**
- …

Performance

- **IP SLA**—delay, jitter, packet loss, MPLS health monitoring, advanced object tracking
- **CBQoS MIB**—(class-based QoS)
- **NBAR**
- **RMON**
- **ERM**—Embedded Resource Manager
- **GOLD**—Generic Online Diagnosis
- …

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
- …

See also: www.cisco.com/go/instrumentation

Introduction & Overview

Evolving the Operations Paradigm

From: Open Loop, Retro-Fit

Paper-based SLA Contracts retro-fitted onto Services. Compliance evaluated against existing Fault- and Perf- Implementation

Supression, Filtering, De-Duplication, Correlation

Arbitrary, often inconsistent, sometimes contradicting events and values

Entire Network Services built without decomposition

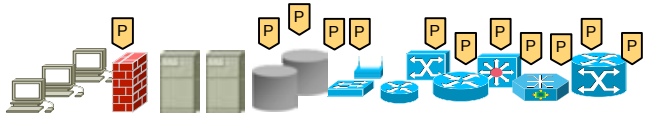
SLA

Known today as SLA Management

Events

Values

Known today as Fault- and Performance Management



To: Closed Loop, Deterministic Design

Well-specified Service Components with associated Metrics are assembled and managed to meet overall (SLA) specification throughout the entire Service Life Cycle

CONNECTIVITY

CORE

L2VPN

MPLS

BGP

Metrics

Metrics

Metrics

SLA

Impact

Policy

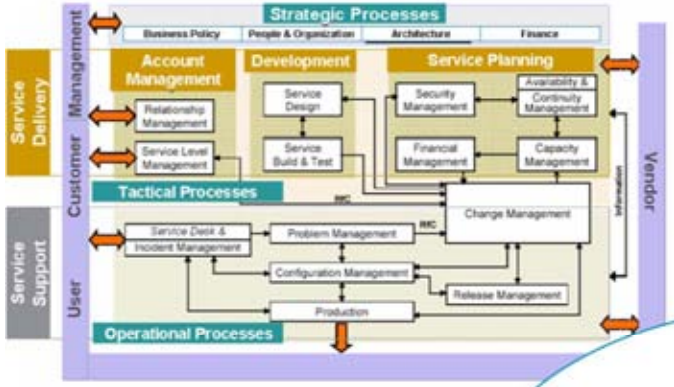
Condition

Action

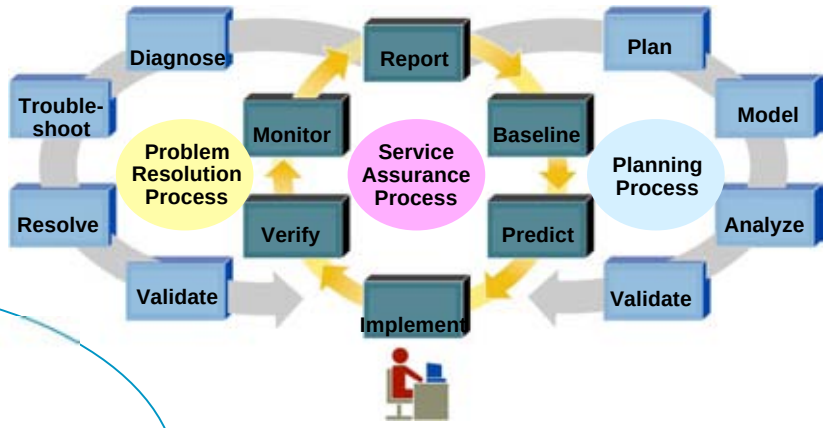
Events

Values

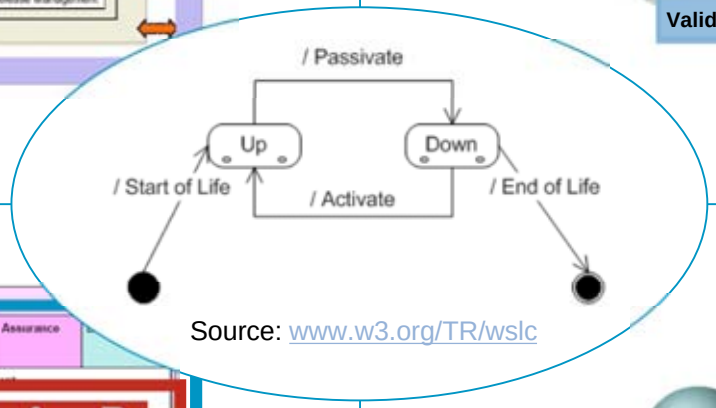
Introduction & Overview
Did You Say Service Life Cycle ?



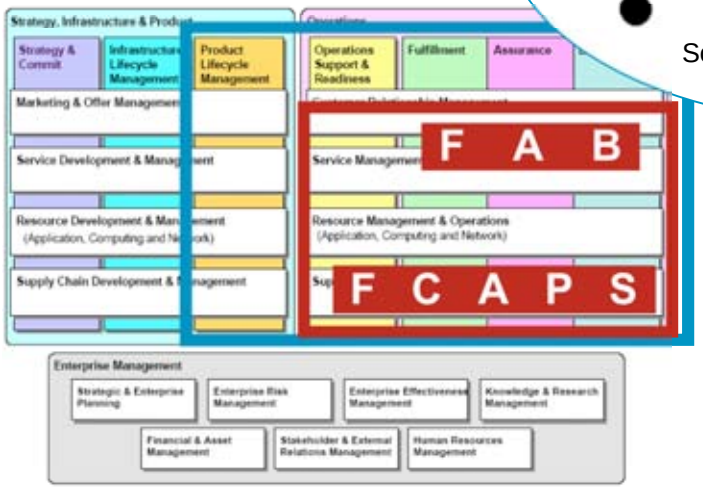
Source: www.itil.org ITSM v2



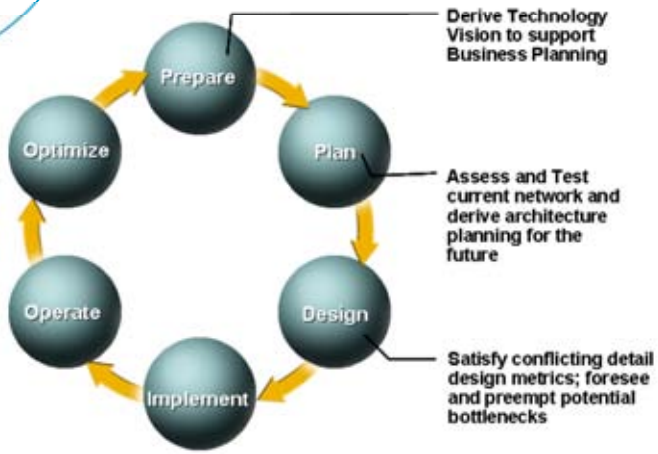
Source: www.cisco.com/go/napas



Source: www.w3.org/TR/wslc



Source: www.tmforum.org



Source: www.cisco.com/go/services

Introduction & Overview

Having a Service Life Cycle

- Have, know, live and continuosely improve **your** lifecycle process
- Make sure it supports you in meeting **your** network service objectives

Introduction & Overview

Questions during a Service Life Cycle

Is there room for yet another service?

- How do we perform today ?
- Are there existing issues ?
- Will we meet specs ?
- Resource consumption ?
- ...

How to configure?

- 1 or many nodes ?
- CLI, scripts, automation ?
- Can we afford downtime ?
- Quality & Security ?
- ...

Is it working as specified?

- Configuration ?
- Control Plane ?
- Data Plane ?
- Were my design assumptions right ?
- ...

- How to Diagnose ?
- Could we offer even tighter SLA ?
- Automate Remedy & Mitigation ?
- ...

What if something goes wrong?

- Will we breach any SLA ?
- What is our performance ?
- ...

Are we meeting SLA?

Introduction & Overview

Feature Availability

- Main focus on what is available in IOS 12.4(15)T on ISR platforms
- Most Features have been around for some time already
- More Details in Appendix I
- Feature Navigator: www.cisco.com/go/fn

Cisco 7304 Router	Cisco 7301 and 7200 Routers	Cisco Catalyst 6500 Series	C 4			12.4(4)T	12.4(2)T	12.3(14)T	12.3(4)T	12.3(2)T	12.2(12)T
12.2SB	12.2SB/SR	12.2SX/ SR				X	X	X	X	X	X
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH				X	X	X	X	X	X
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH				X	X	X	X	X	
12.2(25)S	12.2(31)SB	12.2(1st)SXH				X	X	X	X		
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(44)SE	12.3(14)T	X	X	X			
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(6th)SE	12.4(2)T						
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(6th)SE	12.4(4)T						
12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(31)SGA	NA	NA						
12.2(31)SB	12.2(31)SB	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE							
12.2(31)SB	12.2(31)SB	HD	12.2(13th)SG	12.2(7th)SE	12.5(2nd)T						

Agenda

Introduction & Overview

➲ Service Planning

[Coffee Break]

Service Deployment & Activation

[Lunch Break]

Service Testing, Verification & Assurance

[Coffee Break]

Troubleshooting & Optimization

Summary

“**Plan** [noun]

A set of decisions about how to do something in the future.”

Cambridge Dictionary

http://dictionary.cambridge.org

Service Planning

Learn from your existing Services …

New Services
- Application requirements
- SLA‘
- Product management

Baselining
- Monitoring
- Performance sources
- **Collected data**
- **Historical data**

Service Planning

Current Services
- SLA compliance
- **Resources** and capacity
- Deliverables

Problem Management
- Incidents
- Problem sources
- Troubleticketing

Other Sources
- Paperwork and whiteboard
- Discussions
- Projects and experience

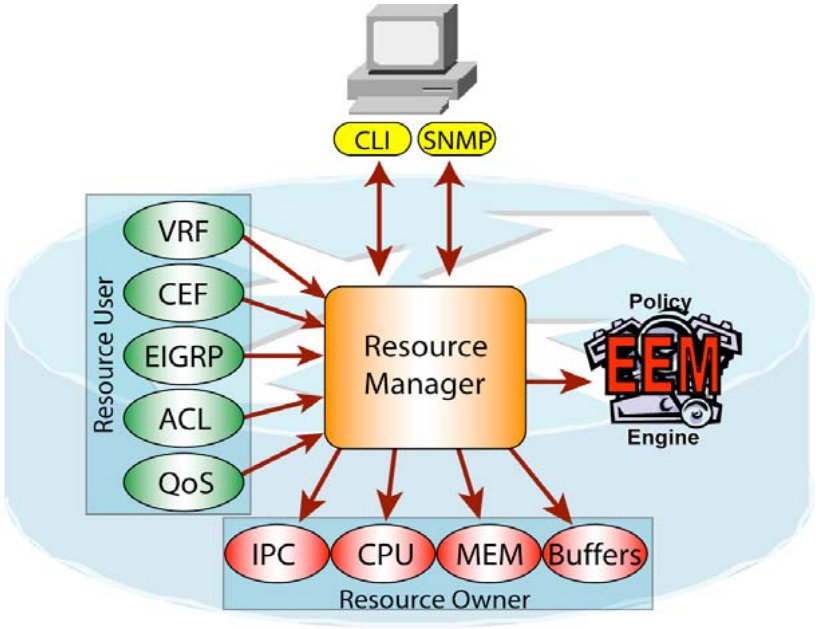
How Is My Current Use of Resources?

Service Planning

Embedded Resource Manager (ERM)

Monitor system resource usage to better understand scalability needs

- Resource: CPU, Buffer, Memory for System or Line Card
- Resource User (RU): Entity or application that consumes one or more resources, e.g. a process
- Resource Owner (RO): Entity that allocates its resources to a RU, e.g. CPU, memory, buffer
- Threshold Notifications:
  - System Global upon entire resource reaching specified value. Notification sent to all RUs.
  - User Local upon a specified RU's utilization reaching specified value. Notification sent to specified RU only.
  - Per User Global upon entire resource reaching specified value. Notification sent to specified RU only
- Interface into EEM



Available since 12.3(14)T (1800, 2800, 3800, 7200)

Service Planning

Example: Monitoring Resources

- **Problem**: During the planning cycle, we would like to understand if total CPU usage reaches critical levels
- **Solution**: Define an ERM policy to notify upon resource depletion

```
resource policy
 policy my-erm-policy-1 type iosprocess
  system
    cpu total
     critical rising 90 interval 15 falling 20 interval 10 global
     major rising 70 interval 15 falling 15 interval 10 global
     minor rising 60 interval 15 falling 10 interval 10 global
!
```

➔ If **Total** CPU Usage Count Rises Above 90% at an Interval of 15s, a Critical Up Notification Is Sent to the iosprocess RU

```
Feb 17 13:32:18.283: %SYS-4-CPURESRISING: System is seeing global
cpu util 62% at total level more than the configured minor limit 60%
```

What Traffic Volumes Flow Through My Network?

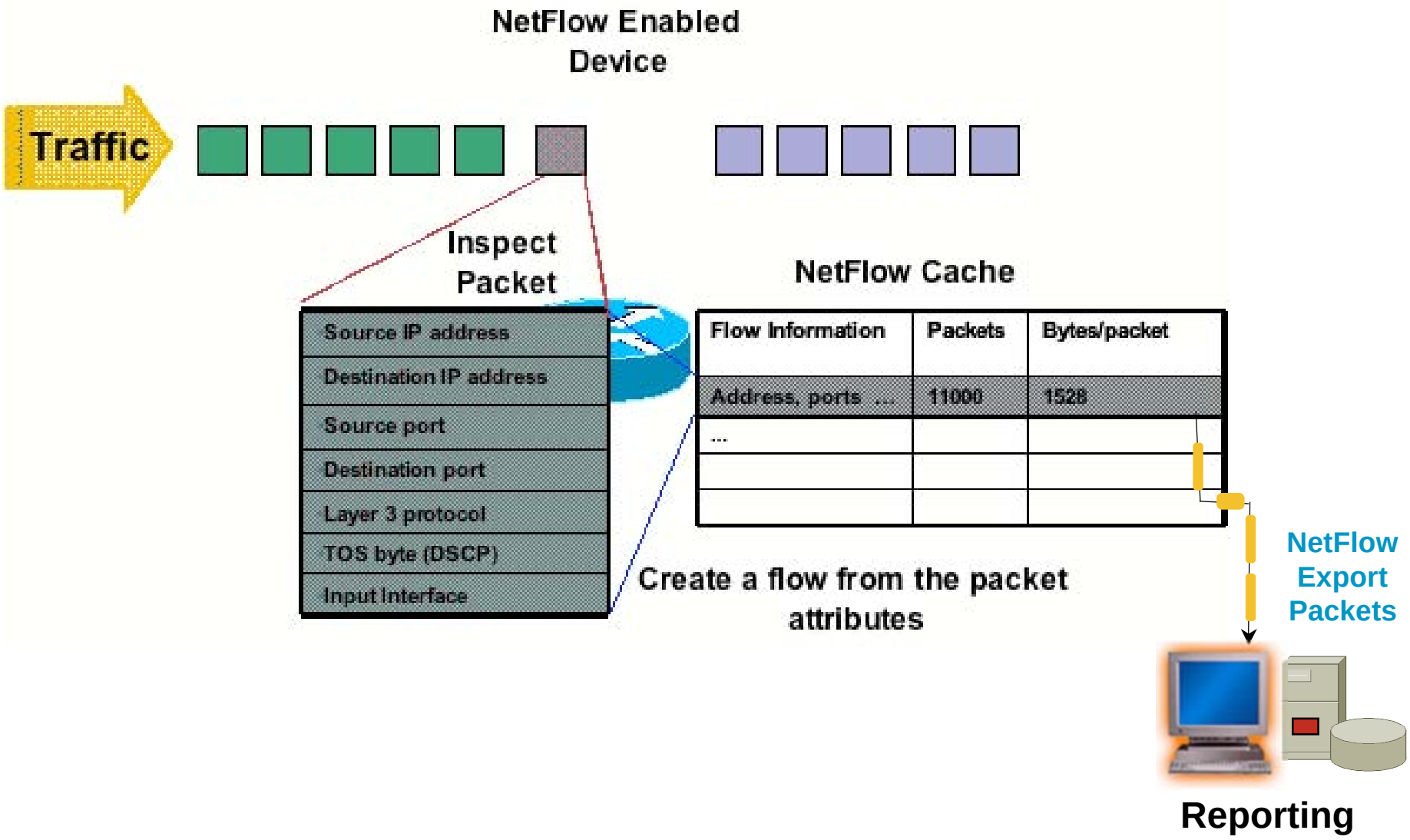
Service Planning
What is NetFlow ?

- Developed and patented at Cisco® Systems in 1996
- NetFlow is the defacto standard for acquiring IP operational data
- Provides network and security monitoring, network planning, traffic analysis, and IP accounting
- NetFlow v9 serves as the basis for IETF IPFIX Standard (RFC3954)

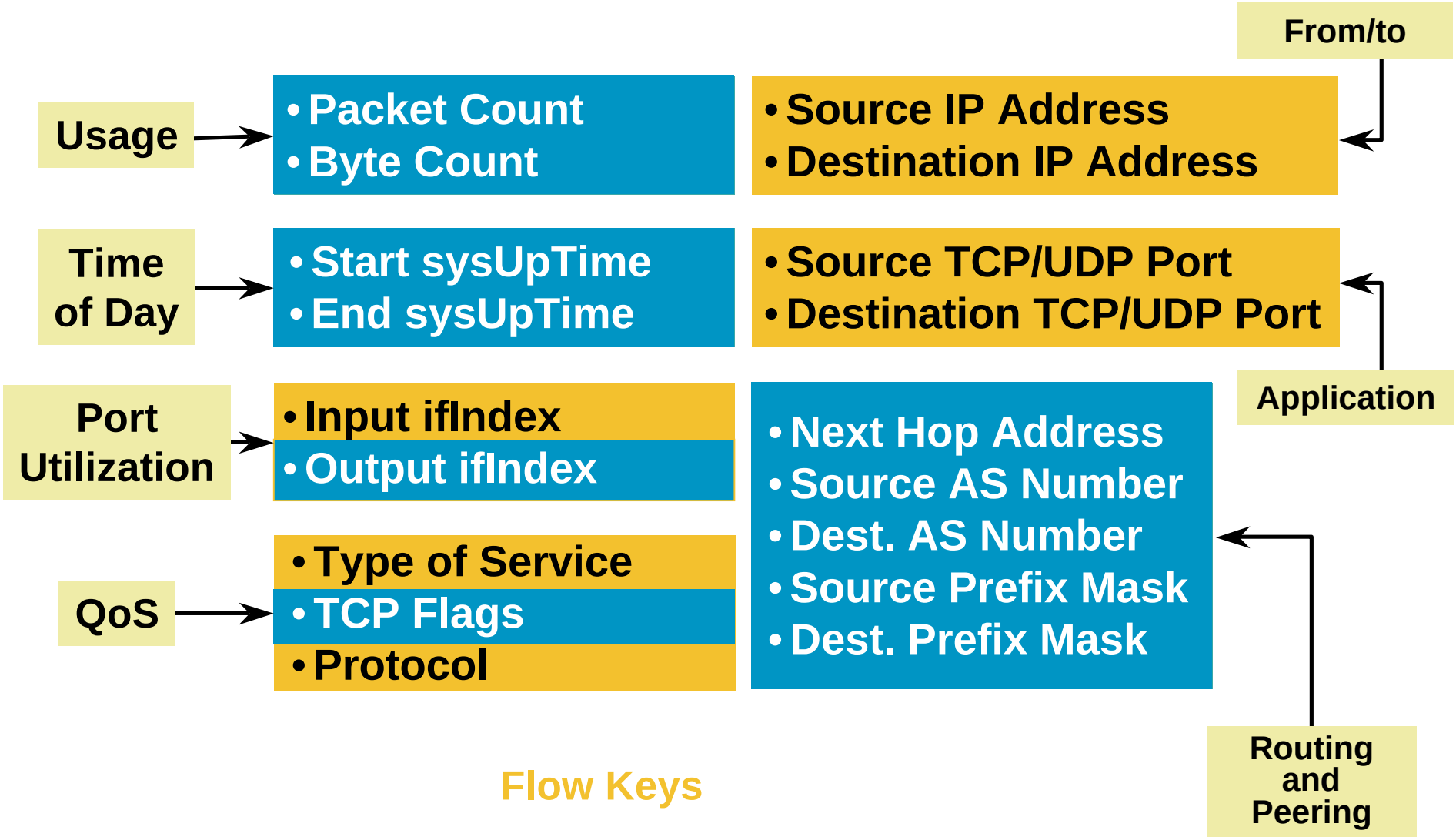
Network World article – NetFlow Adoption on the Rise

http://www.networkworld.com/newsletters/nsm/2005/0314nsm1.html

Flow Is Defined By Seven Unique Keys



Version 5 Flow Format



NetFlow Cache Example

1. Create and update flows in NetFlow cache

SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A 2	/24	5	00A2	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	00A 1	/24	180	00A1	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

2. Expiration

- **Inactive Timer Expired (15 Sec Is Default)**
- **Active Timer Expired (30 Min Is Default)**
- **NetFlow Cache Is Full (Oldest Flows Are Expired)**
- **RST or FIN TCP Flag**

SrcIf	SrcIPadd	DstIf	DstIPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

3. Aggregation

No

Yes

4. Export version

Non-aggregated flows—export version 5 or 9

5. Transport protocol

Export Packet

E.g. Protocol-Port Aggregation Scheme Becomes

Protocol	Pkts	SrcPort	DstPort	Bytes/Pkt
11	11000	00A2	00A2	1528

Aggregated Flows—Export Version 8 or 9

NetFlow Export Version 5 and Main Cache Configuration Example

```
Router(config)# interface <slot/port/subinterface>
Router(config-if)# ip flow ingress
Router(config-if)# ip flow egress

Router(config)# ip flow-cache entries <number>
Router(config)# ip flow-cache timeout active <minutes>
Router(config)# ip flow-cache timeout inactive <seconds>

Router(config)# ip flow-export version 5 peer-as
Router(config)# ip flow-export destination 10.10.10.10 1234
Router(config)# ip flow-export source loopback 0
```

Show NetFlow Information
‘show ip cache flow’

```
router_A#sh ip cache flow
IP packet size distribution (85435 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 1.00 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  2728 active, 1368 inactive, 85310 added
  463824 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
  last clearing of statistics never
Protocol         Total     Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows      /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-X                2       0.0         1  1440      0.0       0.0       9.5
TCP-other        82580      11.2         1  1440     11.2       0.0      12.0
Total:           82582      11.2                      0.0       0.0      12.0

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Et0/0         132.122.25.60   Se0/0         192.168.1.1     06 9AEE 0007     1
Et0/0         139.57.220.28   Se0/0         192.168.1.1     06 708D 0007     1
Et0/0         165.172.153.65  Se0/0         192.168.1.1     06 CB46 0007     1
```

Packet sizes

# of active flows

Rates and duration

Flow details cache

‘show ip cache verbose flow’

```
router_A#sh ip cache verbose flow
IP packet size distribution (23597 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 1.00 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  1323 active, 2773 inactive, 23533 added
  151644 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
  last clearing of statistics never
Protocol         Total     Flows   Packets Byt    Packets Active(Sec   dle(S )
--------         Flows      /Sec     /Flow  /P      /Sec     /Flow     /Flo
TCP-other        22310       3.1         1  14 0     3.1       0.0      12.
T                            3.1         1  14 0     3.1       0.0      12.

SrcIf           SrcIPaddress    DstIf           DstIPaddress    Pr TOS Flgs  Pkts
Port Msk AS                     Port Msk AS     NextHop            B/Pk  Active
Et0/0           216.120.112.114 Se0/0           192.168.1.1     06 00  10       1
5FA7 /0  0                      0007 /0  0      0.0.0.0            1440     0.0
Et0/0           175.182.253.65  Se0/0           192.168.1.1     06 00  10       1
```

Flow rate and duration

ToS byte and TCP flags

Destination information

Source mask and ISP AS

Extensibility and Flexibility Requirements Phases Approach

- Traditional NetFlow with the v5, v7, or v8 NetFlow export
  - New requirements: build something flexible and extensible
- Phase 1: NetFlow version 9
  - Advantages: extensibility
    - Integrate new technologies/data types quicker (MPLS, IPv6, BGP next hop, etc.)
    - Integrate new aggregations quicker
  - Note: for now, the template definitions are fixed

Exporting Process

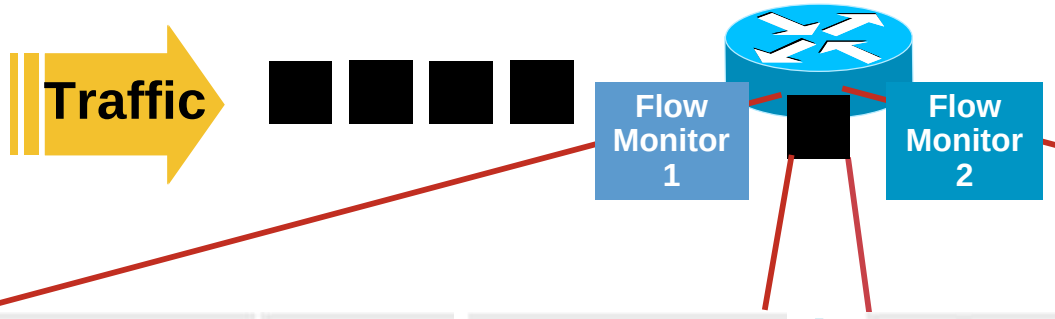
- Phase 2: Flexible NetFlow
  - Advantages: cache and export content flexibility
    - User selection of flow keys
    - User definition of the records

Metering Process

Flexible NetFlow
High Level Concepts and Advantages

- Flexible NetFlow feature allows user configurable NetFlow record formats, selecting from a collection of fields:
  - Key
  - Non-key
  - Counter
  - Timestamp
- Advantages:
  - Tailor a cache for specific applications, not covered by existing 21 NetFlow features
  - Better scalability since flow record customization for particular application reduces number of flows to monitor
  - Different NetFlow configuration:
    - Per subinterface
    - Per direction (ingress/egress)
    - Per sampler
    - Etc.

