



## Service Diagnostics 2.0 – Connectivity Fault Management (CFM) Scripts

12.2(52)SE TAC TOI



**09/15/2009**

**Gaurang Mokashi**

**Jose Liste**

**EDCS-804969**

# Agenda

- Service Diagnostics Overview (5 min)
- CFM Scenarios Overview (20 min)
  - CFM Overview
  - Diagnostic Scenarios
  - Action Scenarios
- Deliverables (15 min)
  - User Interface Package Contents
  - Common Deliverables (Helper scripts, libraries, CFM MDF)
  - Deployment Options (CLI and MDF)
- CFM Scenarios (30 min)
  - Deliverables (CFM EEM policies), Configuration
  - Deployment Considerations
  - Demonstration (deployment, configuration and trigger)
- Troubleshooting (10 min)
  - Debug Commands
  - Common Issues
- Q&A (10)

An estimate of 90 minutes are required for this topic

# Service Diagnostics Overview



# Service