Flexible NetFlow
Multiple Monitors with Unique Key Fields



Key Fields	Packet 1
Source IP	3.3.3.3
Destination IP	2.2.2.2
Source Port	23
Destination Oort	22078
Layer 3 Protocol	TCP - 6
TOS Byte	0
Input Interface	Ethernet 0

Non-Key Fields
Packets
Bytes
Timestamps
Next Hop Address

Key Fields	Packet 1
Source IP	3.3.3.3
Dest IP	2.2.2.2
Input Interface	Ethernet 0
SYN Flag	0

Non-Key Fields
Packets
Timestamps

Traffic Analysis Cache

Source IP	Dest. IP	Source Port	Dest. Port	Protocol	TOS	Input I/F	…	Pkts
3.3.3.3	2.2.2.2	23	22078	6	0	E0	…	1100

Security Analysis Cache

Source IP	Dest. IP	Input I/F	Flag	…	Pkts
3.3.3.3	2.2.2.2	E0	0	…	11000

Flexible Flow Record—Key Fields

IPv4	
IP (Source or Destination)	Payload Size
Prefix (Source or Destination)	Packet Section (Header)
Mask (Source or Destination)	Packet Section (Payload)
Minimum-Mask (Source or Destination)	TTL
Protocol	Options bitmap
Fragmentation Flags	Version
Fragmentation Offset	Precedence
ID	DSCP
Header Length	TOS
Total Length	

Routing
src or dest AS
Peer AS
Traffic Index
Forwarding Status
Is-Multicast
IGP Next Hop
BGP Next Hop

Flow
Sampler ID
Direction

Interface
Input
Output

Transport	
Destination Port	TCP Flag: ACK
Source Port	TCP Flag: CWR
ICMP Code	TCP Flag: ECE
ICMP Type	TCP Flag: FIN
IGMP Type	TCP Flag: PSH
TCP ACK Number	TCP Flag: RST
TCP Header Length	TCP Flag: SYN
TCP Sequence Number	TCP Flag: URG
TCP Window-Size	UDP Message Length
TCP Source Port	UDP Source Port
TCP Destination Port	UDP Destination Port
TCP Urgent Pointer	

Flexible Flow Record—Non-Key Fields

Counters
Bytes
Bytes Long
Bytes Square Sum
Bytes Square Sum Long
Packets
Packets Long

Timestamp
sysUpTime First Packet
sysUpTime Last Packet

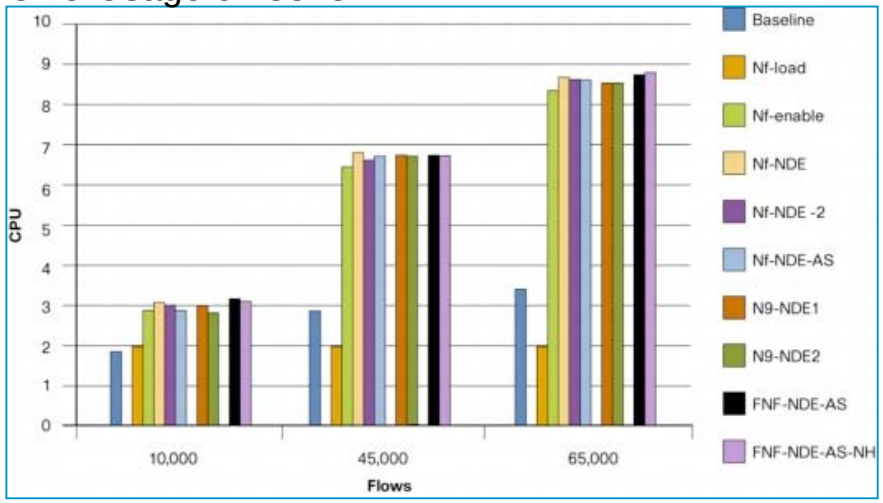
IPv4
Total Length Minimum
Total Length Maximum
TTL Minimum
TTL Maximum

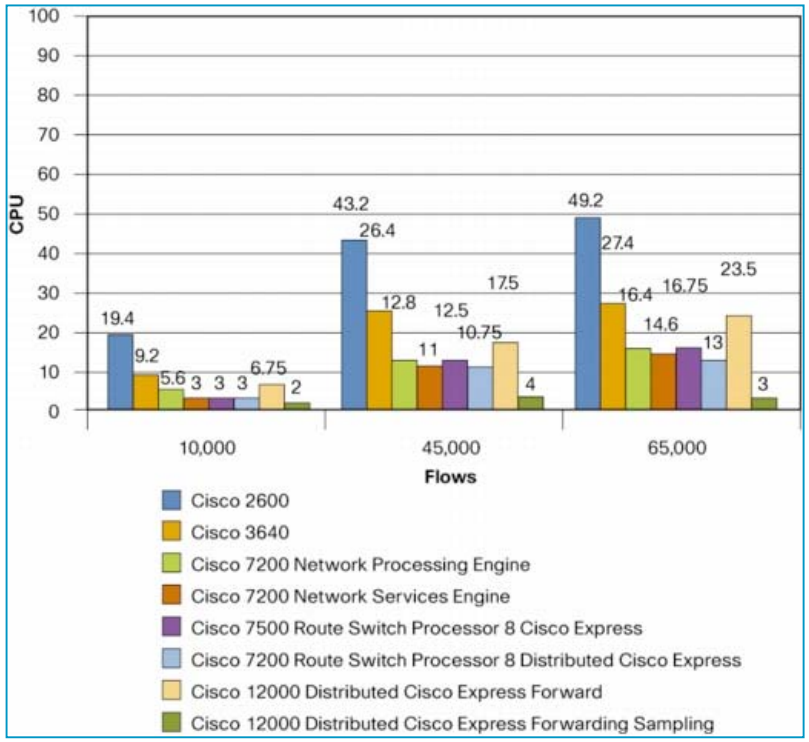
- Plus any of the potential “key” field: will be the value from the first packet in the flow

NetFlow Resource Consumption

- CPU Usage determined by active Flows in the Cache
- Minimal differences between v5, v8 or v9
- Minimal impact of multiple export destinations
- Minor (5-10%) impact of Flexible Netflow (FNF)

CPU Usage on 3845



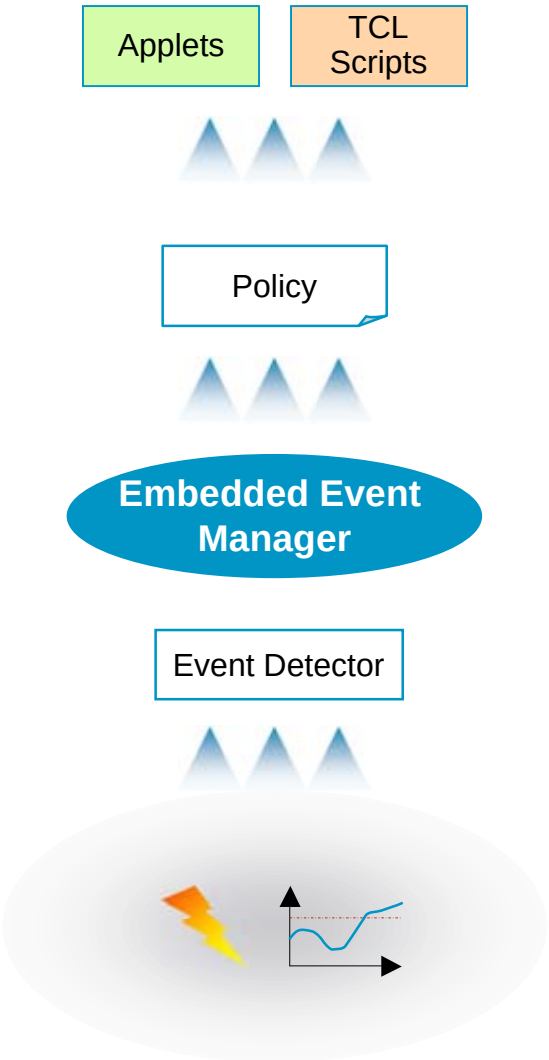


See: http://www.cisco.com/en/US/tech/tk812/technologies_white_paper0900aecd802a0eb9.shtml

How To Analyze Transient Conditions?

Service Testing, Verification and Assurance

Embedded Event Manager (EEM)



- An EEM Policy is activated that initiates a pre-defined set of actions
- An EEM Event Detector receives notification
- Something happens on the  causing an Event to trigger

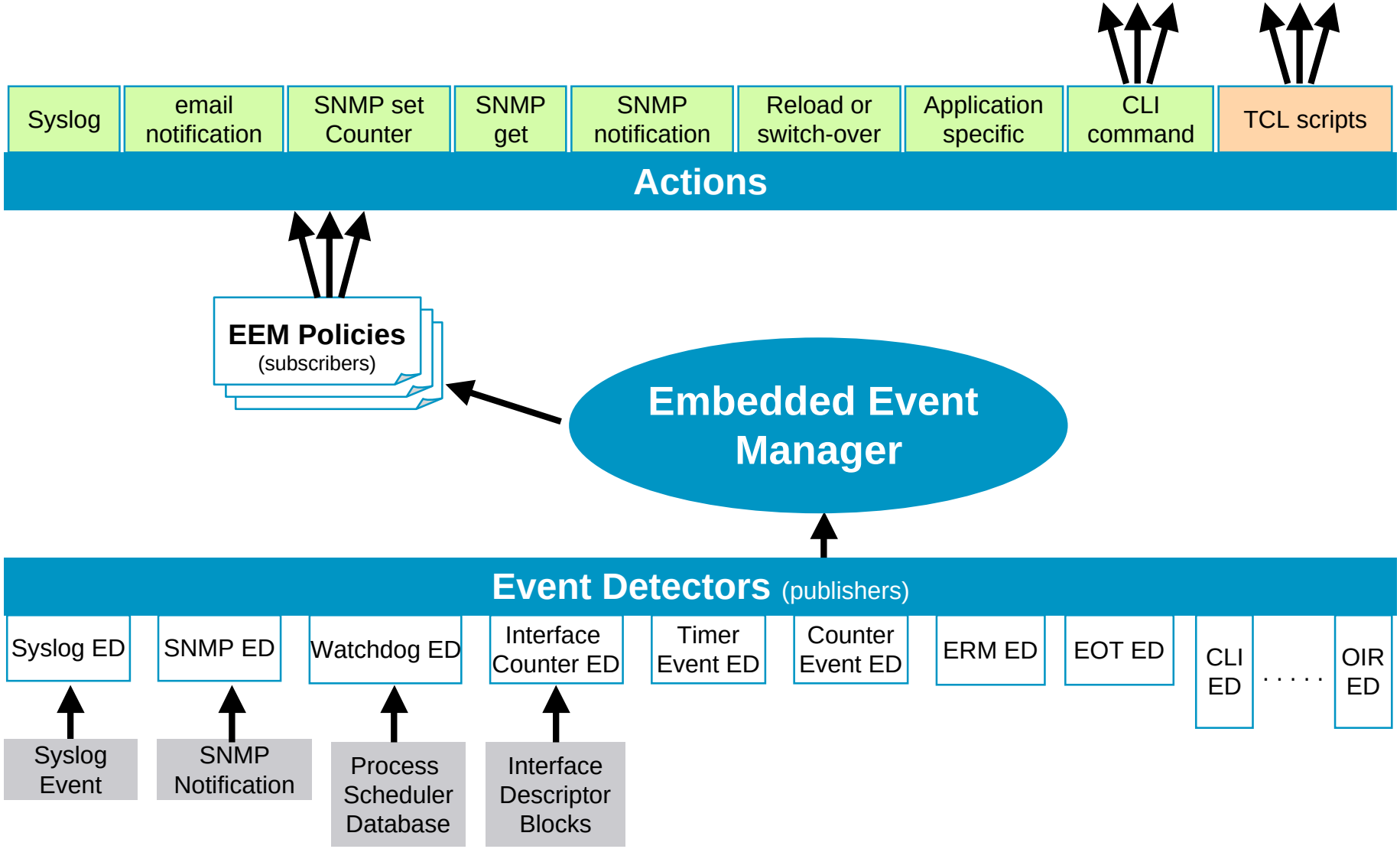
Service Testing, Verification and Assurance

What Is Embedded Event Manager (EEM) ?

- Embedded monitoring of different components of the system via a set of software agents (event detectors)
- Event detectors (ED) notify EEM when an event of interest occurs; based on this, a policy will trigger an action to be taken
- Advantages: Local programmable actions, triggered by specific events – growing set of detectors and actions:
  - Version 1.0 introduced in 12.0(26)S, 12.3(4)T
  - Version 2.0 introduced in 12.2(25)S
  - Version 2.1 introduced in 12.3(14)T
  - Version 2.2 introduced in 12.4(2)T
  - Version 2.3 introduced in 12.4(11)T
  - Upcoming Version 2.4 in 12.4(20)T — Adds multi-event correlation
  - Upcoming Version 3.0 in 12.5(pi1)T
  - stay tuned ...

Service Testing, Verification and Assurance

EEM Architecture



EEM
Policies can be either Applets or TCL Scripts

Applets

(*.txt)

- Applets are created using a set of CLI commands
- The applet becomes part of the Cisco IOS configuration file and is persistent across system reboots
- Use a single “event” statement following by a number of “action” statements

TCL Scripts

(*.tcl)

- TCL scripts cannot be built from the switch CLI
- This form of script offers a more flexible and powerful option for network administrators to apply actions on a given event occurrence
- Like the applet, a registered TCL script is persistent across system reboots

EEM
Applets

Applets

Environment Variable(s) (Optional)

Applet Name

Event Statement

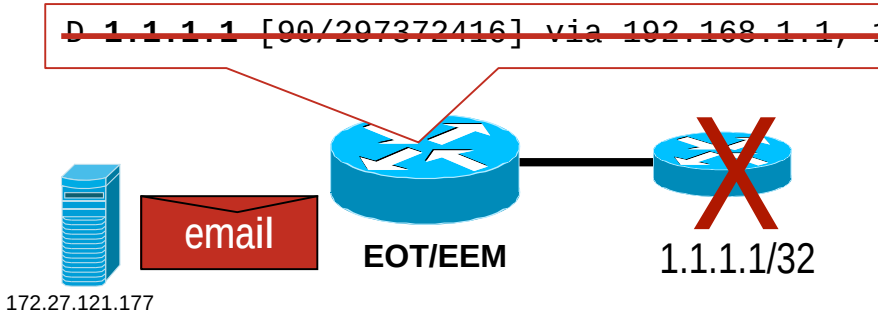
Action Statement

1. Configure any required environment variables
2. Register applet
3. Define event used to trigger applet
4. Specify actions to be taken

Service Testing, Verification and Assurance

Example: Layer 3 Path Failure Detection

- **Problem:** A Notification is required upon failure of a specific route
- **Solution:** Track the Route using Enhanced Object Tracking (EOT) and Embedded Event Manager (EEM)




```
track 400 ip route 1.1.1.1/32 reachability
   delay down 10 up 10
!
event manager environment my_server 172.27.121.177
event manager environment my_from router-abc@customer.com
event manager environment my_to attach@cisco.com
event manager environment my_route 1.1.1.1/32
!
event manager applet email_track_iproute
event track 400 state down
action 1.0 syslog msg "Prefix to [$my_route] has been withdrawn!"
action 1.1 mail server "$my_server" to "$_email_to" from "$my_from“
 subject “EEM: Prefix to Remote Site [$my_route] is DOWN" body ""
action 1.2 syslog msg “EEM: Path Failure alert email sent!"
```

EEM
TCL Scripts

TCL Scripts

- Event Register Keyword
- Environment Variables (Optional)
- Namespace Import
- Body of Code

EEM
TCL Script Example

```
::cisco::eem::event_register_syslog occurs 1 pattern .*%SYS-4-FREEMEM.* queue_priority low nice 1 maxrun 90

##########################################################################################
#
#  Revision #        :  1.7
#  Last Updated      :  September 23, 2007
#  Author/Contributor :  David Lin, dalin@cisco.com
#
#  Description       :  The following script utlizes the Memory Th
#                       introduced in Cisco IOS 12.2(18)S, 12.0(26)S and 12.3(4)T.
#                       This feature allows one to mitigate low-memory conditions on a router.
#                       When free processor or I/O memory has fallen below a configured threshold,
#                       an email will be sent and include output from the syslog, "show version", "show memory summary"
#                       and "sh processes memory sorted holding"
#
#  Requirements      :  -Email related environment variables-
#                          event manager environment _email_server <your-mailserver-ipaddress or dns-name>
#                          event manager envrionment _email_from <your-email-from-address>
#                          event manager environment _email_to <your-email-to-address>
#
#                        Example: event manager environment _email_server 10.10.10.10
#                          event manager environment _email_from router-123@cisco.com
#                          event manager environment _email_to noc@cisco.com
```

Event register

EEM runtime
Default = 20 seconds
Increase this value if you see a "Process Forced Exit" message from the router.

EEM
TCL Script Example

Other types of event registers you may encounter…

None: Triggered manually via “event manager run” command.
::cisco::eem::event_register_none queue_priority low nice 1 maxrun 60

Watchdog Timer: Triggered by time (in sec) specified by value/environment variable after the keyword “time”
::cisco::eem::event_register_timer watchdog name foobar time $time_period queue_priority low nice 1
(The above example requires the global command “event manager environment time_period <sec>”)

Syslog: Triggered by pattern match of syslog msg
::cisco::eem::event_register_syslog occurs 1 pattern .*%SYS-5-CONFIG-I.* queue_priority low nice 1 maxrun 90

Object Tracking: Triggered by state of Enhanced Object Tracking (EOT) reaching “DOWN” state.
::cisco::eem::event_register_track 1 state up queue_priority low nice 1

EEM
TCL Script Example

Other types of event registers you may encounter (cont.)

Cron Job
::cisco::eem::event_register_timer cron name business_hours cron_entry "0 9-17 * * 1-5" queue_priority low nice 1

The above cron job will trigger every hour between 9am-5pm, Mon-Fri

The cron_entry “0 9-17 * * 1-5” will do this:
- The 0 means the first minute of the hour.
- The 9-17 means hours 9am to 5pm
- The next * means every day of the month.
- The next * means every month.
- The final 1-5 means Monday through Friday.

EEM
TCL Script Example

```
…

# Namespace imports

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

#--- Check required environment variable(s) has been defined

if {![info exists _email_server]} {
    set result "EEM Policy Error: variable _email_server has not been set."
    error $result $errorInfo
}
if {![info exists _email_to]} {
    set result "EEM Policy Error: variable _email_to has not been set."
    error $result $errorInfo
}
if {![info exists _email_from]} {
    set result "EEM Policy Error: variable _email_from has not been set."
    error $result $errorInfo
}
```

Import EEM Library Files

Environment Variable Check

EEM
TCL Script Example

```
#------------------- hostname ---------------------
set routername [info hostname]

#
#------------------   " cli open"   ------------------
#
if [catch {cli_open} result] {
  error $result $errorInfo
} else {
  array set cli $result
}
```

copy hostname to variable *'routername'*

Open CLI command

EEM
TCL Script Example

```
#---------------------- "show commands" ----------------

  if [catch {cli_exec $cli(fd) "enable"} result] {
    error $result $errorInfo
  }

  if [catch {cli_exec $cli(fd) "show version"} result] {
    error $result $errorInfo
  }
  set show_version $result

  if [catch {cli_exec $cli(fd) "show memory summary | include ^          Head|^     I/O|^Processor"} result] {
    error $result $errorInfo
  }
  set output_1 $result

  if [catch {cli_exec $cli(fd) "show processes memory sorted holding"} result] {
    error $result $errorInfo
  }
  set output_2 $result

#---------------------- end of show commands ------------
```

Enter CLI enable mode

Capture show commands

EEM
TCL Script Example

```
#---------------------- send mail ----------------------
  action_syslog msg "Creating mail header..."
  set body [format "Mailservername: %s" "$_email_server"]
  set body [format "%s\nFrom: %s" "$body" "$_email_from"]
  set body [format "%s\nTo: %s" "$body" "$_email_to"]
  set _email_cc ""
  set body [format "%s\nCc: %s" "$body" ""]
  set body [format "%s\nSubject: %s\n" "$body" "Router is running low on memory! (hostname:$routername)"]
  set body [format "%s\n%s" "$body" "Report Summary:"]
  set body [format "%s\n%s" "$body" "   - Show Version"]
  set body [format "%s\n%s" "$body" "   - Syslog Message"]
  set body [format "%s\n%s" "$body" "   - Show Memory Summary"]
  set body [format "%s\n%s" "$body" "   - Show Processes Memory Sorted Holding"]
  set body [format "%s\n\n%s" "$body" "---------- Show Version ----------"]
  set body [format "%s\n%s" "$body" "$show_version"]
  set body [format "%s\n\n%s" "$body" "---------- Syslog Message ----------"]
  set body [format "%s\n\n%s" "$body" "$syslog_msg"]
  set body [format "%s\n\n%s" "$body" "---------- Show Memory Summary ----------"]
  set body [format "%s\n\n%s" "$body" "$output_1"]
  set body [format "%s\n\n%s" "$body" "---------- Show Processes Memory Sorted Holding ----------"]
  set body [format "%s\n\n%s" "$body" "$output_2"]
  if [catch {smtp_send_email $body} result] {
    action_syslog msg "smtp_send_email: $result"
  }
#--------------------- cli close ------------------------
  cli_close $cli(fd) $cli(tty_id)
# End of Script
```

Compose email message from show output

Service Testing, Verification and Assurance

EEM Event Detectors currently available

Cisco IOS CLI
Triggers policies based on commands entered via the CLI.

Cisco IOS Counter
Policies can be triggered based on a change of the designated Cisco IOS counter.

Cisco IOS Redundancy Facility
Provides for detection of hardware and software failures related to the Stateful Switchover service. This ED will trigger policies based on the RF state change. It is also used to initiate switchovers as a result of a policy action.

Cisco IOS Timer Services
Policies can be scheduled to occur at the designated time or interval.

Cisco IOS Watchdog / System Monitor
Triggers policies based on certain conditions relative to a certain Cisco IOS process or subsystem's activity.

EEM Application Specific
Application specific events can be detected or set by a Cisco IOS subsystem or a policy script. This provides the ability for one policy to trigger another policy.

XML RPC (SOAP over SSHv2) (new in EEM 2.4)
Triggers upon receipt of an incoming XML message

Interface Counter
Policies can be triggered based on the specific interface counter; includes thresholds.

Online Insertion and Removal
Triggers policies based on hardware installation and removal activity.

Object Tracking
Triggers policies based on routing protocol events.

SNMP
Triggers policies based on the associated SNMP MIB variable; includes MIB variable threshold setting.

SNMP Proxy (new in EEM 2.4)
Triggers upon receipt of an incoming trap or inform

Syslog
Triggers policies based on the regular expression match of a local Syslog message.

Resource Thresholding (ERM)
Triggers policies based on certain internal resource usage and conditions; interface to Embedded Resource Manager.

Generic Online Diagnostics (GOLD)
Triggers policies based on diagnostic results

"None" ED
Triggers policies by command

Service Testing, Verification and Assurance

EEM 2.4: Multiple Event Correlation

- Previous to EEM v2.4, there was a one-to-one correspondence between a single event and the triggered policy
- In other words, a policy could only be triggered by a single event and any event correlation had to be coded by the user
- ***Multiple Event Support ushers in an event correlation specification such that multiple events may be considered together to trigger a policy***
- For example:

  If (Event 1 OR Event 2) AND Event 3, then
  Trigger Policy A

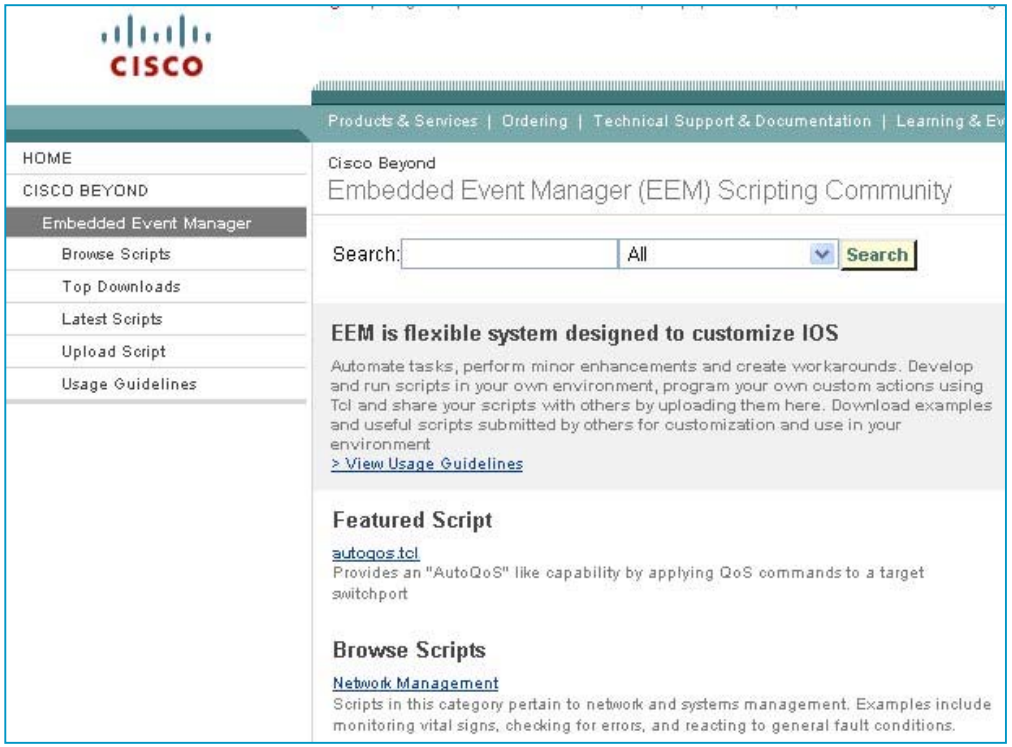
Service Testing, Verification and Assurance

Example: Multiple Event Correlation

- **Problem:** A Syslog message is required upon state change of either Ethernet1/0 or Ethernet1/1
- **Solution:** Use Embedded Event Manager (EEM) Multiple Event Correlation with a correlate statement within the trigger block to define the logic between individual events and optional occurs clauses to define the number of times a specific event must be raised before being used in the correlation (inner level), or the number of times the total correlation must be true before invoking the action (outer level):

```
event manager applet example
  event tag e1 syslog pattern ".*UPDOWN.*Ethernet1/0.*"
  event tag e2 syslog pattern ".*UPDOWN.*Ethernet1/1.*"
  trigger occurs 1
    correlate event e1 or event e2
    attribute e1 occurs 1
    attribute e2 occurs 1
  action 1.0 syslog msg "Critical interface status change"
  set 2.0 _exit_status 0
```

Embedded Event Manager – engage now!



- Device Manageability Instrumentation (DMI): www.cisco.com/go/instrumentation
- Embedded Event Manager: www.cisco.com/go/eem
- EEM Scripting Community: www.cisco.com/go/ciscobeyond
  (internally: http://wwwin-swpkg.cisco.com/fm/central/index.html)

Agenda

Introduction & Overview

Service Planning

[Coffee Break]

➲ Service Deployment & Activation

[Lunch Break]

Service Testing, Verification & Assurance

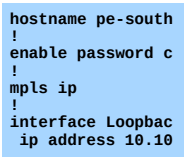
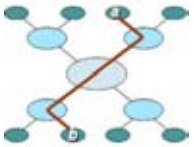
[Coffee Break]

Troubleshooting & Optimization

Summary

Deployment & Activation

Definition of Activities

	Deployment	Move physical network equipment into it‘s operating location
	Commissioning	Make new network equipment ready for use and reachable by operations, NMS
	Configuration	Configure a network element depending on it’s role and function in the network
	Provisioning	Configure portions of a network for the purpose of a specific user and/or service
	Activation	Enable users to start using a service

Focus

Deployment & Activation

The Human Factor ...

```
!
interface Serial1/0.121 point-to-point
  description " NOC Bastelstunde "
```

On the CLI of a Single Router…

Deployment & Activation

IOS Configuration Features

- Contextual configuration diff utility (from 12.3(4)T, 12.2(25)S)
  - Easily show differences between running and startup configuration
  - Compare any two configuration files
- Config change logging and notification (from 12.3(4)T, 12.2(25)S)
  - Tracks config commands entered per user, per session
  - Notification sent indicating config change has taken place—changes can be retrieved via SNMP
- Configuration replace and rollback (from 12.3(7)T, 12.2(25)S)
  - Replace running config with any saved configuration (only the diffs are applied) to return to previous state
  - Automatically save configs locally or off box
- Configuration locking (from 12.3(14)T, 12.2(25)S)
  - Ensures exclusive configuration change access

Deployment & Activation

Example: Using Config Rollback

- **Problem**: critical config change to a remote router may result in loss of connectivity, requiring a reload
- **Solution**: replace the running configuration with the latest good archive after two minutes—unless the change being made is confirmed

```
Router#show archive
There are currently 4 archive configurations saved.
The next archive file will be named disk0:/config-archive-4
 Archive #  Name
   0
   1       disk0:/config-archive-1
   2       disk0:/config-archive-2
   3       disk0:/config-archive-3 <- Most Recent

Router#config replace disk0:/config-archive-3 time 120
  :
  ... your Config Change work here ...
  :
Router# no config replace disk0:/config-archive-3
```

Deployment & Activation

Tool Command Language (TCL)

- Language resources found at: http://www.tcl.tk/
- TCL 7.x has been in Cisco IOS since 1994
- TCL 8.3.4 first released in Cisco IOS in 12.3(2)T and merged into 12.2(25)S
- Use 12.3(14)T or later for best results
- Signed TCL Scripts introduced in 12.4(15)T

```
Router#tclsh slot0:myscript.tcl
Router#tclsh
Router(tcl)#source tftp://10.1.1.1/myscript.tcl
```

- Use low-memory to prevent malloc failures

```
Router(config)#scripting tcl low-memory  <water_mark>
```

- TCL process runs at medium priority, so be careful with loops

Deployment & Activation

Tool Command Language (TCL)

- http://www.cisco.com/go/ciscobeyond
- http://www.cisco.com/go/eem
- http://www.cisco.com/go/ioscommercial

  Example: A VPN failure is defined as failure to reach a set of remote peer's L3 tunnel interface(s) that are configured using GRE + IPSEC over DMVPN

- "Guide To Writing EEM Policies" documentation

```
Router#tclsh
Router(tcl)#puts "Hello Nordics"
Hello Nordics
Router(tcl)#ios_config "interface fa0/0"
       "description Main Uplink"
Router(tcl)#exit
Router#
```

TCL Cisco IOS Extended Commands

TCL Built In Command

Cisco IOS Command

Deployment & Activation
Kron Scheduler

- Run EXEC commands periodically or at a specified time
- First introduced in 12.3(1)
- Runs commands in a fully-automated mode
- Interactive commands (e.g. reload) are NOT supported

Note:

- NTP must be configured or the router clock must be authoritative
- Kron and Tcl can run together since 12.4(4)T

Alternative Option: use Embedded Event Manager (EEM) Timer ED

Deployment & Activation

Example: Archiving Configuration – 1/5

- **Problem:** Device configurations must be archived periodically, collecting them from the outside should not be the only answer.
- **Solution 1:** Archive the running configuration once every day locally:

```
archive
 path disk0:/config-archive
 maximum 7
 time-period 1440
```

View the content of the archive:

```
Router#show archive
There are currently 3 archive configurations saved.
The next archive file will be named disk0:config-archive-3
 Archive #  Name
   0
   1       disk0:config-archive-1
   2       disk0:config-archive-2 <- Most Recent
   3
   4
   5
   6
   7
```

Deployment & Activation

Example: Archiving Configuration – 2/5

Solution 2: Archive the running configuration once every day to a server:

```
archive
 path tftp://10.1.1.1
 time-period 1440
```

Note: Config can also be archived on-demand:

```
Router#archive config
```

Solution 3: Use Kron to schedule periodic archiving (plus other activity)

```
archive
 path tftp://10.1.1.1
!
kron policy-list backupconfig
 cli archive config
!
kron occurrence backup-occur at 23:23 recurring
 policy-list backupconfig
```

multiple policy-lists possible

Deployment & Activation

Example: Archiving Configuration – 3/5

Solution 4: Use Embedded Event Manager (EEM) with a Syslog Event Detector and a TCL Applet to only archive configs if there was a change

Define EEM Environment Variable

```
Router(config)# event manager environment filename <myfile.txt>
Router(config)# event manager directory user policy "flash:/TCL"
Router(config)# event manager policy archive.tcl type user

Router(config)# archive
Router(config-archive)# path flash:disk0
Router(config-archive)# maximum 14
```

Register EEM TCL Script

Configure Archive Location and Size

The script is available from www.cisco.com/go/ciscobeyond

Deployment & Activation

Example: Archiving Configuration – 4/5

```
::cisco::eem::event_register_syslog pattern ".*%SYS-5-CONFIG.*"
#######################################################
# EEM TCL Script to archive the config upon change
#
# Developed by Marisol Palmero
#
# The following EEM environment variable is used:
# - filename: name of the file specified in the path command within
#
# Lets check if all the variable exists, otherwise quit
#######################################################
if {![info exists filename]} {
  set result "Policy cannot be run: variable filename not set"
error $result $errorInfo
}

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

if [catch {cli_open} result] {
  puts stderr $result
  exit 1
} else {
  array set cli1 $result
}
```

Sylog Event

Deployment & Activation

Example: Archiving Configuration – 5/5

```
if [catch {cli_exec $cli1(fd) "en"} result] {
  puts stderr $result
  exit 1
}

set showarchive [cli_exec $cli1(fd) "show archive"]
set lines [split $showarchive "\n"]

foreach line $lines {
  set result [regexp {<- Most Recent} $line ]
  if {$result != 0} {
    set result1 [regexp {^\s+\d+\s+( +)-(\d+)\s+<-} $line -> path extension]
    set output [cli_exec $cli1(fd) "show archive config differences
        system:/running-config flash:$filename-$extension"]
    if { [regexp "!No changes were found" $output] } {
      break
    } else {
      cli_exec $cli1(fd) "archive config"
      break
    }
 } }
if {$result == 0} {
 cli_exec $cli1(fd) "archive config"
}
```

Archive if there was a change of if there was no archived version yet

What if CLI Doesn’t Scale?

Deployment & Activation

Zero-Touch Deployment Methods

Method	Cisco IOS Deployment Agents	External Mediation Server	Notes
DOCSIS	DOCSIS	Cisco Broadband Access Center (BAC)	For Cable Modem Access Only Widely Standardized
TR-069	TR-069	Cisco Broadband Access Center (BAC)	For DSL Access Standard Is Work in Progress with Currently Loose Definition, Check Interop Test from Plugfest
EEM	Embedded Event Manager	FTP, TFTP, SCP,…	Flexibility for Scenarios Not Covered by Any Other Method Sometimes Used in Concert with Other Methods
Kron	Kron and TCL	FTP, TFTP, SCP,…	When EEM Is Not Available
DHCP	DHCP	Cisco Network Registrar, TFTP	Agnostic of Access Technology Partially Standardized, Multiple Options Used
CNS	CNS Config Agent CNS Image Agent CNS Inventory Agent CNS Event Agent	Cisco Configuration Engine	Most Secure and Robust Agnostic of Access Technology Agnostic of IP Addressing

Zero-Touch Deployment = Embedded Agents + External Mediation

Deployment & Activation

Example: Zero-Touch Deployment – 1/3

- **Problem**: A large number of Teleworker Routers have to be deployed. Access Technology and Service Provider vary; IP Addressing is not known in advance

- **Solution**: Pre-Configure Routers with a generic boostrap config
This config ensures initial IP connectivity, identifies the device and communicates back to Configuration Engine for appropriate config

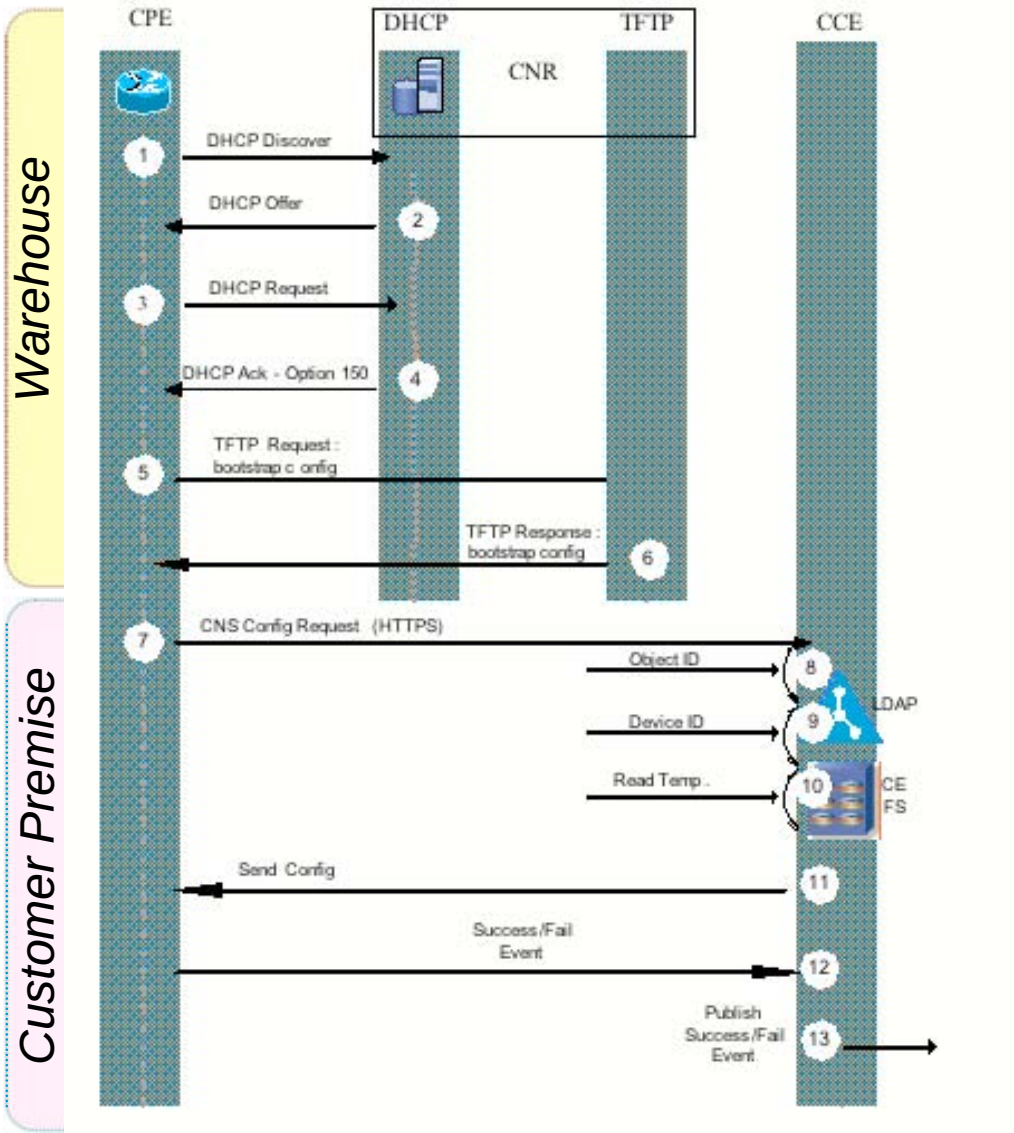
```
Router # cns id hardware-serial
Router # cns config initial MyConfigEngine 80 event no-persist
Router # cns id hardware-serial event
Router # cns event MyConfigEngine 11011
```

Note: Many other options for ID exist and are often used instead of hardware-serial:

```
AMB07300FZX(config)#cns id ?
  Async              Async interface
  BVI                Bridge-Group Virtual Interface
  CTunnel            CTunnel interface
  Dialer             Dialer interface
  Ethernet           IEEE 802.3
  FastEthernet       FastEthernet IEEE 802.3
  Group-Async        Async Group interface
  Loopback           Loopback interface
  MFR                Multilink Frame Relay bundle interface
  Multilink          Multilink-group interface
  Tunnel             Tunnel interface
  Vif                PGM Multicast Host interface
  Virtual-PPP        Virtual PPP interface
  Virtual-Template   Virtual Template interface
  Virtual-TokenRing  Virtual TokenRing
  hardware-serial    Use hardware serial number as unique ID
  hostname           Use hostname as unique ID
  string             Use an arbitrary string as the unique ID
```

Deployment & Activation

Example: Zero-Touch Deployment – 2/3



1. CPE sends DHCP Discover
2. DHCP Server replies with Offer
3. CPE sends DHCP Request
4. DHCP Server replies with option 150
5. CPE requests **`bootstrap-confg`** file via TFTP
6. TFTP server sends CPE **`bootstrap-config`** file

⇒ CPE is shipped to Customer Site

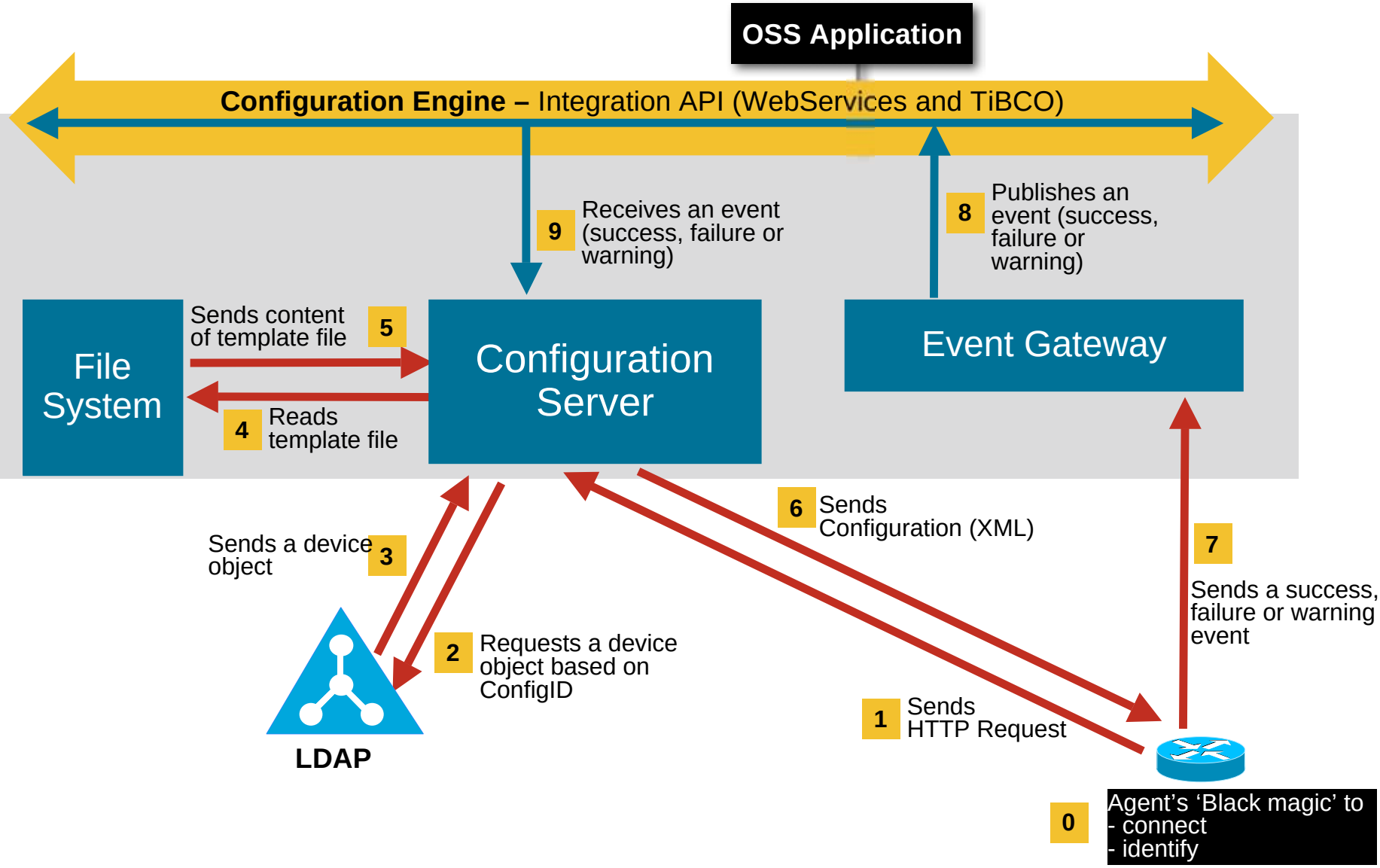
⇒ Customer Order linked to CPE ID

7. CPE sends HTTP request to CNS-CE
8. CNS-CE verifies object ID
9. CNS-CE verifies Device ID
10. CNS-CE reads template from File System
11. CNS-CE sends Config (= template + parameters from LDAP)
12. Successful event
13. Publish success event

Solution Tested

Deployment & Activation

Example: Zero-Touch Deployment – 3/3



Deployment & Activation

XML Programmatic Interface is ... – 1/3

An XML Interface to a Cisco IOS Network Element, for customers and partners needing to remotely adapt and control the behavior of Cisco devices.

XML-PI provides unambiguous and robust information access without the complexity and expense of screen-scraping technologies or external mediation software.

Deployment & Activation

XML Programmatic Interface is ... – 2/3

- XML-PI runs on top of NETCONF and SSH V2 to send and receive CLI commands through a reliable stack without screen scraping or expect scripts
- XML-PI and NETCONF is currently being implemented on many major Cisco platforms
- Devices can have their running configuration changed
- Applications can retrieve the current running configuration
- NETCONF uses XML-based data encoding for the configuration data and protocol messages
- NETCONF runs over SSH and BEEP

Deployment & Activation

XML Programmatic Interface is ... – 3/3

Retrieve the running config

Change the running config

Run a “show” command

Deployment & Activation

Example: Edit the running config

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="3"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <edit-config>
  <target><running/></target>
  <config>
  <xml-config-data>
         <Device-Configuration>
         <ip>
                  <host>
                           <NameHost>
                                    valhalla
                           </NameHost>
                           <HostIPAddress>
                                    10.2.3.5
                           </HostIPAddress>
                  </host>
         </ip>
          </Device-Configuration>
  </xml-config-data>
  </config>
  </edit-config>
</rpc>]]>]]>
```

Request

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply message-id="3" xmlns="urn:ietf:params:netconf:base:1.0">
         <ok/>
</rpc-reply>
```

Response

Deployment & Activation

XML PI – Why do we care ?

- IETF standard-based configuration management
- Provides reliable and secure transport of configurations over encrypted TCP connections
- Improves the speed of configuration changes since it is not limited to console speeds
- Eliminates scripting and “screen scraping” via telnet
- Allows concurrent configuration changes
- Leverages the vast number of XML tools available
- Foundation for future XML configuration capabilities

Multiple Devices and Scripting

Deployment & Activation

What is Enhanced Device Interface (E-DI) ?

- E-DI is:

  An extension to the network device's interface

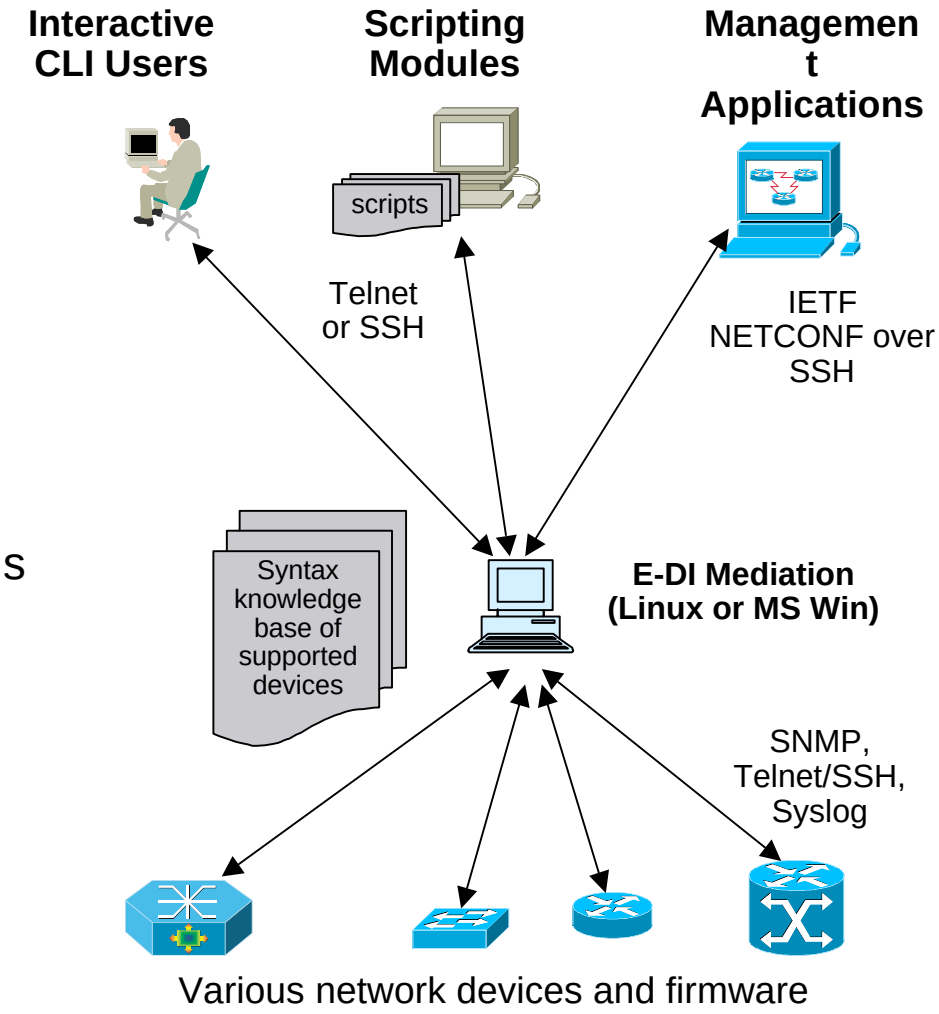
  Complemetary to EMS/NMS

- E-DI provides three interfaces

  1. Enhanced Command Line Interface (CLI) to human users
  2. Perl Scripting Interface and platform for scripting applications
  3. XML programmatic interface to management applications

     IETF NETCONF draft 5 compliant

     Codeployment with Cisco IOS XML PI



Deployment & Activation

E-DI IOS-like Command Line Interface

- Real-time syntax validation and visual feedback

192.168.2.50 - PuTTY

```
admin@jahi-es-50[network]# sh dev
admin@jahi-es-50[network]# sh devices
Number of devices in network: 21

 IP Address      Name                  Type                Vendor   Status
 --------------- --------------------- ------------------- -------- --------
 192.168.1.5     accesssw_1_5          Cat355024           Cisco    P3-alarm
 192.168.1.10    termaccessrtr_1_10    CiscoAS2511RJ       Cisco    normal
 192.168.2.2     JahiTestRtr-1         Cisco2621           Cisco    P3-alarm
 192.168.2.4     rtr_2_4               Cisco831            Cisco    P5-alarm
 192.168.2.5     sw-2-5                Cat355024           Cisco    P3-alarm
 192.168.2.173   Jahi-                 CiscoAP1210         Cisco    P2-alarm
 192.168.2.204   ap-lita               CiscoAP1100         Cisco    P3-alarm
 192.168.3.1     accessrtr_1_1         Cisco2621           Cisco    normal
 192.168.3.3     rtr-3-3               Cisco7505           Cisco    P3-alarm
 192.168.3.6     sw_3_6                Cat2924XL           Cisco    P3-alarm
 192.168.3.7     sw_3_7                Cat2924XL           Cisco    P3-alarm
 192.168.3.8     sw-3-8                Cat37xxStack        Cisco    P2-alarm
 192.168.3.20    rtr-3-20              Cisco3640           Cisco    P2-alarm
 192.168.3.203   ap-3-203              CiscoAP350IOS       Cisco    offline
 192.168.3.204   ap-3-204              CiscoAP350IOS       Cisco    offline
 192.168.3.206   ap-3-206              CiscoAP350IOS       Cisco    normal
 192.168.3.207   ap-3-207              CiscoAP1210         Cisco    normal
 192.168.3.208   ap-3-208              CiscoAP1100         Cisco    normal
 192.168.16.5    sw-3-5                Cat355024           Cisco    P2-alarm
 192.168.16.15   ap-16-15              CiscoAP1100         Cisco    P5-alarm
 192.168.16.16   ap-16-16              CiscoAP1100         Cisco    P5-alarm

admin@jahi-es-50[network]#
```

192.168.2.50 - PuTTY

```
admin@jahi-es-50[MyGroup]# net 192.168.16.5 192.
You are now in network view.
Your present working directory: /network/groups/

admin@jahi-es-50[MyGroup]# sh cdp neighbors

 Local-Dev       Local-If Neighbor          Nei
                          ID/IPAddress      If
 --------------- -------- ----------------- ---
 192.168.3.206   Fa0      192.168.3.5       Fa0
 192.168.3.207   Fa0      192.168.3.5       Fa0
 192.168.16.5    Fa0/1    192.168.3.1       Eth
 192.168.16.5    Fa0/2    192.168.3.6       Fa0
 192.168.16.5    Fa0/2    192.168.3.6       Fa0
 192.168.16.5    Fa0/3    192.168.3.206     Fa0
 192.168.16.5    Fa0/4    192.168.3.208     Fa0
 192.168.16.5    Fa0/7    192.168.3.203     Fa0
 192.168.16.5    Fa0/10   192.168.3.8
 192.168.16.5    Fa0/12   10.0.0.1
 192.168.16.5    Fa0/13   192.168.16.16
 192.168.16.5    Fa0/14   192.168.16.15
 192.168.16.5    Fa0/15   192.168.16.1
 192.168.16.5    Fa0/19   192.168.3.204
 192.168.16.5    Fa0/20   192.168.3.207
 192.168.16.5    Fa0/24   192.168.3.6

admin@jahi-es-50[MyGroup]#
admin@jahi-es-50[MyGroup]#
```

E-DI - 172.19.103.125

```
admin@EDI-server[GRP:~/Routers/]# sh run | include sla
[172.19.103.67] running-config ------------->
[172.19.103.68] running-config ------------->
[172.19.103.69] running-config ------------->
[172.19.103.85] running-config ------------->
      ip sla 7
      ip sla schedule 7 life forever start-time now

admin@EDI-server[GRP:~/Routers/]#
```

Deployment & Activation

Example: E-DI Perl Scripting

- **Problem:** Finding and grouping devices that match multiple criteria (such as IOS Version, specific config commands, group membership, etc) is a common task
- **Solution:** Write a script to automate this task

Config Command | Output group

```
admin@edi-jms-6[SVR:/server/perl-samples]# perl GroupByCfgMatch.pl Routers 12.3 archive myGroup
 Added device 172.25.86.110 to group tmpGroup
 Added device 172.25.86.143 to group tmpGroup
 Added device 172.25.86.144 to group tmpGroup
 Added device 172.25.86.166 to group tmpGroup
!Status Code: 0
```

Script | Input group | IOS Version

The Resulting Group of Devices:

```
admin@edi-jms-6[SVR:/server]# show devices group myGroup
Number of devices in myGroup group: 4
  Device          Name                    Type                    Status
- --------------- --------------------- --------------------- --------
  172.25.86.110   issc-1760-1           Cisco1760             P3-alarm
  172.25.86.143   issc-831-1            Cisco831              P3-alarm
  172.25.86.144   issc-1841-1           Cisco1841             P5-alarm
  172.25.86.166   issc-2811-1.yourdoma  Cisco2811             P3-alarm
                  in.com
!Status Code: 0
admin@edi-jms-6[SVR:/server]#
```

The full Script can be found in Appendix II

Deployment & Activation

New in Enhanced Device Interface 2.2

New Feature	Description
Linux / Windows	Support for server and client apps on Linux and Windows
IDU	Incremental Device Support
Operational Data Model	XML interface for the show commands from NEs
Macro CLI Commands	- Define consistent Macros for a set of commands across various OS versions - CLI and GUI Interface for Macro CLI configuration - Provision the Network using Macro Grouping capability
Command Modeler and Analyzer	- IDE over the EDI Device CLI KB. - Analyze Commands across Device/OS. - Model Based Configs can be created using this.

Free of Charge Download from:

https://upload.cisco.com/cgi-bin/swc/fileexg/main.cgi?CONTYPES=ccu-forum

(easier to remember url: http://tinyurl.com/2jrtrr)

Agenda

Introduction & Overview

Service Planning

[Coffee Break]

Service Deployment & Activation

[Lunch Break]

➲ Service Testing, Verification & Assurance

[Coffee Break]

Troubleshooting & Optimization

Summary

Testing, Verification & Assurance

Two Types of Questions

- **Is it working ?** **Testing and Verification**

  Verify planning and design assumptions were valid

  Ensure Deployment & Activation Phase was successful

  Proactively eliminate well-known potential problems

- **Are we meeting SLA ?** **Service Assurance**

  Ensure business objectives and service level agreements are met on an ongoing basis

  Proactively mitigate well-known potential incidents

Testing, Verification & Assurance

Two Types of Connectivity

- **Connectivity, Yes/No** **Testing and Verification**

  If the user can reach the IP endpoint the service is available

  Can be calculated using basic availability equation

$$Availability = 1 - \frac{[\text{Probes with No Response}]}{[\text{Total Probes Sent}]}$$

- **Bounded Criteria Connectivity** **Service Assurance**

  The user can reach the IP endpoint within some bounded criteria agreed upon between the service provider and customer

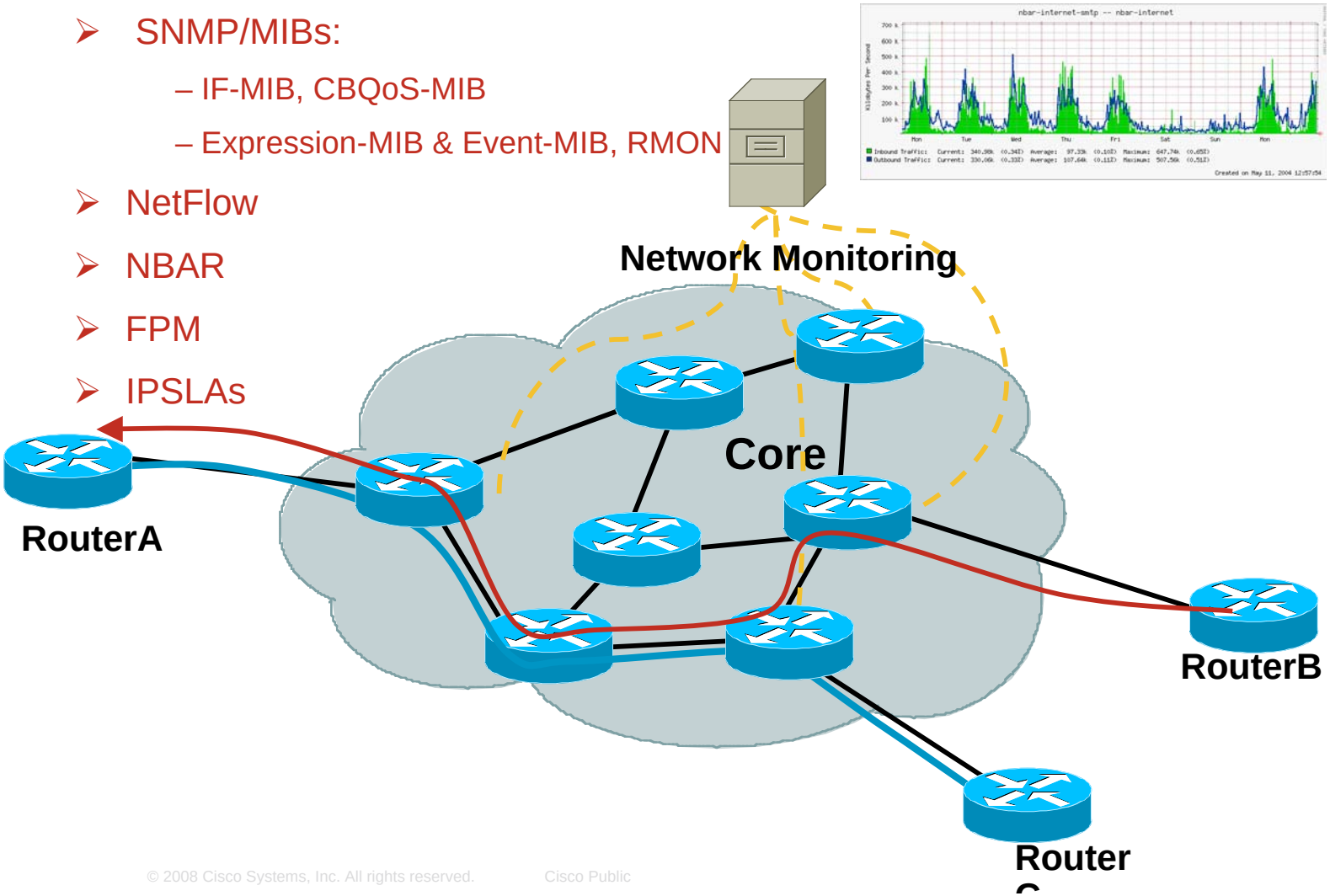
  Connectivity is a prerequisite for bounded crieria connectivity

Testing, Verification & Assurance

Verify (bounded criteria) Connectivity

Proposal:

- CLI
- SNMP/MIBs:
  - IF-MIB, CBQoS-MIB
  - Expression-MIB & Event-MIB, RMON
- NetFlow
- NBAR
- FPM
- IPSLAs



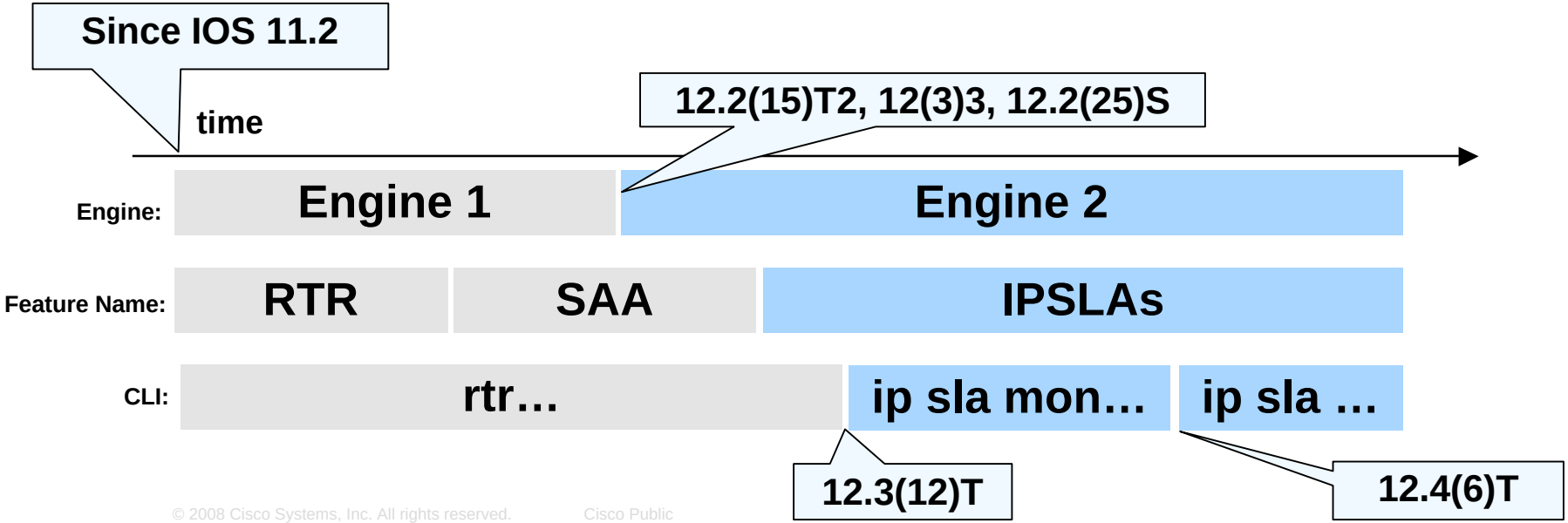
Testing, Verification & Assurance

IPSLA – Introduction 2/2

- Cisco IOS feature available on most platforms
- Measure Delay, Jitter, Loss Probability

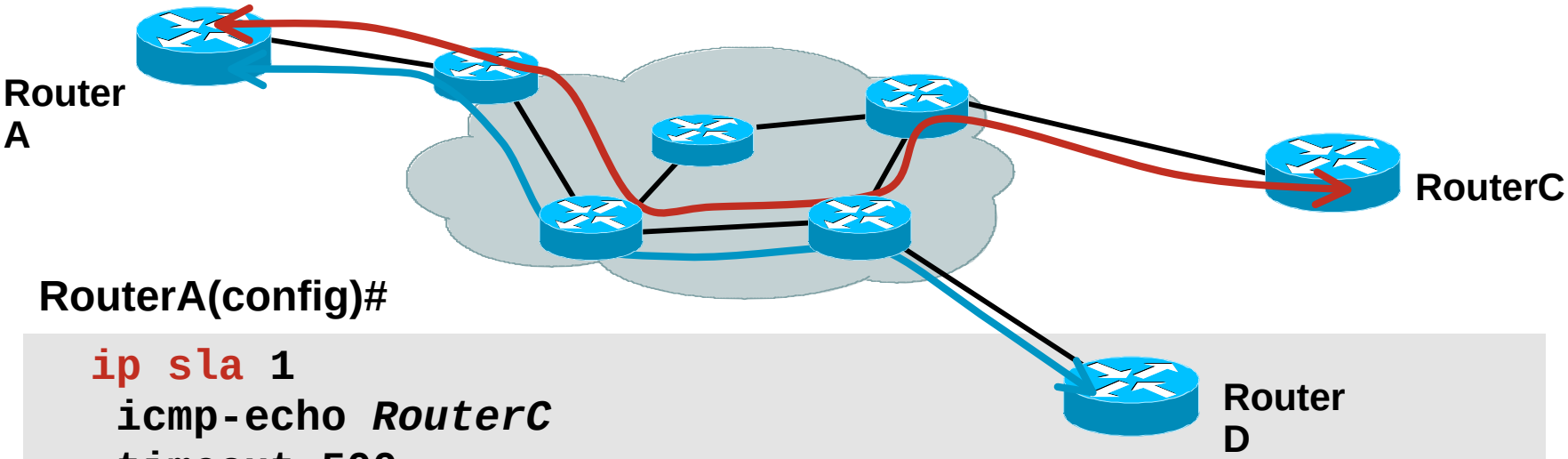
Accessible via CLI and SNMP (CISCO-RTTMON-MIB)

- IPSLAs responder and ICMP echo probe were available within IP Base in 12.4(6)T and above
- IPSLAs functionality is available in IPVoice and above packages
- In 12.3T a customer can still obtain the old package types and use IPSLAs
- As of 12.4T the old packages have been removed



Testing, Verification & Assurance

IPSLA – ICMP and UDP Jitter Examples



RouterA(config)#

```
ip sla 1
 icmp-echo RouterC
 timeout 500
 frequency 10
ip sla monitor schedule 1 start-time now
```

```
ip sla 10
 udp-jitter RouterD 16384 num-packets 1000 interval 20
 request-data-size 172
 tos 20
 frequency 60
ip sla schedule 10 start-time now
```

Testing, Verification & Assurance

IPSLA – ICMP Echo Operation

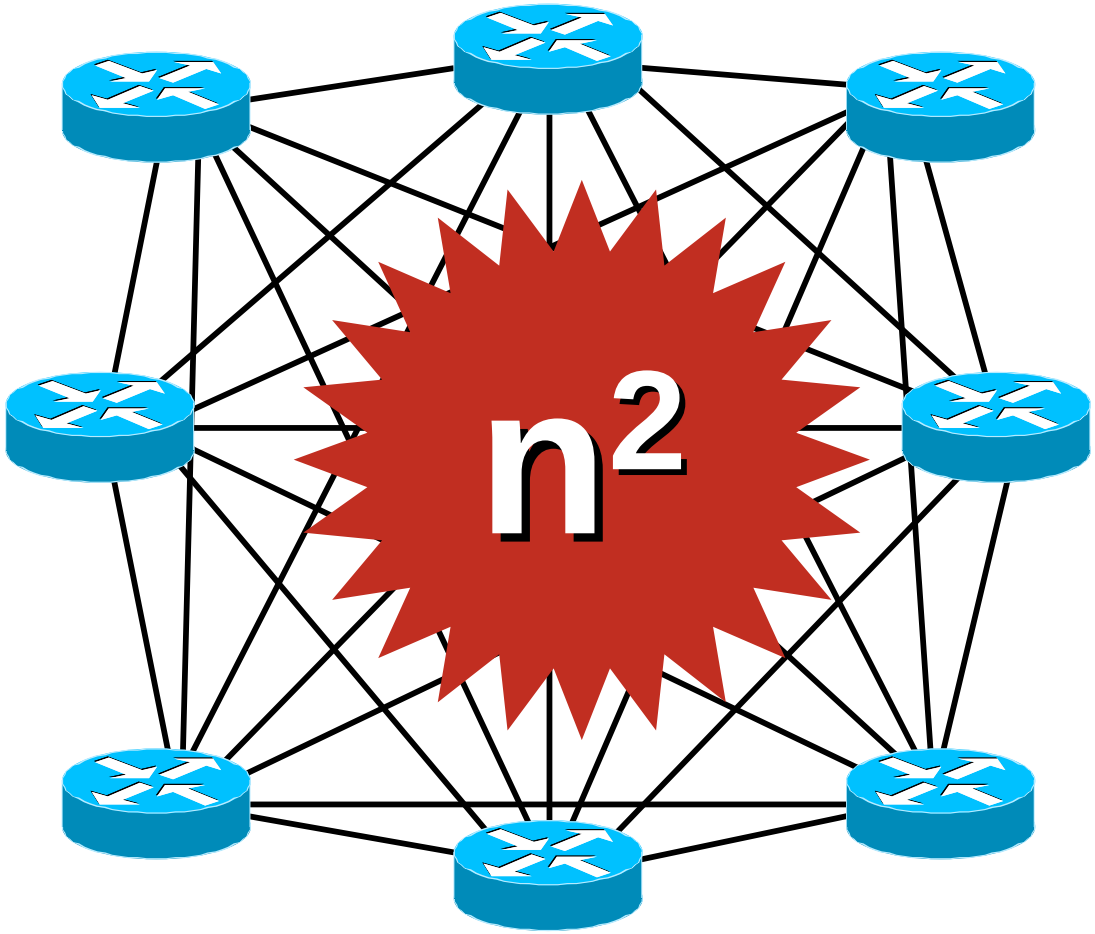
```
Router#show ip sla sta mon 1
Round trip time (RTT)   Index 1
         Latest RTT: 1 ms
Latest operation start time: *05:26:00.226 UTC Fri Jan 4 2008
Latest operation return code: OK
Number of successes: 1
 Number of failures: 0
Operation time to live: 188 sec

Router#sh ip sla mo sta 1 detail
Round trip time (RTT)   Index 1
         Latest RTT: 1 ms
Latest operation start time: *05:26:30.224 UTC Fri Jan 4 2008
Latest operation return code: OK
Over thresholds occurred: FALSE
Number of successes: 2
Number of failures: 0
Operation time to live: 155 sec
Operational state of entry: Active
Last time this entry was reset: Never
```

Testing, Verification & Assurance

IPSLA – UDP Jitter Operation

```
Router#sh ip sla statistics 10
Round trip time (RTT)   Index 10
         Latest RTT: 1 ms
Latest operation start time: *05:43:28.720 UTC Fri Jan 4  2008
Latest operation return code: OK RTT Values
         Number Of RTT: 10
         RTT Min/Avg/Max: 1/1/1 ms
Latency one-way time milliseconds
         Number of one-way Samples: 0
         Source to Destination one way Min/Avg/Max: 0/0/0 ms
         Desination to source one way Min/Avg/Max: 0/0/0 ms
Jitter time milliseconds
         Number of Jitter Samples: 9
         Source to Destination Jitter Min/Avg/Max: 20/20/23 ms
         Destination to Source Jitter Min/Avg/Max: 22/21/24 ms
Packet Loss Values
         Loss Source to Destination: 0           Loss Destination to
Source: 0
         Out Of Sequence: 0      Tail Drop: 0    Packet Late
Arrival: 0
Number of successes: 1
Number of failures: 0
Operation time to live: 3567 sec
```

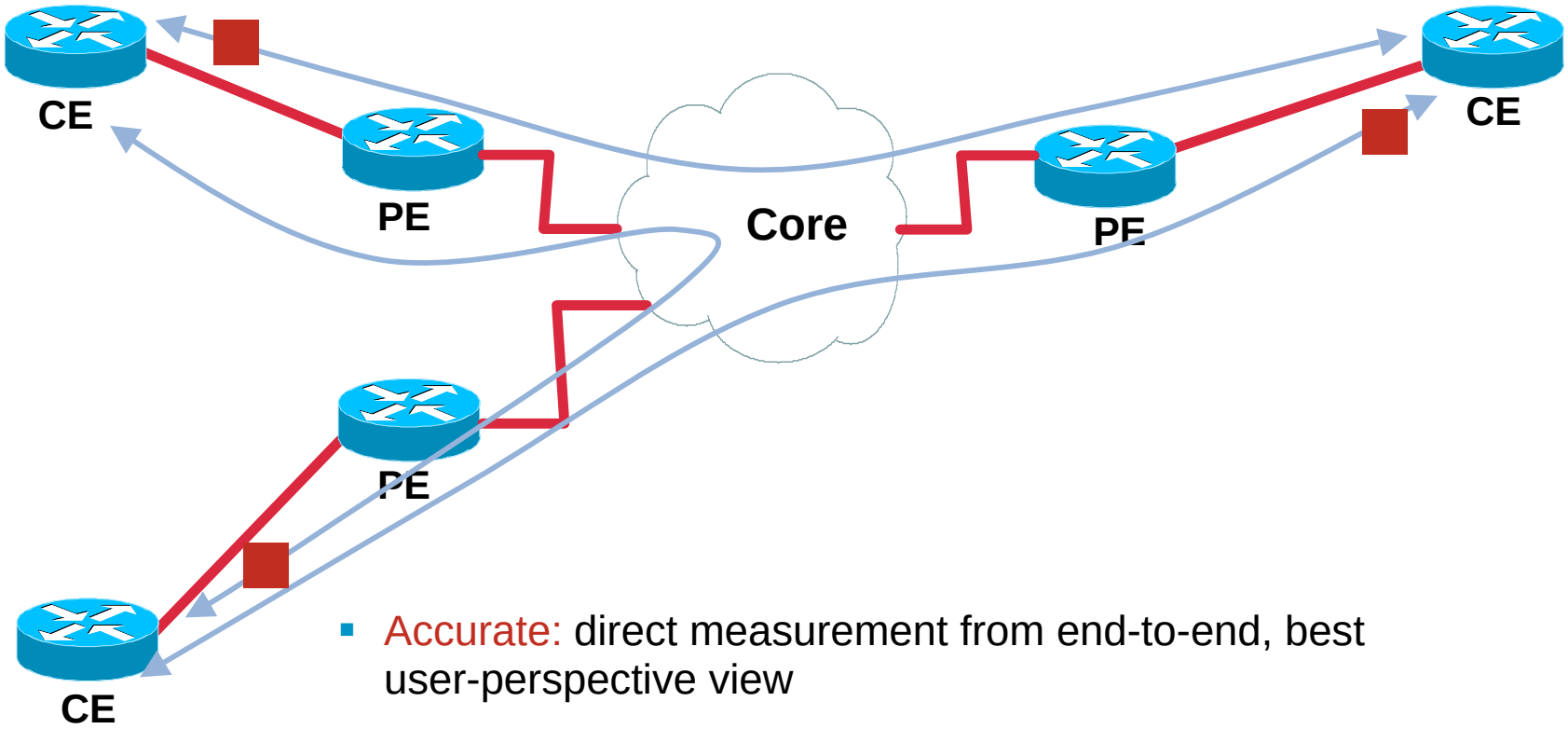
Full Mesh



Nodes	Operation
2	1
3	3
4	6
5	10
6	15
7	21
8	28
…	…
100	4950

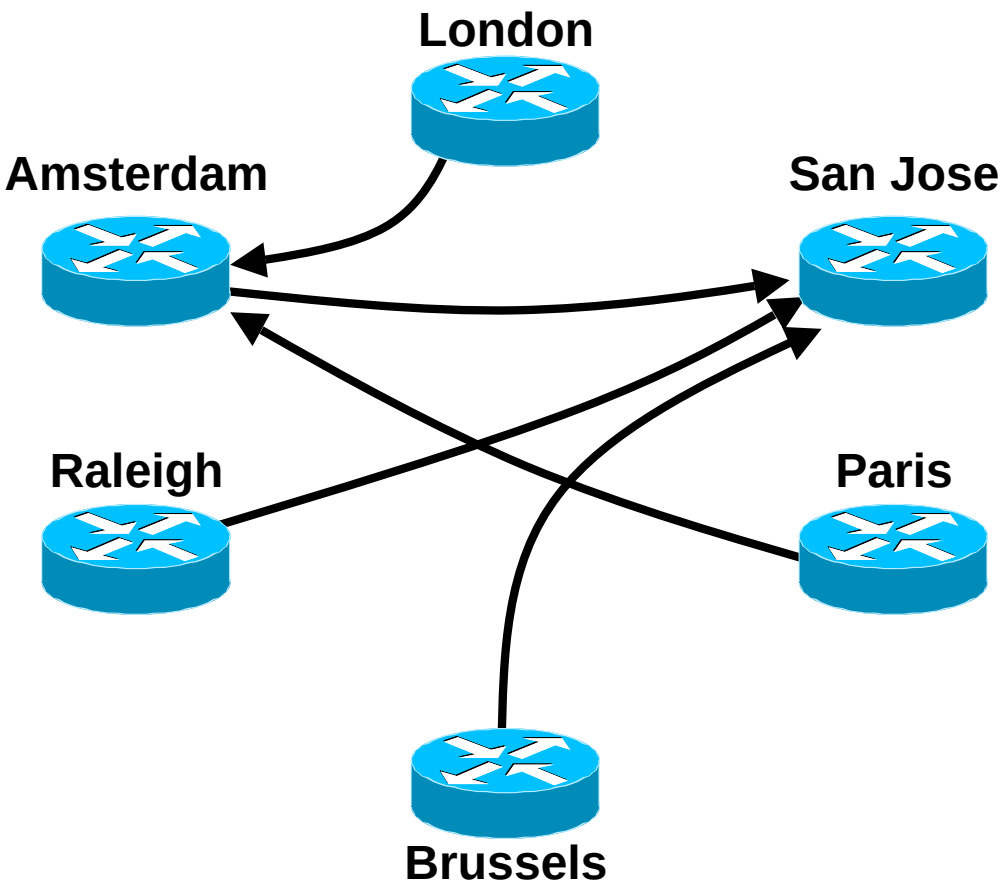
- **Number of operations is proportional to the square of the number of nodes**
- **Does not scale**

Full Mesh CE-to-CE [Example]



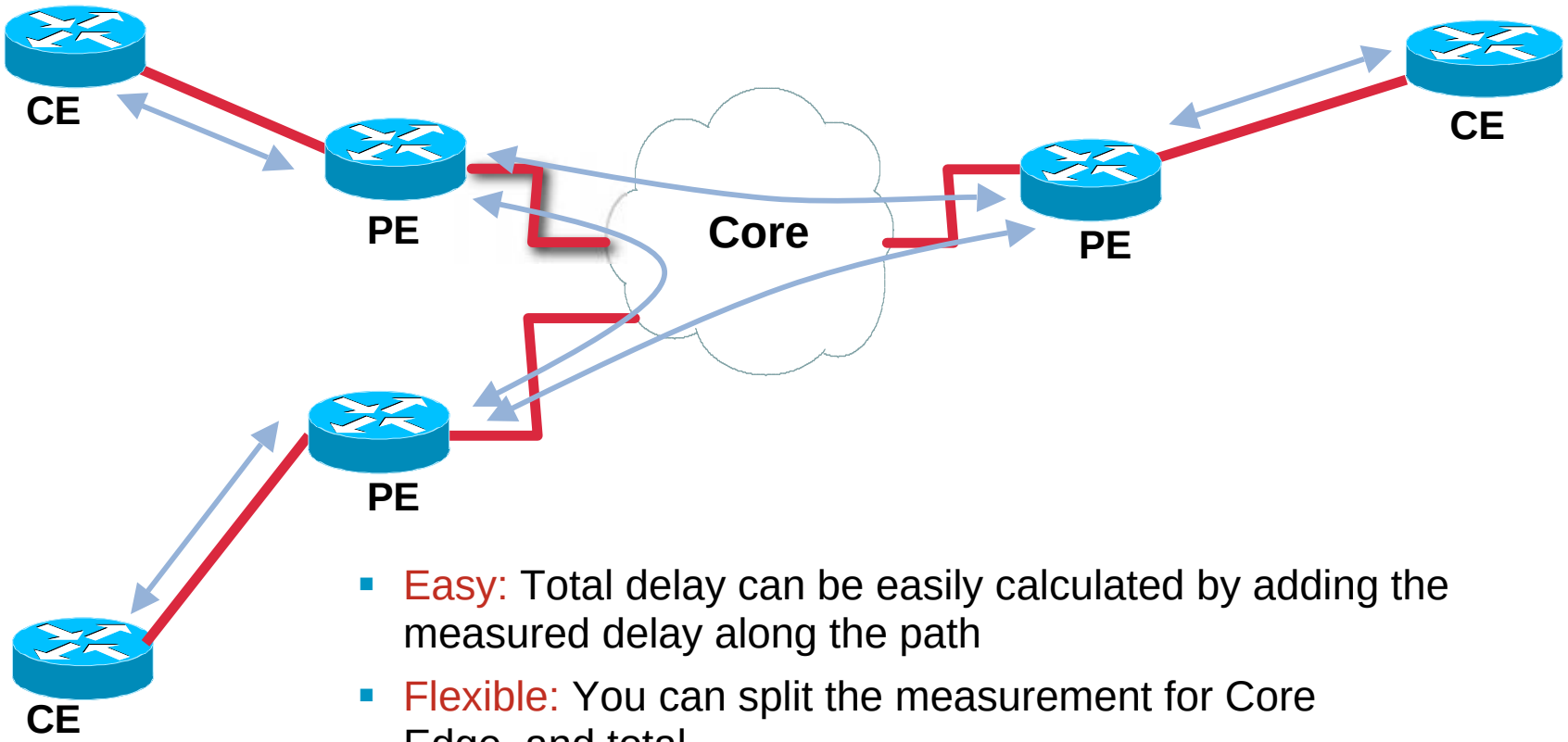
- Accurate: direct measurement from end-to-end, best user-perspective view
- Expensive: for n nodes, requires n(n-1)/2 operations
- In certain cases, it might be difficult to poll the results with SNMP on the CE

Partial Mesh



- Full mesh is not always desirable
- Select only critical path, like branch offices to headquarters
- Dramatically reduces the number of probes

Composite SLA for Delay [Example]



- Easy: Total delay can be easily calculated by adding the measured delay along the path
- Flexible: You can split the measurement for Core Edge, and total
- Measurements are less accurate, as each measurement carry its own error tolerance (typically ± 1 ms per measurement)

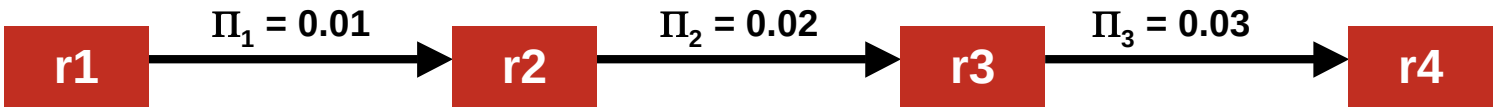
Composite SLA for Packet Drop [1/2]

- A trivial solution might is to consider the sum of drop probabilities; this is conservative
- A more accurate approach is to invert the probability of a successful packet delivery
- If Π_x is the loss probability across section x, then the total loss probability is:

$$\boxed{\prod_{1\ldots x} = 1 - [(1 - \prod_1).(1 - \prod_2)\cdots(1 - \prod_n)]}$$

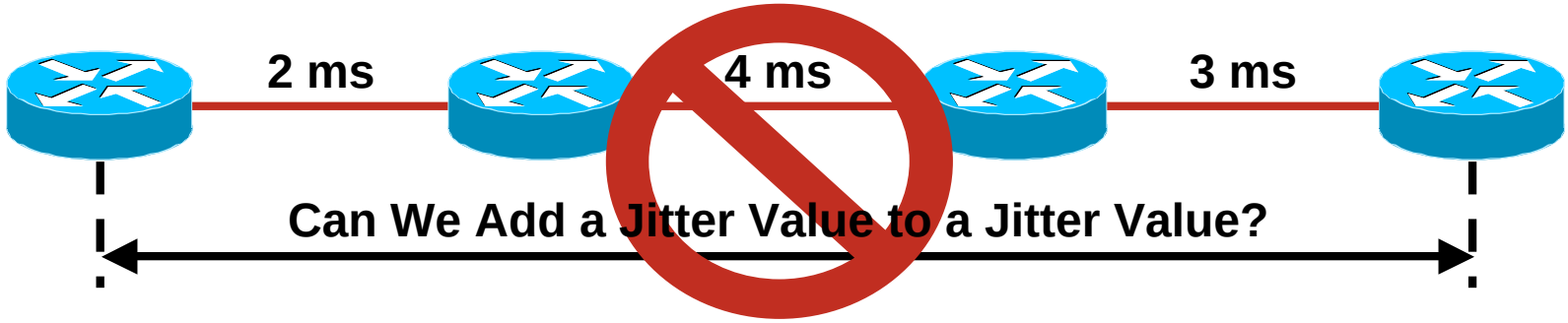
Composite SLA for Packet Drop [2/2]

Example: We Have Three Sections with Various Drop Probabilities:



- First solution:
  0.01+0.02+0.03=0.06 (6%)
- Second solution:
  1-[(1-0.01).(1-0.02).(1-0.03)]=0.058906 (5.8%)

Composite SLA for Jitter



- Short answer: NO!
- This is not a valid approach to calculate total jitter based on measured jitter
  (jitter is not additive)
- Too many factors: positive jitter, negative jitter, percentile-95 of jitter, average jitter,…
- You'd better measure it, not calculate it

Testing, Verification & Assurance

IPSLA – Multiple Operations Scheduling

- Operations of the same type and same frequency should be used with IPSLA multiple operations scheduling:
  - Notion of group, it lets you start many operations at once
  - Reduced load on the network
  - If you do not specify a frequency, the default frequency will be the same as that of the schedule period)
- Example, start operations 1 to 3 within the next 20 seconds

```
Router (config)# ip sla 1
Router (config)# icmp-echo RouterC
Router (config)# ip sla 2
Router (config)# icmp-echo RouterD
Router (config)# ip sla 3
Router (config)# icmp-echo RouterE

Router (config)#ip sla group schedule 1 1-3 sch 20 start now

Router #show ip sla group schedule
```

***12.3(8)T**

Testing, Verification & Assurance

IPSLA – Recurring Scheduling

- You can schedule a single IPSLAs operation to start automatically at a specified time and for a specified duration every day:

  The life value for a recurring IPSLAs operation should be less than one day.

  The ageout value for a recurring operation must be "never" (which is specified with the value 0, this is the value by default), or the sum of the life and ageout values must be more than one day.

- Example:

```
Router(config)# ip sla  schedule 5 start-time 12:00:10
  life 3600 recurring
```

***12.3(8)T**

Testing, Verification & Assurance

IPSLA – Random Scheduling

Problem: Strictly periodically starting IPSLA operations might be subject to ‘synchronization effects’ with other processes (ie. routing updates), leading to inaccurate data.

Solution: Use IPSLA Random Scheduling to randomize start time

This example starts operation 1 to 3 within the next 44 seconds, and each operation will have a random frequency varying between 10 and 15 seconds:

```
Router(config)#ip sla group schedule 1 1-3 schedule-period 44 frequency range
  10-15 start-time now life forever


Router#sh ip sla op | i start
Latest operation start time: *12:56:12.243 PST Fri Jan 4 2008
Latest operation start time: *12:56:06.323 PST Fri Jan 4 2008
Latest operation start time: *12:56:07.743 PST Fri Jan 4 2008
router#sh ip sla op | i start
Latest operation start time: *13:00:19.423 PST Fri Jan 4 2008
Latest operation start time: *13:00:15.895 PST Fri Jan 4 2008
Latest operation start time: *13:00:21.015 PST Fri Jan 4 2008
```

***12.4(2)T**

Testing, Verification & Assurance

IPSLA – Reaction Configuration

```
RouterA(config)#
ip sla 20
 icmp-echo ServerB
 frequency 10
ip sla reaction-configuration 20 react timeout threshold-type consecutive 3
  action-type trapAndTrigger
ip sla schedule 20 life forever start-time now
ip sla reaction-trigger 20 30
ip sla 30
 icmp-echo ServerC
 frequency 20
ip sla schedule 30 start-time pending
ip sla reaction-configuration 30 react timeout threshold-type immediate
  action-type traponly

logging on
ip sla logging trap
snmp-server host nms_server version 2c public
snmp-server enable traps syslog
```

Testing, Verification & Assurance

IPSLA and Enhanced Object Tracking

- The Enhanced Object Tracking(EOT)* feature separates the tracking mechanism from the protocol and creates a separate standalone tracking process that can be used by any other process
- Subset of the EOT Cisco IOS feature:

  Track the output from the IP SLAs objects and use the provided information to trigger an action
- Aspects of an IPSLAs operations which can be tracked:
  - state
  - reachability

http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps5207/products_feature_guide09186a00801d2d74.html

***12.3(4)T and 12.2(25)S**

Service Testing, Verification and Assurance

Example: Track Server Reachability

IP SLA

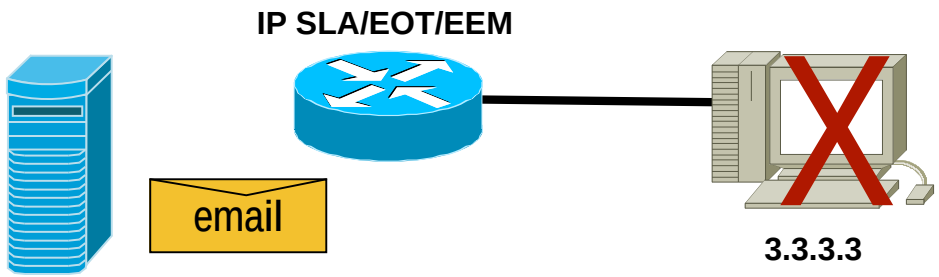
```
ip sla 10
 icmp-echo 3.3.3.3
 timeout 500
 frequency 3
ip sla schedule 10 life forever start-time now
```

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!"
```

How Much Bandwidth is Utilized?

Testing, Verification & Assurance

Simple Bandwidth Utilization – on the CLI

```
Router#sh int FastEthernet0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is Gt96k FE, address is 000b.fdc9.a640 (bia 000b.fdc9.a640)
  Internet address is 10.48.71.24/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, 100BaseTX/FX
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/7/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 8000 bits/sec, 7 packets/sec
  5 minute output rate 2000 bits/sec, 3 packet
     5335540 packets input, 698590034 bytes
     Received 4871407 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog
     0 input packets with dribble condition detected
     656702 packets output, 111094753 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

ifSpeed

Input Utilization(bps)

Output Utilization(bps)

ifInOctects(Δ)

IfOutOctets (Δ)

Testing, Verification & Assurance

Simple Bandwidth Utilization – via SNMP

- **ifInOctets(ifHCInOctets)** — Total number of octets received on the interface, including framing characters
- **ifOutOctets(ifHCOutOctets)** — Total number of octets transmitted out of the interface, including framing characters
- **ifSpeed** — An estimate of the interface's current bandwidth in bits per second; for interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth

$$InputUtilization(bits/\sec) = \frac{[(\Delta(ifInOctets)) \times 8 \times 100]}{[(\Delta \sec) \times ifSpeed]}$$

$$OutputUtilization(bits/\sec) = \frac{[(\Delta(ifOutOctets)) \times 8 \times 100]}{[(\Delta \sec) \times ifSpeed]}$$

How To Identify Applications?

Testing, Verification & Assurance

How To Identify Applications?

Application/Protocol	How to Identify?
VoIP	UDP TOS = 5
IPVC	TOS = 4
H.323	TCP Port = 1719 , 1720 and TOS = 3
IPv6 Multicast	Format Prefix (FP) = 1111 1111
VOD	TCP Port 507

- L3 and L4 Access Control Lists:

  Identifies protocols based on IP address, protocol type and port number

- NetFlow

  Provides statistics such as traffic volume details (packets, bytes) and time information (start/stop timestamp, duration)

- Classify Network Traffic into Traffic Classes (QoS):

  Allows Accounting per class-of-service

(Cont.)

How To Identify Applications – ACL

Layer 3 and Layer 4 ACL

Protocol Type

Protocol Port Number

```
access-list 103 permit tcp any host 192.168.1.1 eq telnet (23)
access-list 103 permit tcp any 192.168.1.0 0.0.0.255 eq ftp (20)
access-list 103 permit tcp any 192.168.1.0 0.0.0.255 eq www (80)
access-list 103 deny   ip any any log
```

Server/Network IP Address

```
interface ethernet 0
 ip access-group 103 in
```

Apply acl to the Interface: In/Out

Testing, Verification & Assurance

How To Identify Applications – NetFlow

NetFlow v5 (“classic”)

Packet Sizes

```
Router# show ip cache flow
IP packet size distribution (85435 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448  480
   .000 .000 .125 .125 .250 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 .500 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  2728 active, 1368 inactive, 85310 added
  463824 ager polls, 0 flow alloc failures
  Active flows timeout in 30 minutes
  Inactive flows timeout in 15 seconds
  last clearing of statistics never
Protocol         Total     Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows      /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
TCP-X                2       0.0         1  1440      0.0       0.0       9.5
TCP-other        82580      11.2         1  1440     11.2       0.0      12.0
Total:           82582      11.2         1  1440     11.2

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Et0/0         132.122.25.60   Se0/0         192.168.1.1     06 9AEE 0007     1
Et0/0         139.57.220.28   Se0/0         192.168.1.1     06 708D 0007     1
Et0/0         165.172.153.65  Se0/0         192.168.1.1     06 CB46 0007     1
```

# of Active Flows

Rates and Duration

Flow Details

How To Identify Applications – CBQoS

Classify Network Traffic into Traffic Classes (QoS):

- Variety of match criteria such as the IP Precedence value, differentiated services code point (DSCP) value, class of service (CoS) value, source and destination Media Access Control (MAC) addresses, input interface, or protocol type

```
Router# configure terminal
Router(config)# class-map myclass
Router(config-cmap)# match fr-dlci 500

Router(config)# policy-map mypolicy
Router(config-pmap)# class myclass
Router(config-pmap-c)# bandwidth percent 50

Router(config)# interface serial4/0
Router(config-if)# service-policy output mypolicy
Router(config-if)# end
```

Define Class-map *(cbQosClassMapCfg)*

(cbQosMatchstmtCfg)

Define Policy-map *(cbQosPoliceCfg)*

Assigned the Policy to the Interface (In or Out)

- Show policy statistics

```
Router#show policy-map interface
```

cbQosPoliceStats
cbQosClassMapStats
cbQosMatchstmtStats

Testing, Verification & Assurance

How To Identify Applications?

- "Well-known" protocols

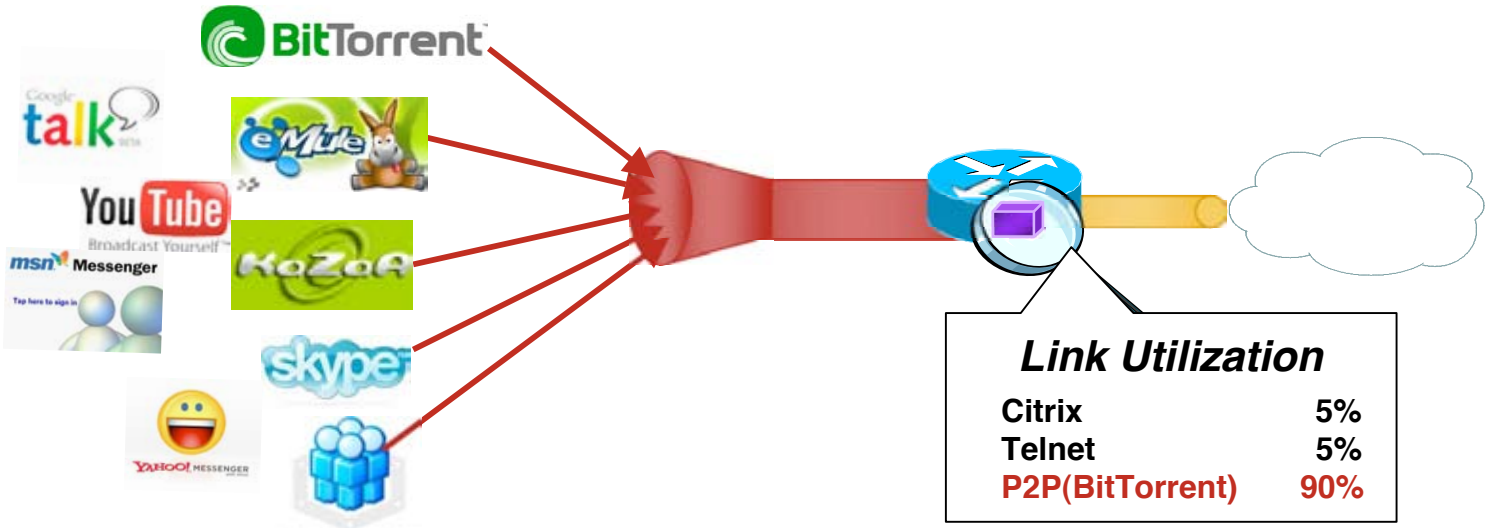
  IP Protocol based Services (Non-UDP/Non-TCP Protocols):

  EGP, ICMP, GRE, IPSec, ...

  UDP and TCP Protocols:

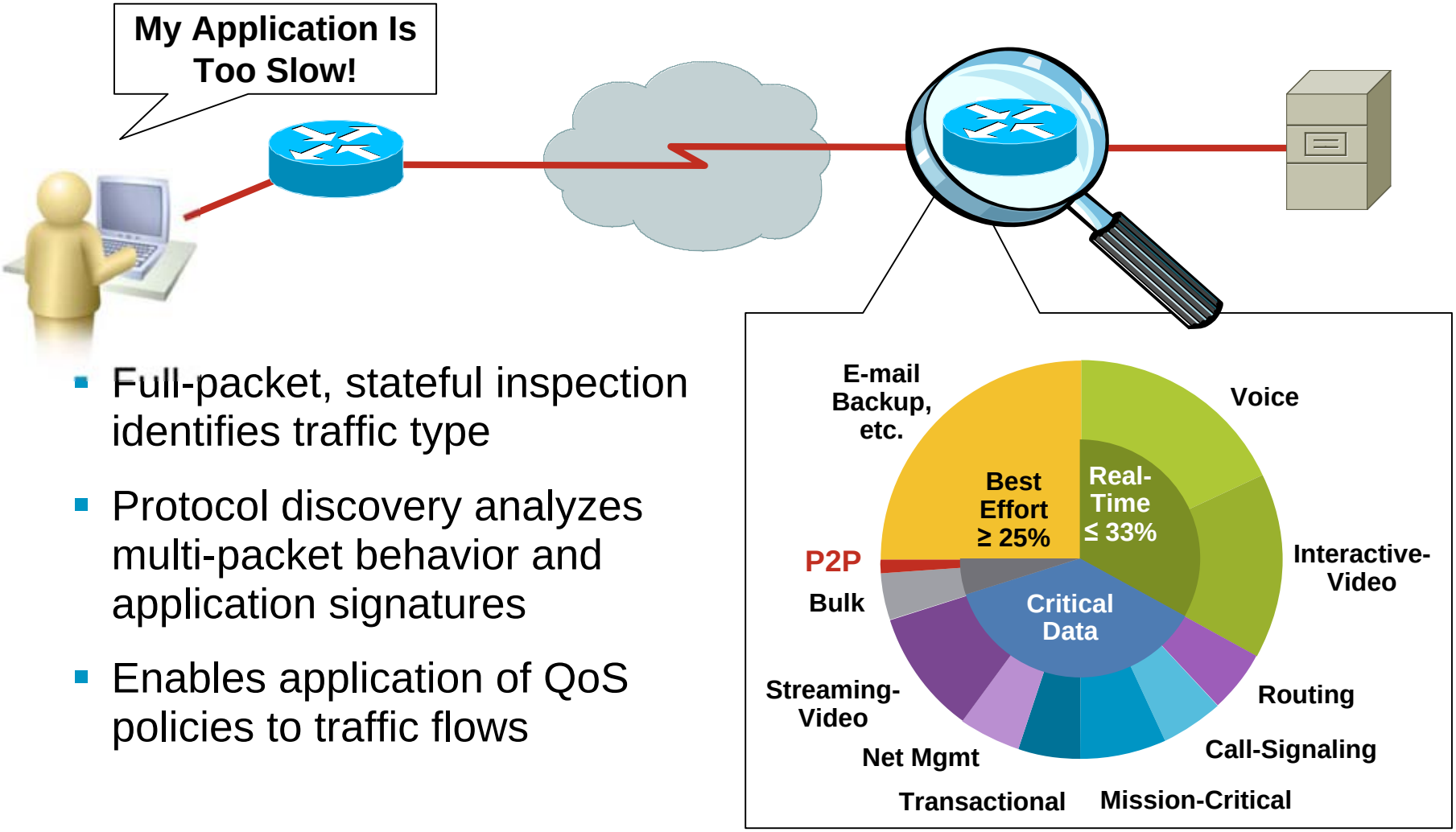
  DNS, Finger, Gopher, http, https, ntp, PCAnywhere, RIP, ...

- But, what about "not well-known" protocols?



Testing, Verification & Assurance

Network Based Application Recognition



- Full-packet, stateful inspection identifies traffic type
- Protocol discovery analyzes multi-packet behavior and application signatures
- Enables application of QoS policies to traffic flows

Link Utilization

Testing, Verification & Assurance

NBAR Principles

- Network-Based Application Recognition classifies traffic by protocol (Layers 4–7)
- Protocol discovery analyzes application traffic patterns in real time and discovers which applications are running on the network
- NBAR supports Cisco IOS QoS features to apply application-level QoS policies
  - Guaranteed bandwidth with Class-based Weighted Fair Queuing (CBWFQ)
  - Policing and limiting bandwidth
  - Marking (ToS or IP DSCP)
  - Drop policy with weighted random early detection (WRED)
- Accounting functionality is provided by the NBAR “protocol discovery” feature

Testing, Verification & Assurance

NBAR Main Supported Platforms

Cisco IOS Release		
12.4T	12.4 Mainline	12.2S
Cisco 800 above 871	Cisco 800 above 831	Cisco 7200
Cisco 1700	Cisco 1700	Cisco 7301
Cisco 1800	Cisco 1800	Cisco 7304-NPE
Cisco 2600XM	Cisco 2600XM	
Cisco 2800	Cisco 2800	
Cisco 3600	Cisco 3600	
Cisco 3700	Cisco 3700	
Cisco 3800	Cisco 3800	
Cisco 7200	Cisco 7200	
Cisco 7301	Cisco 7301	
	Cisco 7500 with VIP2-50 or above	

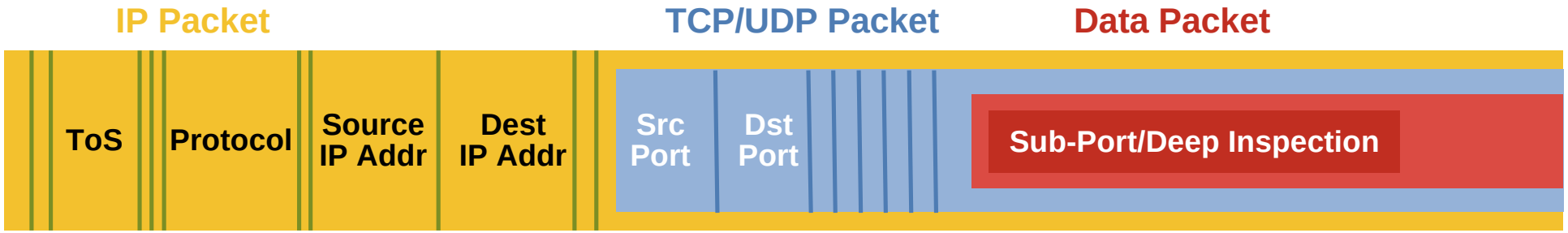
Cisco Catalyst® 6500

- SUP1/SUP1a/SUP2: software-based implementation
- SUP720: SIP-200, FlexWAN and enhanced FlexWAN interfaces (software-based implementation)
- SUP32 PISA. Also supports the enhanced FlexWAN, SIP-200, SIP-400
- Also supported on the Multiprocessor WAN Application Module (MWAM) (6*7200 on a board)

Testing, Verification & Assurance

NBAR Deep Packet Inspection (DPI)

Stateful and Dynamic Inspection



- Identifies over 90 applications and protocols TCP and UDP port numbers
  - Statically assigned
  - Dynamically assigned during connection establishment
- Non-TCP and non-UDP IP protocols
- Data packet inspection for matching values
- Header classification and data packet inspection

Testing, Verification & Assurance

NBAR Two Modes Of Operation

Passive Mode

CISCO-NBAR-PROTOCOL-DISCOVERY-MIB

- Protocol discovery per interface
  - Discovers and provides real time statistics on applications
  - Per-interface, per-protocol, bi-directional statistics:
    - Bit rate (bps), Packet counts and Byte counts

Active Mode

CISCO-CLASS-BASED-QOS-MIB

- Modular QoS traffic Classification
  - NBAR ensures that network bandwidth is used efficiently by QoS features:
    - Guaranteed bandwidth
    - Bandwidth limits
    - Traffic Shaping and Packet coloring

Note: Accounting Functionality Is Provided by “Protocol Discovery” Feature

Testing, Verification & Assurance

NBAR Two Modes Of Operation

Passive Mode

```
Router(config-if)#interface fastethernet 0/0
Router(config-if)#ip nbar protocol-discovery
```

Active Mode

```
Router(config)#class-map [match-any|match all] myProt
Router(config-cmap)#match protocol protocol
```

```
class-map match-any my-video
  match protocol cuseeme
  match protocol h323
  match protocol rtp video
```

Testing, Verification & Assurance

NBAR Protocol Discovery

Passive Mode

- Configure traffic statistics collection for all protocols known to NBAR
- Discover application protocols transiting an interface
- Supports both input and output traffic
- Can be applied independently of a Service Policy (MQC)
- Configuration Command

```
Router(config-if)#ip nbar protocol-discovery
```

- Show Command

```
Router# show ip nbar protocol-discovery [interface
interface-spec][stats {byte-count|bit-rate|packet-
count}][protocol protocol-name| top-n number}]
```

Testing, Verification & Assurance

Passive Mode

Example: NBAR Protocol Discovery

```
router# show ip nbar protocol-discovery interface FastEthernet 6/0

  FastEthernet6/0
                  Input                          Output
  Protocol                Packet Count                   Packet Count
                  Byte Count                             Byte Count
                  5 minute bit rate (bps)        5 minute bit rate (bps)
  -----------     ------------------------       ------------------------
  http            316773                         0
                  26340105                       0
                  3000                           0
  pop3            4437                           7367
                  2301891                        339213
                  3000                           0
  snmp            279538                         14644
                  319106191                      673624
                  0                              0
  ftp             8979                           7714
                  906550                         694260
                  0                              0
  …
  Total           17203819                       151684936
                  19161397327                    50967034611
                  4179000                        6620000
```

Testing, Verification & Assurance

NBAR Top-N Statistics

Passive Mode

```
Router#show ip nbar protocol-discovery top-n 5 Serial0/0
                Input                     Output
Protocol        Packet Count              Packet Count
                Byte Count                Byte Count
                5 minute bit rate (bps)       5 minute bit rate (bps)
  ---------- ------------------------- ------------------------
  custom-01       40565                     40565
                  2596160                   2596160
                  3000                      3000
  telnet          395                       75
                  28539                     6415
                  0                         0
  icmp            101                       100
                  7360                      6860
                  0                         0
  snmp            28                        0
                  1988                      0
                  0                         0
  netbios         9                         0
                  738                       0
                  0                         0
  unknown         205                       204
                  14976                     10404
                  0                         0
  Total           41304                     40944
                  2649809                   2619839
                  3000                      3000
```

Interface Where NBAR PD Is Enabled

- Top-N for all interfaces with NBAR protocol discovery enabled
- NBAR-PD- MIB provides Top-N for all interfaces where N can differ for each interface

Passive Mode

Testing, Verification & Assurance

NBAR Protocol Discovery MIB

Traffic Classification and Real-Time Statistics

- Automatically uses all PDLMs

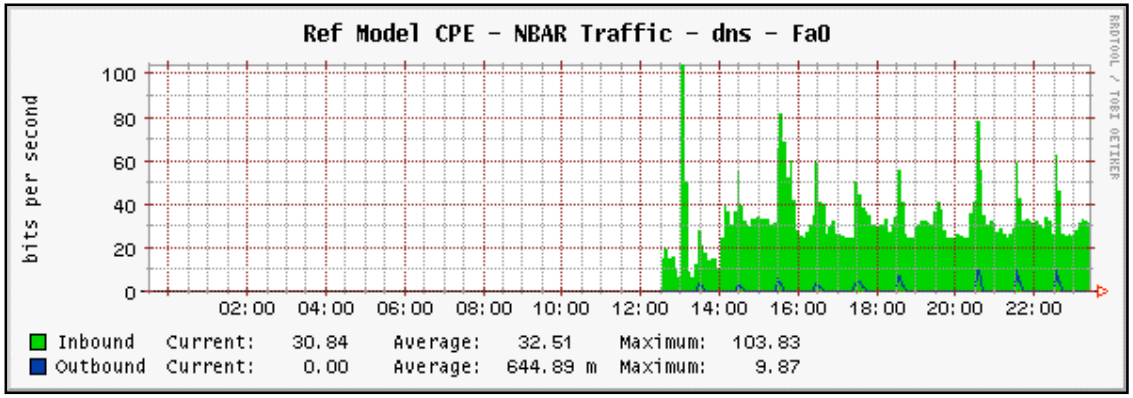
  Run protocol discovery instead of specifying individual protocols

- Provides statistics per application recognized by NBAR via SNMP:

  Bit rate (bps), Packet counts, Byte counts

  Includes statistics for traffic identified with user-defined custom application classification

- Enable or disable protocol discovery per interface
- Configure and view multiple top-n tables listing protocols by bandwidth usage
- Configure thresholds and configure notifications when these thresholds are crossed



Testing, Verification & Assurance

NBAR Defining A Class-Map

Active Mode

```
Router(config)#class-map match-all nbar_test
Router(config-cmap)#match ?
      access-group            Access group
      any                     Any packets
      class-map               Class map
      cos                     IEEE 802.1Q/ISL class of service
      destination-address     Destination address
      discard-class           Discard behavior identifier
      dscp                    Match DSCP in IP(v4) and IPv6 packets
      fr-de                   Match on Frame-relay DE bit
      fr-dlci                 Match on fr-dlci
      input-interface         Select an input interface to match
      ip                      IP specific values
      mpls                    MPLS specific values
      not                     Negate this match result
      packet                  Layer 3 Packet length
      precedence              Match Precedence in IP(v4) packets
      protocol                Protocol
      qos-group               Qos-group
      source-address          Source address
```

Enables NBAR

Testing, Verification & Assurance

Active Mode

NBAR Configuring Traffic Classification

```
router(config)# interface FastEthernet 0/1
router(config-if)# ip nbar protocol discovery

router(config)# class-map match-all MyTraffic
router (config-cmap)# match protocol gnutella file-transfer "*"
router (config-cmap)# match protocol gnutella file-transfer "*.mpeg"

router(config)# policy-map MyPolicy
router(config-pmap)# class MyTraffic
router(config-pmap-c)#
router(config-pmap-c)# set dscp 1
router(config-pmap-c)# set ip precedence 5
router(config-pmap-c)# police rate percent 50

router(config)# interface FastEthernet 0/1
router(config-if)# service-policy output MyPolicy
```

Enable Protocol Discover

Define Traffic Match, with Match Statements

Option to Create a Policy

Apply Policy

Testing, Verification & Assurance

NBAR – PDL and PDLM

PDLM (Protocol Description Language Module), the heart of the NBAR engine

PDL (native): Part of the Cisco IOS image (show ip nbar version)

- PDLM (non-native extensions): Download from Cisco Connection Online
- PDLMs are separated files that add quick support for new protocols
  and applications
- PDLMs become PDLs in the next Cisco IOS release (show ip nbar pdlm)
- PDLM are loaded from flash memory, usually no reboot
- Do not require an Cisco IOS upgrade; exception: Skype with Cisco IOS 12.4(4)T (no PDLM)
- PDLM size ~ 100kB (e.g., http 115kB)

Testing, Verification & Assurance

NBAR – PDLM Configuration

- CLI “match protocol” displays the protocols that NBAR supports

```
Router(config)#class-map match-all nbar_test
Router(config-cmap)#match protocol ?
 …
 bittorrent    bittorrent
 …
 citrix        Citrix Systems Metaframe 3.0
 …
 directconnect Direct Connect Version 2.0
 …
…
```

All Protocols Listed, Even If Added as PDLM

Testing, Verification & Assurance

NBAR – PDLM Show and Load Commands

```
Router# show ip nbar version

  NBAR software version: 6

  …

  14 napster          Mv: 3
  15 fasttrack        Mv: 2
  16 gnutella         Mv: 3, Nv: 2; disk1:gnutella.pdlm
  17 kazaa2           Mv: 7
```

Added with a PDLM

To Load the PDLM to the Router

```
Router(config)# ip nbar pdlm device:pdlm-name
```

Testing, Verification & Assurance

NBAR – Supported Protocols

Enterprise Applications	Security and Tunneling	Network Mail Services	Internet
Citrix ICA	GRE	IMAP	FTP
PCAnywhere	IPINIP	POP3	Gopher
Novadigm	IPsec	Exchange	HTTP
SAP	L2TP	Notes	IRC
Routing Protocols	MS-PPTP	SMTP	Telnet
BGP	SFTP	**Directory**	TFTP
EGP	SHTTP	DHCP/BOOTP	NNTP
EIGRP	SIMAP	Finger	NetBIOS
OSPF	SIRC	DNS	NTP
RIP	SLDAP	Kerberos	Print
Network Management	SNNTP	LDAP	X-Windows
ICMP	SPOP3	**Streaming Media**	**Peer-to-Peer**
SNMP	STELNET	CU-SeeMe	BitTorrent
Syslog	SOCKS	Netshow	Direct Connect
RPC	SSH	Real Audio	eDonkey/eMule
NFS	**Voice**	StreamWorks	FastTrack
SUN-RPC	H.323	VDOLive	Gnutella
Database	RTCP	RTSP	KaZaA
SQL*NET	RTP	MGCP	WinMX 2.0
MS SQL Server	SIP	**Signaling**	
	SCCP/Skinny	RSVP	
	Skype		

NBAR Recent and Upcoming Additions 2008

Enterprise Applications
Citrix ICA Priority Tagging
SAP(c-app, c-msg, app-app)
Peer-to-Peer
BitTorrent
Direct Connect
eDonkey/eMule
FastTrack
Gnutella (update)
WinMX 2.0
Streaming Media
RTSP
MGCP
Voice
RTCP
SIP
SCCP/Skinny
Skype v1 12.4(4)T
Security and Tunneling
L2TP
User-Defined
HTTP header field 12.3(11)T
Multiple matches per port 12.4(2)T

Enterprise Applications
DiCom
HL7
FIX
CIFS
Messaging
Yahoo
AOL
MSN
Sametime / Lotus
GoogleTalk
Voice
SKYPE 2.0, 3.0
Softphone
Network Mail Services
Exchange 2003
Peer to Peer

On PISA in Summer, followed by IOS 2HCY08

Cisco Software Download: NBAR Packet Description Language Modules

http://tools.cisco.com/support/downloads/go/Redirect.x?mdfid=268437899

How to Identify “unclassified” Traffic?

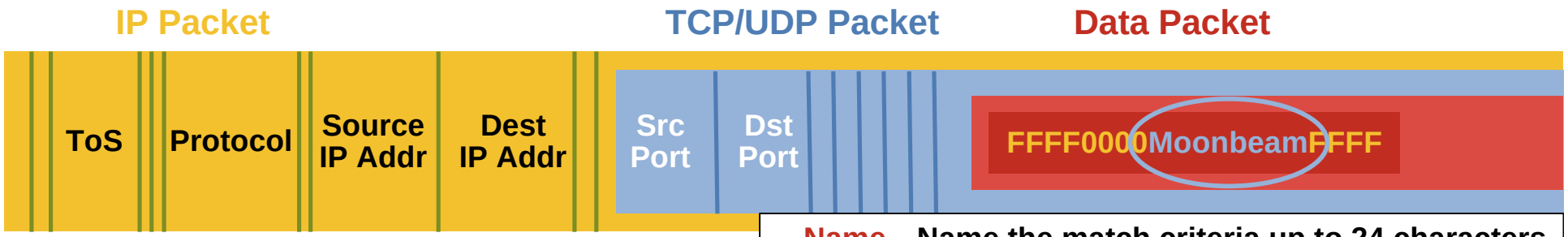
```
Router# show ip nbar unclassified-port-stats

Port Statistics for unclassified packets is not turned on.

Router# debug ip nbar unclassified-port-stats
```

```
Router# debug ip nbar filter destination_port tcp <#>

Router# debug ip nbar capture a b c d
  a: number of bytes (40-512)
  b: number of starting packets to capture (after TCP SYN)
  c: number of final packets to capture
  d: number of total packets to capture
```

The Debug IP NBAR Commands Should Be Enabled Only Under Carefully Controlled Circumstances!

http://www.cisco.com/en/US/tech/tk543/tk757/technologies_tech_note09186a0080094ac5.shtml

NBAR User-Defined Custom Application Classification Example




```
Router(config)#

ip nbar custom lunar_light
  8 ascii Moonbeam tcp
  range 2000 2999


class-map solar_system

match protocol lunar_light

policy-map astronomy
  class solar_system
  set ip dscp AF21

interface Serial1

 service-policy output astronomy
```

Name—Name the match criteria up to 24 characters >> lunar light

Offset—Specify the beginning byte of string or value to be matched in the data packet, counting from zero for the first byte >> Skip first 8 bytes

Format—Define the format of the match criteria ASCII, hex or decimal >> ascii

Value—Should match with the value in the packet If ASCII, up to 16 characters >> Moonbeam

[Source or destination port]—Optionally restrict the direction of packet inspection; defaults to both directions if not specified >> [source | destination]

TCP or UDP— Indicate the protocol encapsulated in the IP packet >> tcp

Range or selected port number(s)—"Range" with start and end port numbers >> Range 2000 2999

NBAR User-Defined Custom Application Multiple Matches per Port*

- "Multiple Matches Per Port" increases flexibility of user-defined application recognition

```
Router(config)# ip nbar custom <name> [offset [format value]]
[variable field-name field-length] [source|destination] [tcp | udp]
[range start end | port-number]
```

- Example:

```
Router(config)# ip nbar custom virus_home 20 hex variable scid 1
dest udp 5001 5005
Router(config)# class-map active-craft
Router(config-cmap)# match protocol virus_home scid 0x15
Router(config-cmap)# match protocol virus_home scid 0x21
Router(config)# class-map passive-craft
Router(config-cmap)# match protocol virus_home scid 0x11
Router(config-cmap)# match protocol virus_home scid 0x22
```

- Successor: Flexible Packet Matching (FPM)

*12.4(2)T

Testing, Verification & Assurance

Flexible Packet Matching (FPM)

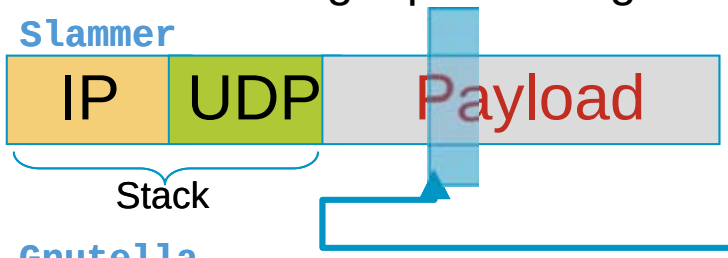
- Matches any characteristics in a packet header and payload:

  Matches L2-L7 information/Specify arbitrary bits/bytes at any offset

- Traffic matching a given protocol stack is subject to an FPM Deep Packet inspection rule

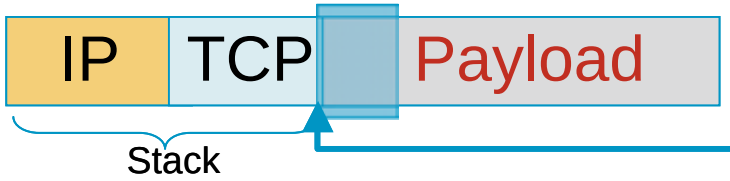
- Supports pattern matching through regular expressions and string matching: up to 48 regular expressions (32 Bytes window in 12.2(18)ZY)

Slammer



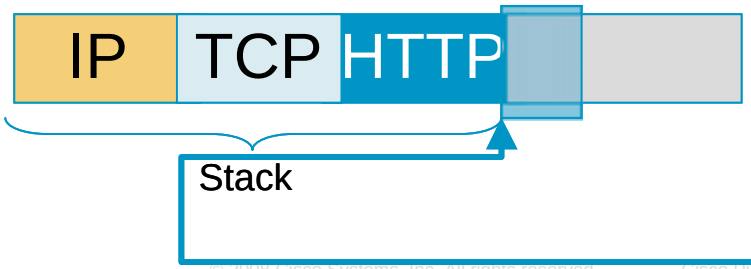

```
class-map type access-control match-all slammer
  description "match on slammer packets"
  match field UDP dest-port eq 1434
  match field IP length eq 404
  match start UDP payload-start offset 196 size 4 eq 0x4011010
```

Gnutella




```
class-map type access-control match-all gnutella
  match start TCP payload-start offset 0 size 32 regex "^GNUTELLA
   CONNECT"
```

Nimda




```
class-map type access-control match-any cm-nimda1
  match start l3-start offset 40 size 32 regex "[\\/]csrss\.exe"
  match start l3-start offset 40 size 32 regex "[\\/]httpodbc\.dll"
  match start l3-start offset 40 size 32 regex "[\\/]sample\.exe"
  match start l3-start offset 40 size 32 regex "[\\/]dnsservice\.exe"
  match start l3-start offset 40 size 32 regex "[\\/]puta\.eml"
  match start l3-start offset 40 size 32 regex "[\\/]puta\.scr"
  match start l3-start offset 40 size 32 regex "[\\/]readme\.eml"
```

Testing, Verification & Assurance

FPM – Policy CLI Overview

- Define and load packet header characteristics:
  - Use predefined Protocol Header Definition File (PHDF) or build custom PHDF. Examples include Ethernet, IP, TCP, UDP, GRE, ICMP, HTTP PHDF
- Define the protocol stack for packets subject to the FPM rule
  - Examples of protocol stacks include IP, IP/TCP, IP/UDP, IP/TCP/HTTP, IP/GRE/IP
- Define the FPM filter
  - This defines the payload characteristics/pattern
- Define the FPM rule
  - For a given FPM filter, the traffic can be permitted, dropped, or/and logged
- Associate the FPM rule with the relevant protocol stack
  - Only specific types of packets are subject to the FPM rule

```
load protocol disk0:ip.phdf
load protocol disk0:<>.phdf
```

```
class-map type stack match-all FPM-Stack
  match field <IP/TCP, IP/UDP, …>
```

```
class-map type access-control match-all FPM-Filter
  match field <stack header fields>
  match start <pattern at specific offset>
```

```
policy-map type access-control FPM-Rule-Child
  class FPM-Filter
    <action>
```

```
policy-map type access-control FPM-Parent
  class FPM-Stack
    service-policy FPM-Rule-Child
```

```
<interface X>
   service-policy access-control input FPM-Parent
```

Testing, Verification & Assurance

Catalyst 6500 Supervisor Engine 32 PISA

Supervisor Engine 32 PISA
8x1GE Uplinks + 1x 10/100/1000

Supervisor Engine 32 PISA
2x10GE Uplinks + 1x 10/100/1000

- **NBAR**
  - Application awareness and intelligent classification
  - Multigigabit Performance

- **Flexible Packet Matching**
  - Rapid Security Protection
  - Multigigabit Performance

- Programmable architecture
  - Seamless new service adoption

- Full Integration with
  - IPv4 & IPv6 in hardware
  - Advanced multicast & MPLS
  - Enhanced Manageability
  - HA with NSF/SSO and more

Testing, Verification & Assurance

NBAR on Supervisor Engine 32 PISA

- Supported features:
  - NBAR and FPM are accelerated in hardware
  - PISA accelerates Layer 3 IPv4 unicast packets **only**
  - PISA does **not** accelerate Layer2 packets or multicast packets
  - Microflow policing does **not** work with PISA
  - L2 NDE is **not** supported
- Supported interfaces:
  - Fast/Gig/TenGig Ethernet interfaces, Portchannels, VLANs, Trunks, Subinterfaces (Routed ports and SVIs **only**)
  - NBAR/FPM are **not** accelerated by PISA when configured on WAN interfaces. NBAR can however be accelerated by on WAN interfaces by the Enhanced FlexWAN and the SIP-200
  - Accelerated features can **not** be applied on MPLS, VPN/Tunnel interfaces

Testing, Verification & Assurance

NBAR on Supervisor Engine 32 PISA

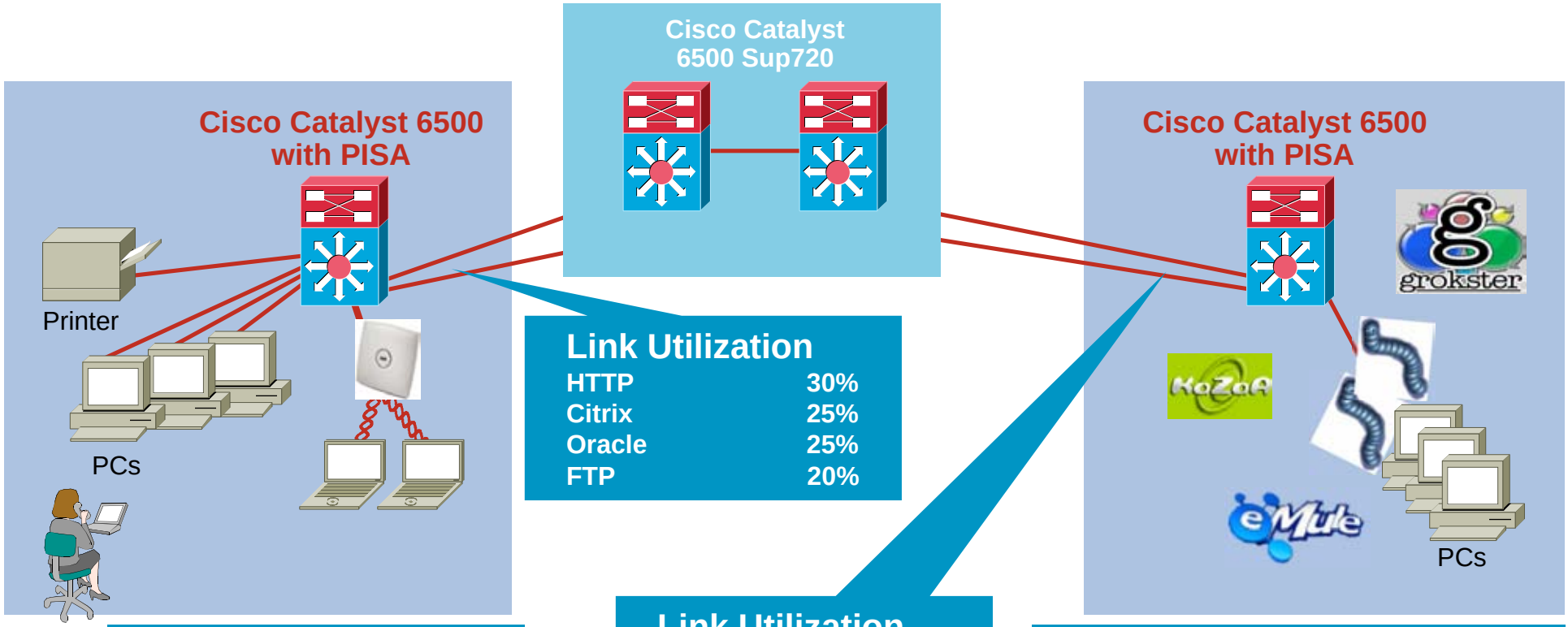
Scalability Summary for 12.2(18)ZY

	NBAR
Stateful	Yes
Maximum performance	2Gbps
Maximum packet size	4096B (scalable to 8KB)
Supports L3 IPv4 packets	Yes (only L3 IPv4 packets)
Current number of PDLM supported	90+
Supports regular expressions	Yes (only for PDLM)
Supports heuristics (dynamic port allocation)	Yes
Maximum number of Bytes matched in a row by regular expressions	32B
Sub-classification for selective protocol	HTTP (MIME/URL/host, server and client header filtering), Citrix (ICA tag, application), RTP (payload type, audio, video), Gnutella (file transfer)
Maximum number of concurrent sub-classifications	24 per system
Maximum URL size (sub-protocols parameter)	80 characters
Maximum number of interfaces	256
Supports custom policies with match at an offset	Yes (TCP and UDP packets only)
Custom policies (how far into payload)	256B into the payload
Supports packets with IP Options	No
Supports fragment reassembly	No

Testing, Verification & Assurance

NBAR on Supervisor Engine 32 PISA

Campus Access Deployments



Link Utilization

HTTP	30%
Citrix	25%
Oracle	25%
FTP	20%

Link Utilization

HTTP	50%
Grokster	15%
eMule	10%
Citrix	9%
Oracle	9%
FT	7%

- Mark Business-critical applications real-time as Gold service
- Police non-priority applications

- Block worms like Slammer using Flexible Packet Matching
- Detect and Rate-limit undesired p2p traffic with NBAR

Agenda

Introduction & Overview

Service Planning

[Coffee Break]

Service Deployment & Activation

[Lunch Break]

Service Testing, Verification & Assurance

[Coffee Break]

➲ Troubleshooting & Optimization

Summary

Be Prepared – Some Good Practices

Be Prepared – Some Good Practices

Troubleshooting & Optimization

Good Practice: Reserve Memory for Cons.

- **Problem:** Network or Device Problems may consume a lot of Memory and/or Memory may become extensively fragmented – potentially there won't be enough Memory left for the Console …
- **Solution:** Reserve Memory for the console ahead of time, on every device

```
Router(config)# memory reserved console <number-of-kilobytes>
```

Rule of Thumb: for the number of kilobytes use a value greater than 3 times the NVRAM size

- IOS Default is 256 kilobytes
- available since 12.0(22)S, 12.2(28)SB (7300), 12.4(15)T

Troubleshooting & Optimization

Good Practice: Check SNMP OID Statistics

Which OIDs are my NMS Apps (CiscoView) polling ?

```
Router#show snmp statistics oid

time-stamp     #of times requested  OID
16:16:50 CET Jan 12 2005     97 sysUpTime
16:16:50 CET Jan 12 2005     9  cardTableEntry.7
16:16:50 CET Jan 12 2005     9  cardTableEntry.1
16:16:50 CET Jan 12 2005     4  cardTableEntry.9
16:16:50 CET Jan 12 2005     16 ifAdminStatus
16:16:50 CET Jan 12 2005     16 ifOperStatus
16:16:50 CET Jan 12 2005     6  ciscoEnvMonSupplyStatusEntry.3
16:16:50 CET Jan 12 2005     17 ciscoFlashDeviceEntry.2
16:16:50 CET Jan 12 2005     8  ciscoFlashDeviceEntry.10
16:16:50 CET Jan 12 2005     2  ltsLineEntry.1
16:16:50 CET Jan 12 2005     2  chassis.15
16:16:27 CET Jan 12 2005     11 ciscoFlashDeviceEntry.7
16:16:27 CET Jan 12 2005     2  cardIfIndexEntry.5
16:16:24 CET Jan 12 2005     1  ciscoFlashDevice.1
```

Not yet widely available

Troubleshooting & Optimization

Good Practice: IfIndex Persistence – 1/3

- Feature which can make ifIndex persist across reboots (In Switches is on by default)
- ifIndex persistence means that the mapping between the ifDescr (or ifName) and ifIndex object values from the IF-MIB is retained across reboots.
- Useful:

  SNMP: monitoring the interfaces counters

  NetFlow: reporting of the interface ifIndex

  RMON: events/alarms based on specific interfaces
- 25 bytes of NVRAM used by this feature per interface.

Applying ifIndex persistence to all interfaces

```
Router(conf)# snmp-server ifindex persist
Router(config-if)# snmp-server ifindex persist
```

Applying ifIndex persistence to an specific interface

Good Practice: IfIndex Persistence – 2/3

Now there is a show command:

```
Router# show snmp mib ifmib ifindex
 Ethernet0/0: Ifindex = 1
 Loopback0: Ifindex = 39
 Null0: Ifindex = 6
:

Router# snmp mib ifmib ifindex loopback 0
 Loopback0: Ifindex = 39
```

Introduced in 12.0(7)S, 12.2(2)T

http://www.cisco.com/en/US/customer/products/sw/iosswrel/ps1839/products_feature_guide09186a0080087b0d.html

Troubleshooting & Optimization

Good Practice: IfIndex Persistence – 3/3

```
Router(config)# snmp-server ifindex persist
Router(config)# snmp mib persist event
Router(config)# snmp mib persist expression
Router(config)# snmp mib persist circuit
Router(config)# snmp mib persist cbqos
```

EVENT-MIB

EXPRESSION-MIB

CIRCUIT-MIB

CISCO-CLASS-BASED-QOS-MIB

- You must perform a *`copy running starting`* command to persist the newly assigned ifIndex values.

copy running start!

```
Router # dir nvram:ifIndex-table
  Directory of nvram:/ifIndex-table
      2   -rw-   283  <no date>  ifIndex-table
  126968 bytes total (114116 bytes free)
```

Reliable Delivery and Filtering of Syslog

Reliable Delivery and Filtering of Syslog

- Provides for reliable and secure delivery for syslog messages using Blocks Extensible Exchange Protocol (BEEP)

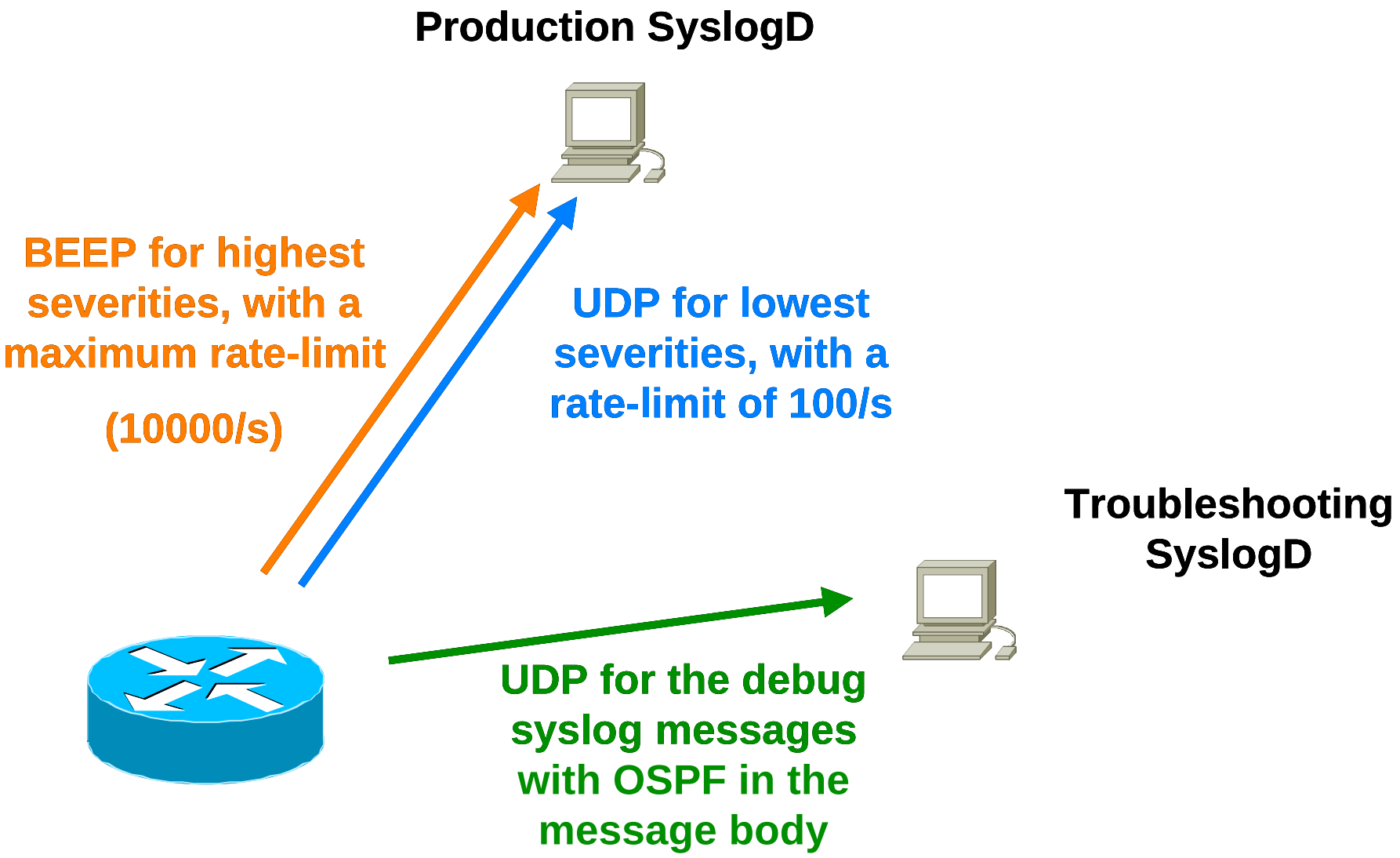
  RFC 3195, “Reliable Delivery for syslog”

- Provides a filtering mechanism per syslog session, called a message discriminator
- Provides a rate-limiter per syslog session
- Integrated in 12.4(11)T, even if the BEEP framework was supported for quite some time, 12.4(2)T
- Which syslog servers support BEEP?

  http://www.syslog.cc/ietf/rfcs/3195.html

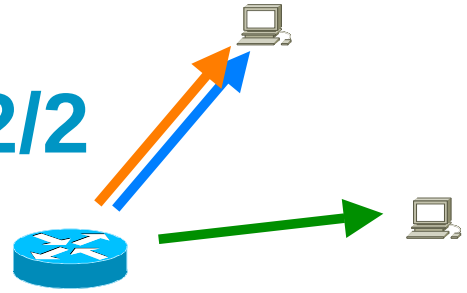
Troubleshooting & Optimization

Example: Filtering of Syslog – 1/2



Troubleshooting & Optimization

Example: Filtering of Syslog – 2/2



```
Router(config)# logging discriminator filter1
  severity includes 0,1,2,3 rate-limit 10000
Router(config)# logging discriminator filter2
  severity includes 4,5,6,7 rate-limit 100
Router(config)# logging discriminator filter3 msg-
  body includes debug includes facility OSPF

Router(config)# logging trap debugging

Router(config)# logging host <production> transport
  beep discriminator filter1
Router(config)# logging host <production> transport
  udp port 1471 discriminator filter2
Router(config)# logging host <troubleshooting>
  discriminator filter3
```

POST (Power-On Self-Test) is a great thing ...

... but some errors you prefer to know while the system is still running

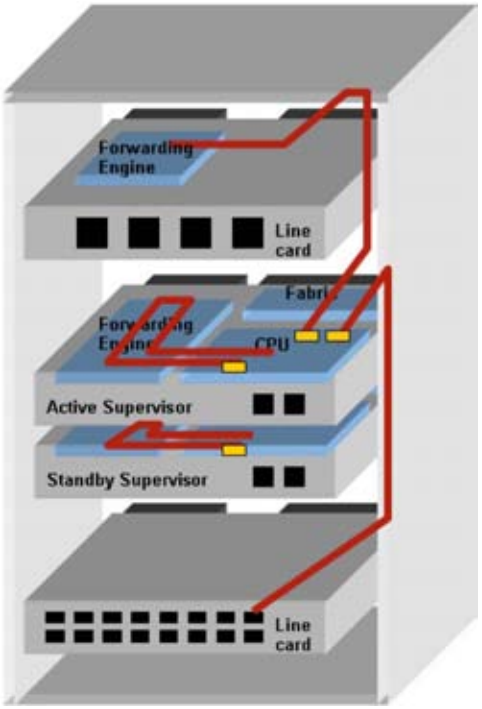
Troubleshooting & Optimization

Generic OnLine Diagnostics (GOLD)

CLI and scheduling for Functional Runtime Diagnostics

- Bootup Diagnostics (upon bootup and OIR)
- Periodic Health Monitoring (during operation)
- OnDemand (from CLI)
- Scheduled Testing (from CLI)
- Test Types include:
  - Packet switching tests
    - Are supervisor control plane & forwarding plane functioning properly?
    - Is the standby supervisor ready to take over?
    - Are linecards forwarding packets properly?
    - Are all ports working?
    - Is the backplane connection working?
  - Memory Tests
  - Error Correlation Tests
- Complementary to POST

Good Practice: schedule all non-disruptive tests periodically



Available on CRS-1, 7600, 6500, 4500, 3750, ...

Troubleshooting & Optimization

Example: The effect of wear and tear – 1/2

Problem: Repeated insertion and removal of Modules can lead to wear and tear damage on connectors. This in turn can cause failures … how do you find out during operation, without power-cycling the box ?

Solution: Use GOLD to verify functionality of a mis-behaving module

1) Let's see which GOLD tests are available and scheduled for our Module:

```
Router# show diagnostic content module 3
Module 3:

  Diagnostics test suite attributes:
    M/C/* - Minimal level test / Complete level test / Not applicable
      B/* - Bypass bootup test / Not applicable
      P/* - Per port test / Not applicable
    D/N/* - Disruptive test / Non-disruptive test/ Not applicable
      S/* - Only applicable to standby unit / Not applicable
      X/* - Not a health monitoring test / Not applicable
      F/* - Fixed monitoring interval test / Not applicable
      E/* - Always enabled monitoring test / Not applicable
      A/I - Monitoring is active / Monitoring is inactive

                                                        Testing Interval
  ID   Test Name                                 Attributes   (day hh:mm:ss.ms)
  ==== ================================== ============ =================
    1) TestScratchRegister -------------> *B*N****A    000 00:00:30.00
    2) TestSPRPInbandPing --------------> *B*N****A    000 00:00:15.00
    3) TestGBICIntegrity ---------------> *BPD****I    not configured
  :
   18) TestL3VlanMet -------------------> M**N****I    not configured
  :
```

Troubleshooting & Optimization

Example: The effect of wear and tear – 2/2

2) Now let's run TestL3VlanMet on-demand for Module 3:

```
Router# diagnostic start module 3 test 18
:
00:09:59: %DIAG-SP-3-MINOR: Module 3: Online Diagnostics detected a
Minor Error. Please use 'show diagnostic result <target>' to see
test results.
```

3) Then check the test results:

show diagnostics result module 3 detail

```
Router# show diagnostic result module 3
Module 3: CEF720 48 port 1000mb SFP  SerialNo : xxxxxxxx

  Overall Diagnostic Result for Module 3 : MINOR ERROR
  Diagnostic level at card bootup: minimal

  Test results: (. = Pass, F = Fail, U = Untested)
    1) TestTransceiverIntegrity:
   Port  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
   ---------------------------------------------------------------------------
         U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U

   Port 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
   ---------------------------------------------------------------------------
         U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U  U

:
:
   18) TestL3VlanMet -------------------> F
```

How is Cisco TAC using DMI to Troubleshoot?

How is Cisco TAC using DMI to Troubleshoot?

Acknowledgments to Marisol Palmero

Why this Session?

- We are sure our Device Instrumentation can solve many problems and can raise “CUSTOMER SATISFACTION” even if the Service Request (aka TAC case) was not open against it, any specific DI feature.
- Device Instrumentation offers great help to TROUBLESHOOT and WORKAROUND problems in the network
- This session is only composed of “Best Practices”

Give us your pain points!

Agenda

- Troubleshooting Techniques
  - Simplify and Speed up Network Troubleshooting
  - High CPU Utilization due to Processes
  - Using SNMP Utility in the router
- Best Practices (out of curiosity)
  - Show Diag Interpreter
  - Snapshots
  - Ideally On by Default
- Workaround limitations in IOS
  - Config Commands Don´t Process after Reload(**CSCsf32390)**
  - Tomasz´s SR
  - Maite's SR
  - Adam´s SR
  - Nenad´s SR
  - Raphael´s SR

But also…
Cisco Built-In Scripts (Cont.)

```
Router#show event manager policy available system          *latest 12.4(15)T
No.  Type    Time Created              Name
1    system  Thu Feb 7  06:28:15 2036  ap_perf_test_base_cpu.tcl
2    system  Thu Feb 7  06:28:15 2036  no_perf_test_init.tcl
3    system  Thu Feb 7  06:28:15 2036  sl_intf_down.tcl
4    system  Thu Feb 7  06:28:15 2036  tm_cli_cmd.tcl
5    system  Thu Feb 7  06:28:15 2036  tm_crash_reporter.tcl
6    system  Thu Feb 7  06:28:15 2036  tm_fsys_usage.tcl
```

```
Router#show event manager policy available system          *latest 12.2(33)SC
No.  Type    Time Created              Name
1    system  Thu Feb 7  06:28:15 2036  Mandatory.go_asicsync.tcl
2    system  Thu Feb 7  06:28:15 2036  Mandatory.go_bootup.tcl
3    system  Thu Feb 7  06:28:15 2036  Mandatory.go_fabric.tcl
4    system  Thu Feb 7  06:28:15 2036  Mandatory.go_fabrich0.tcl
5    system  Thu Feb 7  06:28:15 2036  Mandatory.go_fabrich1.tcl
6    system  Thu Feb 7  06:28:15 2036  Mandatory.go_ipsec.tcl
7    system  Thu Feb 7  06:28:15 2036  Mandatory.go_mac.tcl
8    system  Thu Feb 7  06:28:15 2036  Mandatory.go_nondislp.tcl
9    system  Thu Feb 7  06:28:15 2036  Mandatory.go_scratchreg.tcl
10   system  Thu Feb 7  06:28:15 2036  Mandatory.go_sprping.tcl …
```

Troubleshooting Techniques

- **Simplify and Speed up Network Troubleshooting**
- **High CPU Utilization due to Processes**
- **Using SNMP Utility in the router**

Simplify and Speed up Network Troubleshooting

- Create new exec commands combining TCL and alias command

```
Router(config)#alias exec ifchange tclsh flash:ifchange.tcl
Router(config)#exit

Router#ifchange
Syntax: tclsh ifchange.tcl interface [on|off|change|flap]

Router#sh ip int brief
Interface                  IP-Address      OK? Method Status Protocol
FastEthernet0/0            10.48.75.1      YES NVRAM  up     up
FastEthernet0/1            172.16.1.1      YES NVRAM  up     down
Loopback0                  1.1.1.1         YES manual up     up

Router#ifchange loop0 off
Interface loop0 changed state to off

Router#sh ip int brief
Interface                  IP-Address      OK? Method Status   Protocol
FastEthernet0/0            10.48.75.1      YES NVRAM  up       up
FastEthernet0/1            172.16.1.1      YES NVRAM  up       down
Loopback0                  1.1.1.1         YES manual adm.down down
```

Collect the information Needed to Troubleshoot: High CPU Utilization due to Processes

```
event manager applet troubleshoot_info
event none
action 1.1 cli command "enable"
action 1.2 cli command "del /force flash:show_high_cpu"
action 1.3 cli command "show tech | append flash:show_high_cpu"
action 1.4 cli command "show proc cpu | append flash:show_high_cpu"
action 1.5 cli command "show interface | append flash:show_high_cpu"
action 1.6 cli command "show interface stat | append flash:show_high_cpu"
action 1.7 cli command "show align | append flash:show_high_cpu"
action 1.8 cli command "show version | append flash:show_high_cpu"
action 1.9 cli command "show log | append flash:show_high_cpu"
action 1.10 cli command "more flash:show_high_cpu"
action 1.11 syslog msg "Sending show tech..."
action 1.12 mail server $_email_server to mpalmero@cisco.com from
$_email_from cc attach@cisco.com subject " Show Tech Output (SR 60xxxxx)
" body "$_cli_result"
action 1.13 syslog msg "Show High CPU information sent!“

alias exec sendtech event manager run troubleshoot_info
```

Alias optional

Triggering Applet
Router#event manager run troubleshoot_info
Or
Router#sendtech

SNMP Utility in the Router

```
Router(config)# snmp-server manager

Router # snmp get v1 172.17.243.144 public oid system.4.0
  SNMP Response reqid 47, errstat 0, erridx 0
   system.4.0 =
 Router# snmp set v1 172.17.243.144 private oid system.4.0
  string mpalmero
  SNMP Response reqid 48, errstat 0, erridx 0
   system.4.0 = mpalmero
Router# snmp get v1 172.17.243.144 public oid system.4.0
  SNMP Response reqid 49, errstat 0, erridx 0
   system.4.0 = mpalmero
Router# snmp get-next v1 172.17.243.144 public oid system.4.0
  SNMP Response reqid 41, errstat 0, erridx 0
   system.5.0 = Router
```

- 12.0(22)S
- also available from the TCL shell (type ”tclsh”): 12.3(7)T

SNMP Utility in the Router

```
Router# snmp {get|get-next} {v1|v2c} <address> <community>
  retries <n>] [timeout <seconds>] oid <object identifier>
Router# snmp get-bulk {v1 | v2c} <address> <community>
  [retries <n>] [timeout <seconds>] non-repeaters <n> max-
  repititions <n> oid <object identifier>
Router# snmp set {v1 | v2c} <address> <community> [retries
  <n>] [timeout <seconds>] oid <object identifier>
  {integer|string| counter | gauge | ip-address} <value>
Router# snmp inform {v1 | v2c} <address> <community> [retries
  <n>] [timeout <seconds>] trap-oid <object identifier> oid
  <object identifier>  {integer|string|counter|gauge|ip-
  address} <value>
```

Using the tclsh

```
Router# tclsh
Router(tcl)# snmpget…
```

- Requires 12.3(7)T

Best Practices

- **Show Diag Interpreter**
- **Snapshots**
- **Ideally On by Default**
- **Archive Configuration If Changes**

Show Diag Interpreter

```
event manager applet event_interpreter
 event cli pattern "show diag" sync no skip yes
 action 1.0 info type routername
 action 2.0 cli command "enable"
 action 3.0 cli command "show diag"
 action 4.0 mail server "email.cisco.com" to
"diag@external.cisco.com" from "mpalmero@cisco.com" subject
"Show diag on  $_info_routername" body "$_cli_result"
```

Results when typing “show diag” in the router:

```
Date Fri, 14 Dec 2007 152318 -0800 (PST)
To mpalmero@cisco.com
From Show Diag Interpreter <diag@external.cisco.com>
Subject Re Show diag on  matt

Slot 1 PA-2FE-TX



_____________________________________________________________________
To get help, send an email with 'help' as subject.
To contribute feedback, send an email with 'feedback' as subject,
and your comments in the email body.
```

Snapshots

- Safe Interface Status every 60 sec in a tftp/ftp server

```
event manager applet SaveInterfaceStatus
 event timer watchdog name "SaveIfStat" time 60
 action 1.0 cli command "enable"
 action 2.0 cli command "show ip interface brief | redirect
tftp://username:passw@144.254.15.108/interface.txt"
 action 3.0 syslog msg "Interface status saved unmatched"
!
```

- Other Snapshot: We need to troubleshoot SNMP issues, to collect two instances after 60 sec, for CLI and show command to run together

Ideally On by Default

- Exclusive Configuration Change Access and Access Session Locking

```
Router(config)# configuration mode exclusive auto
```

The auto keyword automatically locks the configuration session whenever the configure terminal command is used. Default time: 600 seconds

```
Router# config t
Router# Configuration mode locked exclusively by user
‘marisol' process '3' from terminal '0'. Please try later.
```

Troubleshooting Techniques

- **Release-note for** CSCsf32390
- **Tomasz´s SR- Kron vs EEM**
- **NetFlow Limitation**
- **Raphael´s SR**
- **Adam´s SR**
- **Nenad´s SR**

Config Commands Don´t Process after Reload
CSCsf32390 (R) – Release-note

- Sometimes, parts of router configuration get lost during the reload process: although the configuration commands are saved in NVRAM, they are not processed after the reload and thus do not appear in the running configuration. Re-entering these commands manually solves the problem ...

```
event manager applet CSCsf32390
event syslog occurs 1 pattern "%SYS-5-RESTART: System restarted"
action 1.0 cli command "enable"
action 2.0 cli command "configure terminal"
action 3.0 cli command "buffers particle-clone 16384"
action 4.0 cli command "buffers header 4096"
action 5.0 cli command "buffers fastswitching 8192"
action 6.0 syslog msg "Reinstated buffers command“
```

Kron vs. EEM (Reminder – DI Jun´07)

- Kron was introduced in 12.3(1)
- EEM “event timer cron” was introduced in 12.2(25)S and 12.3(14)T
- Kron & EEM: Trigger a set of CLI commands at reload/periodic intervals
- Only Kron has the ability to specify different username for each event
- Drawback in Kron

  Cannot collect outputs with Kron, and not all EXEC commands are working under the kron policy cli ...

Kron vs EEM
Tomasz´s SR

- Problem: The MD5 configuration has to be re-applied manually to the interface for EIGRP adjacency to be re-established

  Router c3250 doesn´t support EEM in 12.4(11)T >>CSCsh33013

  Headline: Support for EEM v2.2 in 3200 Router Series

- CSCsm78264 (A)

  Headline: ip authentication mode eigrp [] dissapears after ip address is released

  Workaround

```
kron occurrence check_eigrp_auth in 1
recurring
  policy-list check_eigrp_auth
!
kron policy-list check_eigrp_auth
  cli tclsh  flash:check_auth.tcl
```

NetFlow Limitation – Maite´s SR

- Problem Description

  in c7609 12.2(18)SXF3 : TCAM at 100% usage all time, and the counter "Netflow Creation Failures" is increasing in each table

- Possible solution: In SRB NetFlow is done as per interface basis and this problem might be dissapearing...

No Bug > Software and HW Limitation.

- Intermediate solution: after the time SRB could be implemented, AS developed a TCL script based on EEM timer event

  Get the number of creation failures and the number of packets switched for each one of the DFCs and Active supervisor. We will wait 5 sec, and calculate over the total packet switches, the number of NetFlow Creatioin Failures.

- Customer wanted to monitor via SNMP:

  cseL3FlowLearnFailures

  Description "Number of flows that failed to be learned because the Layer 3 flow table in this switching engine was full."

Raphael´s SR (P2)

- Problem Description:

  On a Cisco 7600 router, 12.2(33)SRA, the MAC address of one or more interfaces may change unexpectedly when the ifPhysAddress object of the IF-MIB is accessed by SNMP. This situation prevents the router from receiving packets when an ARP entry that contains the MAC address of the router is refreshed.

- Workaround for bug CSCsf04112 (R)

  Headline: Getting ifPhysAddress changes mac-address of SPA-2X1GE

- Release-note

  Workaround: To prevent the symptom from occurring, configure static ARP on the devices that must be able to send packets to the router. After the symptom has occurred, reload the router to clear the condition.

mac_change.tcl

```
# mac_change.tcl - check if the mac address configured on the interface are
#                  the same as the one configured. The purpose is to provide
#                  a valuable workaround for CSCsf04112.
#
# trigger:  watchog, will be run every "$time_period" seconds
# action:   issues some commands to check the MAC address for configured
#           interface and change it back to it's correct value if it differs
#           from what has been configured.
# env vars (example):
#    Name          Value
#    cm_time_period 5                  -- Poll time (mandatory)
#    cm_int_3       GigabitEthernet5/2 001c.b0b5.6f40
#    cm_int_1       vlan1 001c.b0b5.6f40 -- String with interface name + Mac
#    cm_int_2       vlan2 001c.b0b5.6f40  . At least one mandatory, ending
#                                         numb must start at 1 with no gap
#                                        . No space in interface name
#    cm_verbose     0                  -- If 1: add extra output (non mandatory)
...
```

Adam ´s SR (P1)

- Problem Description:

  At random time OSPF process stop sedning/processing OSPF packets - probably ipprouting process is blocked - this stay there for 10 seconds and then process recover. Problem happen without SNMP enabled about 1 a 14 days with SNMP 1 a 3 days.

  CSCso55659 (I)

  Headline: cat6k/ION: iprouting.iosproc blocked for 10 second cause OSPF flaps.

- Troubleshooting Technique when difficult to reproduce in customer network/TAC LAB:

  – SNMP has been enabled to have problem triggered more often.

  –EEM script triggered by BFD debug has been started. When connected device do not receive OSPF packet it put OSPF down and put BFD to admin down status and report it to problem device - we run "debug bfd event" when we see UP - FAILING we run EEM script to collect data.

Adam´s SR

- EEM when we don´t know the cause of the problem

```
event manager applet test event syslog pattern "UP -> FAILING" maxrun 600
action 0.1 cli command "enable"
action 1.0 cli command "sh proc det iprouting.iosproc"
action 2.0 syslog msg "show proc: $_cli_result"
action 2.5 cli command "proc start taskinfo.proc"
action 2.6 syslog msg "taskinfo: $_cli_result"
action 3.0 cli command "sh proc eve 24622"
action 4.0 syslog msg "show proc events: $_cli_result"
action 5.0 cli command "sh proc det"
action 6.0 syslog msg "show proc det: $_cli_result"
action 7.0 cli command "show itrace data remote"
action 8.0 syslog msg "show proc det: $_cli_result" !
```

Nenad´s SR (CAP case)

- Problem Description:

  the issue was that ACE (loadbalancer) is doing SSL offloading and was crashing due nitrox chip hung. So, we had about 40 crashes in production and we were not able to find root cause from the core file of the ACE (had almost 10 engineering images) So, Developers idea was to actually disable watchdog on ACE which will reload the ACE in the case of hung and leave in ACE in hung state, in order to access to the ACE and collect the info in the hung state.

  CSCsl42384 R
  CSCsl99468 R
  CSCsm43541 I
  CSCso46209 A

- Possible Solution:

  “So, my simple idea was to actually monitor the MAC move notification for the virtual MAC address of the VirtualServersIP VIP's on ACE, which will occur in the case of the FT. So, as I mentioned I had to use regex to not trigger the script for each MAC address separately”

  The basically if ACE crash it will wait for syslog messages:

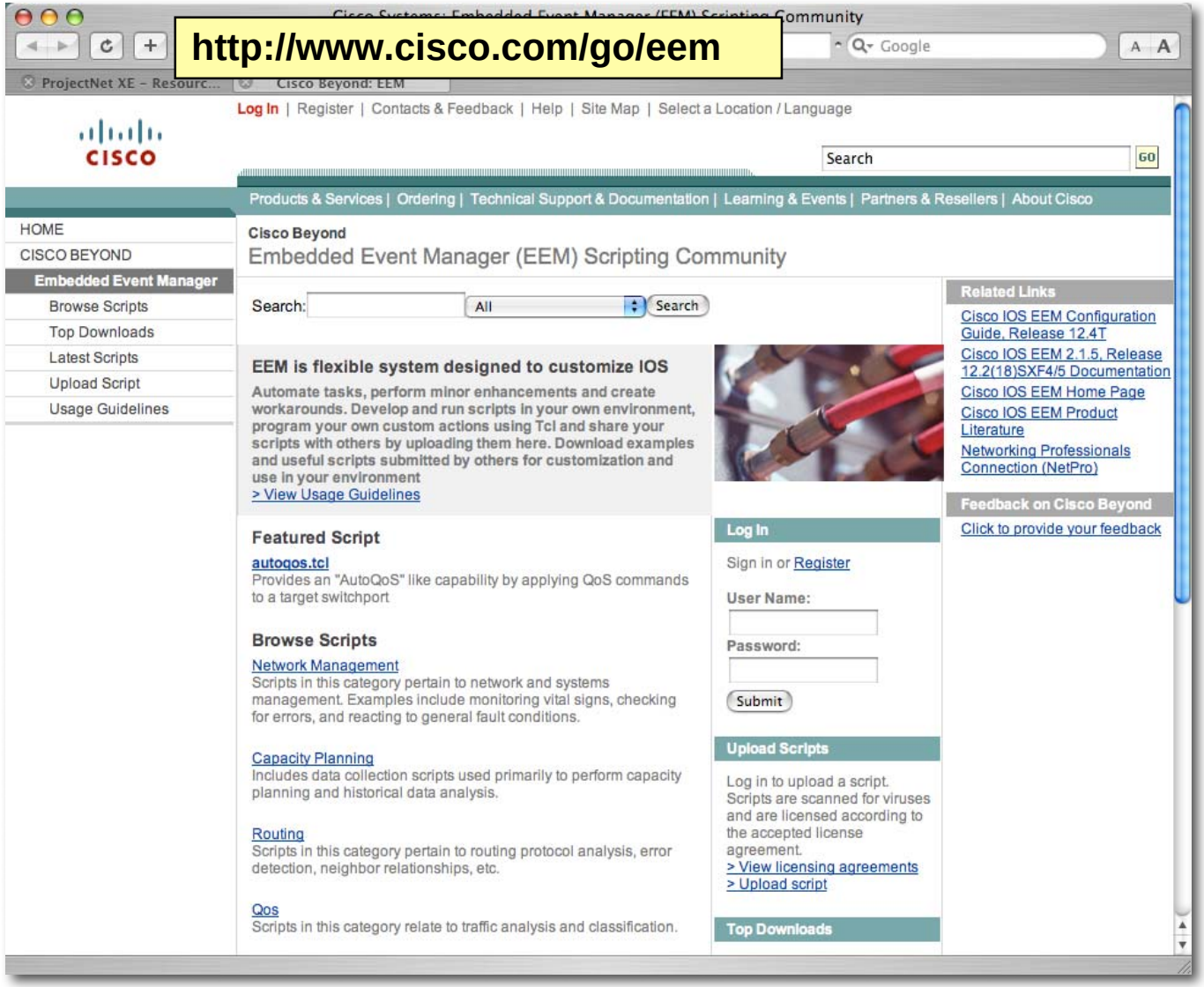
  %C6KPWR-SP-4-DISABLED: power to module in slot 1 set off (Reset)

Nenad´s case

```
event manager applet MAC_MOVE_of_the_ACE_VMAC__possible_crash event syslog occurs 1 pattern
.*MAC_MOVE.*[000b.fcfe.1b0a|000b.fcfe.1b14|000b.fcfe.1b28].*Te4/1.*
 action 1.0 syslog msg "EEM: MAC MOVE ACE issue - call Cisco TAC! SR 607579187"
 action 2.0 cli command "enable"
 action 3.0 cli command "configure terminal"
 action 4.0 cli command "no svclc module 4 vlan-group 101"
 action 5.0 cli command "do clear mac-address-table dynamic interface TenGigabitEthernet 4/1"
 action 6.0 cli command "power enable module 1"
 !  action 7.0 cli command "no event manager applet
MAC_MOVE_of_the_ACE_VMAC__possible_crash"
 action 7.4 cli command "interface Gi5/1"
 action 7.5 cli command "shutdown"
 action 7.6 syslog msg "EEM: shutdown of Gi5/1 done!"
 action 8.0 cli command "end"
 action 9.0 syslog msg "EEM: MAC_MOVE_of_the_ACE_VMAC__possible_crash done!"
 action 9.9 syslog msg "EEM: Trigger was: $_syslog_msg" end !
```

References

Some Other Examples in Cisco Beyond Link in http://www.cisco.com/go/eem



Other References

- NW sessions:
  - –“Advanced IOS Management” > Since NW´2005
  - –“Designing Manageability in the Service Oriented Network“ > New in NW´08 Europe
  - – Getting the Right Events from the Network Elements

Other References

- Cisco Connection Online:

  http://www.cisco.com/go/instrumentation

- Feature Navigator (External)

  http://www.cisco.com/go/fn

  It is Not possible to open bugs against it

- SNMP Object Navigator (External)

  http://www.cisco.com/go/mibs

Conclusion

Conclusion

- We see more and more IOS Device Instrumentation features in Cisco boxes, and many are not linked to a specific Techonology
- This tutorial is shared across TAC and HTTS
- This tutorial was to
  - –Highlight existing features to keep in mind
  - –Sharing Best practices, Troubleshooting Techniques, Workarounds for known IOS limitations and even known Defects (CDETs) to resolve and provide Best effort support always than possible for TAC cases,
  - –Given an added value to complete our TAC Support and raise Customer Satisfaction

Give us your pain points!

Agenda

Introduction & Overview

Service Planning

[Coffee Break]

Service Deployment & Activation

[Lunch Break]

Service Testing, Verification & Assurance

[Coffee Break]

Troubleshooting & Optimization

➲ Summary

Wrap-Up & Close

Questions during a Service Life Cycle

Is there room for yet another service?

- **Embedded Event Manager (EEM)**
- **Embedded Resource Manager (ERM)**
- Data Collection Manager (DCM)
- **NetFlow** MIB and Top Talkers
- ...

How to configure?

- **CLI Config Diff and Archive**
- CLI Config Locking
- **TCL and Cron**
- **IOS Deployment Agents**
- **IOS XML PI**
- **Enhanced Device Interface (E-DI)**
- ...

Is it working as specified?

- **IF-MIB, CB-QoS-MIB**
- EVENT-, EXPRESSION-MIB
- **NBAR**
- NetFlow
- **Sup Engine 32 PISA**
- **Flexible Packet Matching**
- **IP SLA**
- ...

- **SNMP, CLI**
- Embedded Resource Manager (ERM)
- Notification Log MIB
- Reliable Syslog
- TCL
- **Embedded Event Manager (EEM)**
- **GOLD**
- ...

What if something goes wrong?

- SNMP Stat OID
- IF-MIB, CB-QoS-MIB
- NBAR
- NetFlow
- **Embedded Event Manager (EEM)**
- **IP SLA and EOT**
- ...

Are we meeting SLA?

Wrap-Up & Close
References

Device Manageability Instrumentation (DMI) www.cisco.com/go/instrumentation

- Embedded Event manager: www.cisco.com/go/eem
  - Cisco Beyond – EEM Community: www.cisco.com/go/ciscobeyond
  - EEM Documentation: www.cisco.com/en/US/products/ps6017/products_feature_guide_book09186a008064206b.html
- IPSLA (aka SAA, aka RTR): www.cisco.com/go/ipsla
- NBAR: www.cisco.com/go/nbar
- NetFlow: www.cisco.com/go/netflow
- IOS TCL Scripting: http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801a75a7.html
  - Signed TCL Scripts: http://www.cisco.com/en/US/products/ps6441/products_feature_guide09186a00808d65fe.html
- DMI at work for Commercial Market: www.cisco.com/go/ioscommercial
- Feature Navigator: www.cisco.com/go/fn
- MIB Locator: www.cisco.com/go/mibs

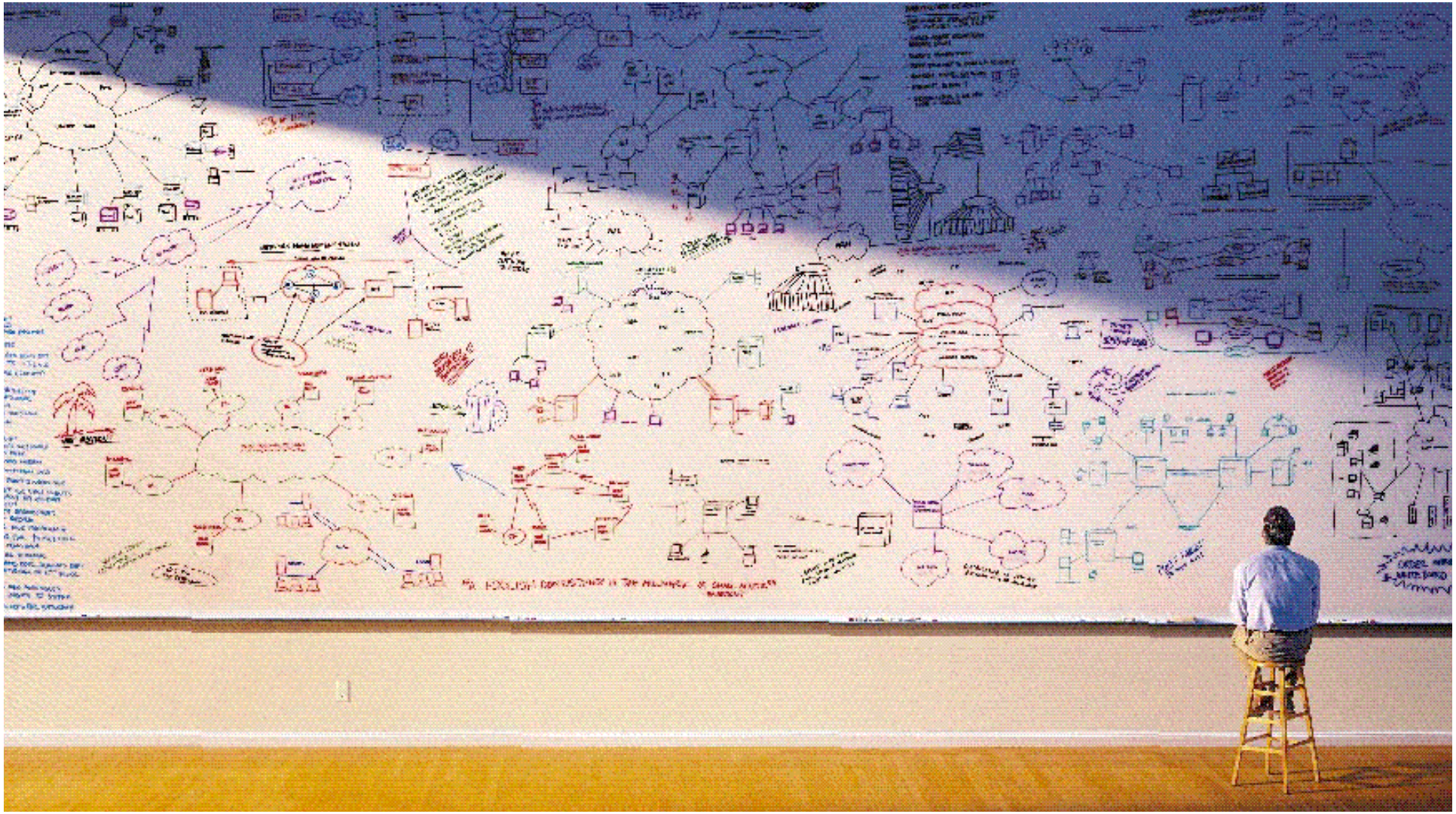
Network Management Applications

- Cisco Configuration Engine: www.cisco.com/en/US/partner/products/sw/netmgtsw/ps4617/
- Enhanced Device Interface (E-DI): www.cisco.com/en/US/products/ps6456/

Monthly Network Management Newsletter

- Cisco Network Management Newsletter (email subscription possible): http://www.cisco.com/external/networkmanagement/cnm-newsletter/April_08.html

Questions ?



Wrap-Up & Close
In Summary

Device Manageability Instrumentation (DMI) allows you to

- capture relevant and accurate information efficiently
- and evaluate results to trigger appropriate actions

How will you leverage DMI in network based Services ?

What new possibilities does DMI offer to you ?

thank you

bklauser@cisco.com

Appendix I:
Feature Availability

EEM Feature/Product Support Matrix

CISCO IOS EMBEDDED EVENT MANAGER
EEM VERSION - PRODUCT MATRIX

2/22/08 4:59 PM

Legend	Shipping	In EFT	EC	Planning	N/A

CISCO ACCESS ROUTERS

EEM Version	Cisco 800 Series	Cisco 1800 Series	Cisco 2800 Series	Cisco 3800 Series	Cisco 1700 Series	Cisco 2600 Series	Cisco 2600XM Series	Cisco 2691 Series	Cisco 3600 Series	Cisco 3700 Series
1.0		12.3(11)T	12.3(11)T	12.3(11)T	12.3(4)T	12.3(4)T	12.3(4)T		12.3(4)T	
2.0										
2.1		12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1	12.3(14)T1
2.1.5										
2.2	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T	12.4(2)T
2.3	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T	12.4(11)T
2.4	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T	12.4(20)T
3.0	12.5(pi1)T	12.5(pi1)T	12.5(pi1)T	12.5(pi1)T	Planning	Planning	Planning	Planning	Planning	Planning

CISCO 5000 SERIES & UP

EEM Version	Cisco 7200 Series	Cisco 7301	Cisco 7304	Cisco 7600 Series	Cisco 10000	Cisco 12000 Series	Cisco XR 12000	Cisco CRS-1	Cisco 7500 Series	Cisco 5000 Series
1.0						12.0(26)S			12.0(26)S	
2.0			12.2(27)SBC				See IOS-XR Fault Mgr	See IOS-XR Fault Mgr		
2.1	12.3(14)T1	12.3(14)T1	12.2(28)SB	12.2(18)SXF5	12.2(28)SB		See IOS-XR Fault Mgr	See IOS-XR Fault Mgr	12.4M	
2.1.5							See IOS-XR Fault Mgr	See IOS-XR Fault Mgr		
2.2	12.4(2)T	12.4(2)T1					See IOS-XR Fault Mgr	See IOS-XR Fault Mgr		
2.3	12.4(11)T	12.2(33)SB	12.2(33)SB	12.2(33)SRB	12.2(33)SB		See IOS-XR Fault Mgr	See IOS-XR Fault Mgr		12.4(11)T
2.4	12.4(20)T	12.2(SR) Eagle	12.2(SR) Eagle	12.2(SR) Eagle	12.2SR		See IOS-XR Fault Mgr	See IOS-XR Fault Mgr		Planning
3.0	12.5(pi1)T	Planning	Planning	Planning	Planning					Planning

CISCO CATALYST SWITCHES

EEM Version	Cisco 3750 Switches	Cisco 4500 Switches	Cisco 6500 Switches
1.0			
2.0			
2.1			IOS w/o Modularity 12.2(18)SXF5
2.1.5			w/ Modularity 12.2(18)SXF4
2.2			
2.3			12.2(33)SXH
2.4	12.2(40)SE	Planning	12.2(33)SXI)
3.0	Planning	Planning	Halfdome

Includes Futures, Subject to Change; No Commitment Implied

Embedded Event Manager

Event Detectors

EEM Event Detector - Release Matrix
Updated: 02/22/2008

Event Detector Name	12.0(26) 12.3(4)T	12.2(25)S	12.3(14)T	12.4(2)T	12.2(18)SXF4 (Mod IOS) 12.2(18)SXF5 IOS	12.4(20)T	Description
Application		YES	YES	YES	YES	YES	Custom application events, action script interaction
CLI			YES	YES	YES	YES	Exec command match and run
Counter		YES	YES	YES	YES	YES	Custom counter events
GOLD					YES		Generic Online Diagnostics event detection
Interface		YES	YES	YES	YES	YES	Interface counters and events
Memory Thresholding (Deprecated)							Detect memory resource related events
None (by run command)			YES	YES	YES	YES	
Object Tracking				YES		YES	Integration with Enhanced Object Tracking
OIR			YES	YES	YES	YES	Card Online Insertion & Removal detection
Resource Thresholding				YES	YES	YES	Integration with Embedded Resource Manager, supercedes Memory Thresholding ED
RF				YES	YES	YES	IOS Infrastructure Redundancy Facility events
SNMP	YES	YES	YES	YES	YES	YES	Detect MIB Var match and thresholds
SNMP Proxy						YES	Allows device to raise an event on RECEIPT of a trap or inform and execute a policy
Syslog	YES	YES	YES	YES	YES		Reg exp pattern match on emitted syslog messages
Timer		YES	YES	YES	YES		Custom timed events
IOS Watchdog Monitor		YES	YES	YES	YES		IOS scheduler, watchdog events
WDSysMon*					YES		IOS Modularity: System monitor event
XML-RPC (SOAP over SSHv2)						YES	Send a message to invoke a policy from outside the box

1.0 2.0 2.1 2.2 2.1+ 2.4

Deployment Agents

Cisco IOS Release Family Roadmap

Cisco IOS Software Platforms	Cisco 10000 Series	Cisco 7600 Series	Cisco 7500 Series	Cisco 7304 Router	Cisco 7301 and 7200 Router	Cisco Catalyst 6500 Series	Cisco Catalyst 4500 Series	Cisco 3750 and 2900 Series	
	12.2SB	12.2SR/SX	12.2SB	12.2SB	12.2SB/SR	12.2SX/ SR	12.2SG	12.2SE	T
UDI Support and Configuration Enhancements	12.2(1st)SB5	12.2(33)SRA	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(44)SE	12.5(1st)T
Deployment Agents (Configuration Agent and Event Agent)	12.2(31)SB	12.2(33)SRB	12.2(31)SB	12.2(31)SB	12.2(31)SB	12.2(1st)SXI	12.2(12th)SG	12.2(25)SEE	12.3(1)
Image Agent	12.2(31)SB	12.2(33)SRB	12.2(31)SB	12.2(31)SB	12.2(31)SB	12.2(1st)SXI	12.2(12th)SG	12.2(25)SEE	12.3(1)
Config Retrieve Retry	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(44)SE	12.4(15)T
Command Scheduler (Kron) Policy for System Startup	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(44)SE	12.4(15)T
Agents over IPv6	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	12.5(1st)T
Netconf over SSHv2	12.2(1st)SB5	12.2(33)SRA	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	12.4(9)T
Netconf over BEEP	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(6th)SE	12.4(9)T
Config Change Notification (Netconf)	12.2(1st)SB5	12.2(33)SRA	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	12.5(1st)T
Netconf over IPv6	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	12.5(1st)T
TR-069 Agent			NA				NA	12.2(7th)SE	12.5(1st)T
TR-069 Agent Phase 2			NA				NA	12.2(7th)SE	12.5(1st)T
Cisco Software Licensing			NA				NA	12.2(37)SE	12.4(7th)T

Parser

Release 12.2S and T Family Roadmap

Cisco IOS Software Platforms	Cisco 10000 Series	Cisco 7600 Series	Cisco 7500 Series	Cisco 7304 Router	Cisco 7301 and 7200 Routers	Cisco Catalyst 6500 Series	Cisco Catalyst 4500 Series	Cisco 3750 and 2900 Series	
	12.2SB	12.2SR/SX	12.2SB	12.2SB	12.2SB/SR	12.2SX/ SR	12.2SG	12.2SE	T
Configuration Replace and Configuration Rollback, Including Config Versioning (Archive) and Timed Rollback	12.2(31)SB	12.2(33)SRA	12.2(25)S	12.2(25)S	12.2(31)SB	12.2(1st)SXH	12.2(11th)SG	12.2(40)SE	12.3(7)T
Configuration Change Notification and Logging	12.2(1st)SB5	12.2(33)SRA	12.2(25)S	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(25)SEC	12.3(4)T
Contextual Configuration Diff Utility	12.2(1st)SB5	12.2(33)SRA	12.2(25)S	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(40)SE	12.3(7)T
Configuration Generation Performance Enhancement	12.2(1st)SB5	12.2(1st)SRC	12.2(25)S	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(44)SE	12.3(7)T
Role-Based Access Control CLI Commands	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(44)SE	12.3(11)T
NVGEN Enhancement Phase II (Config Partitioning Infra)	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	
Configuration Rollback Confirmed Change	12.2(1st)SB	12.2(1st)SRC	NA	12.2(1st)SB	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(44)SE	12.5(1st)T
IPv6 for Config Logger	12.2(1st)SB	12.2(1st)SRC	NA	12.2(1st)SB	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	12.5(1st)T
Config Logger Persistency	12.2(1st)SB5	12.2(33)SRA	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(44)SE	12.4(11)T
Exclusive Configuration Change Access and Access Session Locking	12.2(1st)SB5	12.2(33)SRA	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(44)SE	12.4(11)T
Config Change Tracking Identifier	12.2(1st)SB	12.2(1st)SRC	NA	12.2(1st)SB	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(44)SE	12.5(1st)T
XML Programmatic Interface	SB6	Dragon	NA	SB6	Dragon	HD	12.2(13th)SB	12.2(7th)SE	12.5(2nd)T
SOAP XML Infrastructure	SB6	Dragon	NA	SB6	Dragon	HD	12.2(13th)SB	12.2(7th)SE	12.5(2nd)T

IP SLA

Release 12.2S and T Family Roadmap

Cisco IOS Software Platforms	Cisco 10000 Series	Cisco 7600 Series	Cisco 7500 Series	Cisco 7304 Router	Cisco 7301 and 7200 Routers	Cisco Catalyst 6500 Series	Cisco Catalyst 4500 Series	Cisco 3750 and 2900 Series	
	12.2SB	12.2SR/SX	12.2SB	12.2SB	12.2SB/SR	12.2SX/ SR	12.2SG	12.2SE	T
IPSLAs Responder	NA	NA	NA	NA	NA	NA	NA	12.2(25)SEE	12.4(7) Mainline
IPSLAs CLI Introduction	12.2(31)SB	12.2(33)SRB	12.2(31)SB	12.2(31)SB	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(40)SE	12.3(14)T
IPSLA CLI Phase 2	12.2(31)SB	12.2(33)SRB	12.2(31)SB	12.2(31)SB	12.2(1st)SRC	12.2(1st)SXH	12.2(11th)SG	12.2(40)SE	12.4(2)T
IPSLA CLI Phase 3	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(40)SE	12.4(4)T
IPSLAs—LSP Health Monitor	12.2(27)SBB	12.2(33)SRA	12.2(30)S	12.2(27)SBB	12.2(27)SBB	12.2(1st)SXH	12.2(11th)SG	12.2(40)SE	12.4(6)T
IPSLAs Precision Improvements	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(40)SE	12.3(14)T
IPSLAs Additional Threshold Traps (VoIP)	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(40)SE	12.4(2)T
IP SLAs Random Scheduler	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(40)SE	12.4(2)T
IP SLAs—LSP Health Monitor with LSP Discovery	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	W3	12.2(11th)SG	12.2(40)SE	12.4(7th)T
IP SLAs for Metro Ethernet	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	TBD	12.2(40)SE	12.5(2nd)T
IP SLAs for IPv6 (UDP Jitter, UDP Echo, ICMP Echo, TCP Connect)	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	12.5(2nd)T
Auto IP SLAs for MPLS Pseudo Wire (PWE3) via VCCV	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	W3			
VRF Aware for TCP, FTP, HTTP and DNS Operations	SB7	Eagle	NA	SB7	Eagle	W3	12.2(13th)SG	12.2(7th)SE	12.5(2nd)T
IP SLAs—ICMP Jitter Operation			NA					12.2(7th)SE	12.4(6)T
IP SLA—VoIP Gatekeeper Delay Monitoring	NA	NA	NA	NA	NA	NA	NA	NA	12.3(14)T
IP SLAs VoIP Call Setup (Postdial Delay) Monitoring	NA	NA	NA	NA	NA	NA	NA	NA	12.3(14)T
IP SLAs Metro Ethernet EVC-1.0		Dragon	NA		Dragon	W3			

SNMP (1/2)

Release 12.2S and T Family Roadmap

Cisco IOS Software Platforms	Cisco 10000 Series	Cisco 7600 Series	Cisco 7500 Series	Cisco 7304 Router	Cisco 7301 and 7200 Routers	Cisco Catalyst 6500 Series	Cisco Catalyst 4500 Series	Cisco 3750 and 2900 Series	
	12.2SB	12.2SR/ SX	12.2SB	12.2SB	12.2SB/SR	12.2SX/ SR	12.2SG	12.2SE	T
Periodic MIB Data Collection and Transfer Mechanism	12.2(1st)SB5	12.2(33)SRA	12.2(22)S	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	12.3(2)T
Secure SNMP Views	12.2(1st)SB5	12.2(33)SRA	12.2(22)S	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	12.3(4)T
VPN-Aware SNMP Infrastructure	12.2(31)SB	12.2(33)SRA	12.2(22)S	12.2(25)S	12.2(31)SB	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	
SNMP over IPv6	12(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(11th)SG	12.2(44)SE	12.3(14)T
AES (RFC 3826) and 3DES Encryption for SNMP v3	12(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(6th)SE	12.4(2)T
CISCO-ENTITY-ALARM-MIB enh	12(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(12th)SG	12.2(6th)SE	12.4(4)T
ISSU—SNMP	12.2(27)SBB	12.2(1s)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	12.2(1st)SXI	12.2(31)SGA	NA	NA
Interface MIB Enhancements	12.2(31)SB	12.2(33)SRA	12.2(31)SB	12.2(31)SB	12.2(31)SB	12.2(1st)SXH	12.2(12th)SG	12.2(6th)SE	
CEF-MIB	12.2(31)SB	12.2(1st)SRC	12.2(31)SB	12.2(31)SB	12.2(31)SB	HD	12.2(13th)SG	12.2(7th)SE	12.5(2nd)T
URPF-MIB	12.2(31)SB	12.2(1st)SRC	12.2(31)SB	12.2(31)SB	12.2(31)SB	HD	12.2(13th)SG	12.2(7th)SE	12.5(1st)T
SNMP Infrastructure for MTR	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	HD	NA	NA	NA
IP Tunnel MIB per RFC 4087	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	NA	12.5(1st)T
Interfaces MIB: SNMP Context-Based Access	12.2(1st)SB5	12.2(33)SRB	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	NA
CISCO-DATA-COLLECTION-MIB	12.2(1st)SB5	12.2(1st)SRC	NA	12.2(1st)SB5	12.2(1st)SRC	HD	12.2(12th)SG	12.2(6th)SE	12.5(2nd)T
Licensing MIB			NA				NA	12.2(37)SE	12.4(7th)T

SNMP (2/2)

Release 12.2S and T Family Roadmap

Cisco IOS Software Platforms	Cisco 10000 Series	Cisco 7600 Series	Cisco 7500 Series	Cisco 7304 Router	Cisco 7301 and 7200 Routers	Cisco Catalyst 6500 Series	Cisco Catalyst 4500 Series	Cisco 3750 and 2900 Series	
	12.2SB	12.2SR/ SX	12.2SB	12.2SB	12.2SB/SR	12.2SX/ SR	12.2SG	12.2SE	T
Event MIB and Expression MIB Enhancements			NA			HD			12.5(1st)T
HC-ALARM-MIB			NA			12.2(1st)SXI			
Show Port Status Command			NA			12.2(1st)SXI			
Infrastructure for Test Trap			NA			12.2(1st)SXI			
RMON MIB Enhancement to Support 64-Bit Counters			NA			12.2(1st)SXI			
Flash MIB Enhancements			NA			12.2(1st)SXI			
SNMP Diagnostic Enhancements			NA			HD			12.5(2nd)T

Shipping

Code Committed

EC'd

Feature Availability—NetFlow

Exporting Process

Feature	Software	C6500	C7600	C12000	C10000	C4500	Cisco IOS-XR
Version 5	12.0(1)	12.1(2)E	12.1(2)E	12.0(14)S	12.0(19)SL	12.1(13)EW	
Version 8	12.0(3)T	12.2(14)SX	12.2(14)SX	12.0(6)S	12.0(19)SL	12.1(19)EW	
Version 9	12.3	12.2(18)SXF	12.2(18)SXF	12.0(24)S	12.2(31)SB		3.2
Dual Export	12.2(2)T	12.2(17d)SXB	12.2(17d)SXB		12.2(15)BX	12.1(19)EW	
VRF Destination	12.4(4)T			12.0(26)S			
Reliable Export	12.3(4)T						
IPFIX Support	12.5(4th)T	12.2(2th)SXJ					

Product Manager contact: jgriviau@cisco.com

Available Now | Not Available | Roadmap

Feature Availability—NetFlow

Traditional NetFlow Metering Process

Feature	Software	C6500	C7600	C12000	C10000	C4500	Cisco IOS-XR
IPv4	12.0(1)	12.1(27b)E1	12.2(18)SXF	12.0(22)S	12.2(15)BX	12.1(13)EW	3.2
IPv6	12.3(7)T	12.2(33)SXH	12.2(33)SRB				3.5
Multicast	12.3	12.2(18)SXF	12.2(18)SXF				3.2
BGP Next Hop	12.3	12.2(18)SXF	12.2(33)SRA	12.0(26)S	12.2(31)SB		
Per Interface	Yes	12.2(33)SXH	12.2(33)SRB	No Sub	12.2(15)BX		3.2
TOS Support	Yes	12.2(17b)SXA	12.2(17b)SXA	Yes	Yes		3.2
Packet Sampling	12.3(24)			12.0(11)S	12.2(31)SB		
Min Prefix Aggr.	12.1(2)T			Yes	Yes		
MPLS Egress with EXP					12.2(28)SB		
MPLS Egress	12.2(2)T						3.2
MPLS Aware	12.3(8)T		12.2(33)SRA	12.0(24)S			
MPLS Label Expo	12.2SB		12.2(33)SRB				
MPLS Aggregat.					12.2(31)SB		

Available Now Not Available Roadmap

Feature Availability—NetFlow

Traditional NetFlow Metering Process

Feature	Software	C6500	C7600	C12000	C10000	C4500	Cisco IOS-XR
Egress/ Output NetFlow	12.3(11)T			12.0(10)ST	12.2(31)SB		3.2
Bridged NF		12.2(18)SXE 1	12.2(18)SXE 1			12.2(25)EW	
Input Filters	12.3(4)T						
TCP Flags	12.1(2)T			12.0(10)ST	12.2(28)SB		3.2
Mac Address	12.3(14)T						
Security Exports	12.3(14)T		12.2(33)SRA				
VLAN Export	12.4(4)T						

Available Now Not Available Roadmap

Feature Availability—NetFlow

Miscellaneous Features

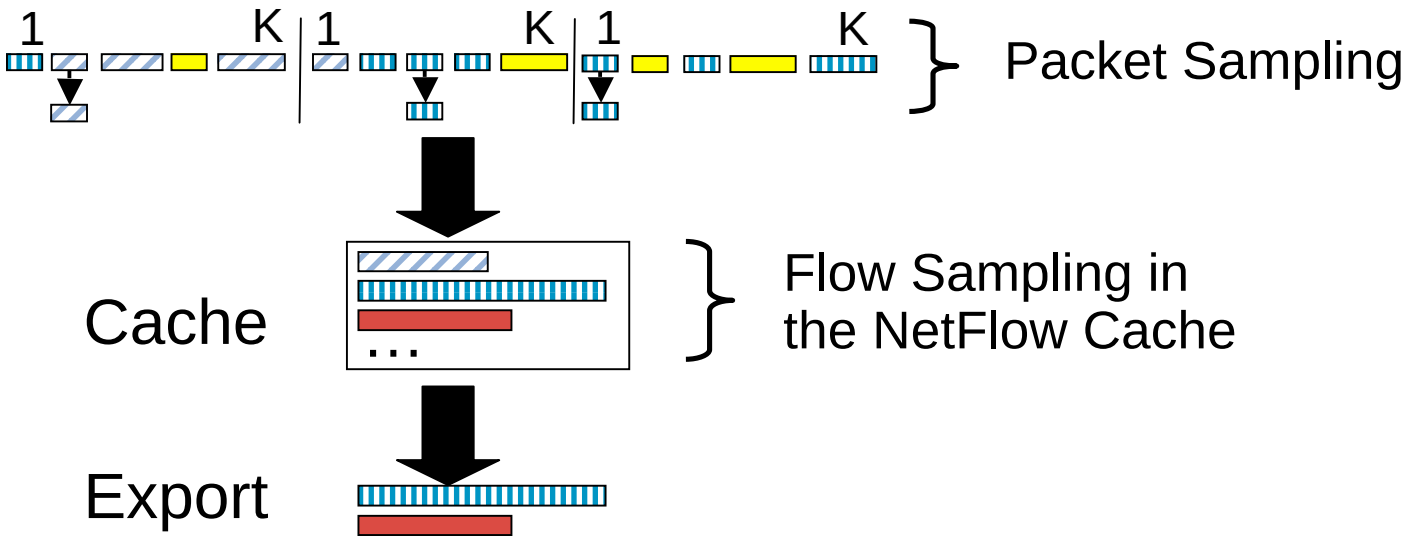
Feature	Software	C6500	C7600	C12000	C10000	C4500	Cisco IOS-XR
NetFlow MIB with Top Talker	12.3(11)T	12.2(33)SXH	12.2(33)SRB				
Dynamic Top Talker CLI	12.4(4)T						
ISSU NetFlow			12.2(33)SRB1				
ifIndex to Name Map	12.4(4)T						

Available Now | Not Available | Roadmap

Feature Availability—NetFlow

Sampled NetFlow

Feature	Software	C6500	C7600	C12000	C10K	C4500	Cisco IOS-XR
Systematic Sampling	12.3(11)T			12.0(11)S			
Random Sampling	12.4(9)T			12.0(33)S	12.2(31)SB		3.2
Output Sampled NetFlow				12.0(24)S			
Flow Sampling		12.1(13)E	12.1(13)E				



Available Now | Not Available | Roadmap

Feature Availability—NetFlow

Flexible NetFlow

Feature	Software	C6500	C7600	C12000	C10K	C4500	Cisco IOS-XR
New Flexible NetFlow CLI	12.4(9)T	12.2(1st)SXJ		12.0(33)S			3.2
Multiple User Defined Caches	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Immediate Cache	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Permanent Cache	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Header Section Export	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Payload Section Export	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Ingress Support	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Egress Support	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Random Sampling	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
Full Flow Support	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
FNF QoS Output Features	12.5(1)	12.2(1st)SXJ					
Dynamic TopNTalkers	12.5(2th)T	12.2(1st)SXJ					
MQC Integration	12.5(3th)T	12.2(1st)SXJ					

Available Now Not Available Roadmap

Feature Availability—NetFlow

Flexible NetFlow

Feature	Software	C6500	C7600	C12000	C10K	C4500	Cisco IOS-XR
NetFlow v5	12.5(2st)T	12.2(1st)SXJ					3.2
NetFlow v9	12.4(9)T	12.2(1st)SXJ		12.0(33)S			3.2
Reliable Export (SCTP)	12.5(4th)T	12.2(2th)SXJ					
IPv4 Unicast Flows	12.4(9)T	12.2(1st)SXJ		12.0(33)S			
IPv4 Predefined Aggregations	12.4(9)T	12.2(1st)SXJ		12.0(33)S			3.2
IPv6 Unicast Flows	12.5(1st)	12.2(1st)SXJ					
IPv6 Predefined Aggregations	12.5(1st)	12.2(1st)SXJ					3.5
IPv4 Multicast Flows	12.5(2th)T	12.2(1st)SXJ		12.0(33)S			
IPv6 Multicast Flows	12.5(3th)T	12.2(1st)SXJ					
Layer 2 Flows	12.5(2th)T	12.2(1st)SXJ					
MPLS Flows	12.5(4th)T	12.2(2th)SXJ					
NBAR Integration	12.5(3th)T						

Available Now | Not Available | Roadmap

Appendix II:
E-DI Scripting Example

Deployment & Activation

Example: E-DI Perl Scripting – 1/3

```
#################################################################
# Cisco Enhanced Device Interface (E-DI)
#
# perl script to include devices in a network group based on
# matching OS Version and running config line
#
# bklauser@cisco.com
#################################################################
use lib '/perlapi';
use EDIPERLAPI;
$| = 1;                              # auto flush
$boDebug = 0;                        # debug flag
#################################################################
# get command line parameters and open API
#################################################################
if ($boDebug) {
  foreach (@ARGV) {
    print "\n $_ \n";
  }
}
if ($#ARGV =~ 3) {
  $inGroup = @ARGV[0];
  $inOSVersion = @ARGV[1];
  $inCfgMatch = @ARGV[2];
  $outGroup = @ARGV[3];
} else {
  die "usage: # perl GroupByCfgMatch.pl input-group version-criteria config-string output-group\n";
}
my $api= EDIPERLAPI->getAPI();
my @matchingDevices;
#################################################################
# set context to input group
#################################################################
($retcode, $cmdout) = $api->executeCMD ("network group $inGroup");
if ($retcode) {
  die " Could not change context to network group $inGroup \n $cmdout \n";
} elsif ($boDebug) {
  print " Set context to network group $inGroup \n";
}
```

Deployment & Activation
Example: E-DI Perl Scripting – 2/3

```
#################################################################
# identify matching devices
#################################################################
($retcode, $cmdout) = $api->executeCMD ("show devices");
if ($retcode) {
  die " Could not list devices in group $inGroup \n $cmdout \n";
} elsif ($boDebug) {
  print " Listed devices in group $inGroup \n";
}
@devlist = split (/\n/,$cmdout);    # Taking ouput into an array
foreach $line (@devlist) {
  if (defined($line)){
    $line=~m/(\s+|\S\s)(\d+\.\d+\.\d+\.\d+)(.*)/;
    if ($1 !~m/\*/) {               # if it is a supported device
      if (defined($2)) {
        $device=$2;
        ($retcode, $cmdout) = $api->executeCMD ("network $device");
        if ($retcode) {
          print " Could not change context to device $device \n $cmdout \n";
          next;
        } elsif ($boDebug) {
          print " Set context to device $device \n";
        }
        ($retcode, $cmdout) = $api->executeCMD ("show report software");
        if ($retcode) {
          print " Could not show software report of device $device \n $cmdout \n";
          next;
        }
        $cmdout =~m/(\s)(\d+\.\d+\.\d+\.\d+)(\s+)(\w+)(\s+)(\w+)(\s+)(\d+\.\d+)(.*)/;
        $curDevOSFam = "$6";
        $curDevOSMVer = "$8";
        if ($boDebug) {
          print " Device $device OS Family: $curDevOSFam\n";
          print " Device $device OS Major Version. $curDevOSMVer\n";
        }
        if (($curDevOSFam =~m/IOS/) && ($curDevOSMVer =~m/$inOSVersion/)) {
          ($retcode, $cmdout) = $api->executeCMD ("show running-config | include \"$inCfgMatch\"");
          if ($retcode) {
            print " Could not show running-config of device $device \n $cmdout \n";
            next;
```

Deployment & Activation
Example: E-DI Perl Scripting – 3/3

```
          } elsif ($boDebug) {
            print " Show running-config returns: $cmdout \n";
          }
          if ($cmdout =~/$inCfgMatch/) {        # we have a match
            push (@matchingDevices, $device);
          }
        }
      }
    }
  }
}
#################################################################
# create output group and include members
#################################################################
($retcode, $cmdout) = $api->executeCMD ("server config");
if ($retcode) {
  die " Could not enter server config mode \n $cmdout \n";
} elsif ($boDebug) {
  print " Entered server config mode \n";
}
($retcode, $cmdout) = $api->executeCMD ("static-group $outGroup");
if ($retcode) {
  die " Could not change context to network group $outGroup \n $cmdout \n";
} elsif ($boDebug) {
  print " Set context to network group $outGroup \n";
}
foreach $device (@matchingDevices) {
  ($retcode, $cmdout) = $api->executeCMD ("include device $device");
  if ($retcode) {
    die " Could not add device $device to group $outGroup \n $cmdout \n";
  } else {
    print " Added device $device to group $outGroup \n";
  }
}
$api->closeAPI();
1;
#################################################################
# EOF
#################################################################
```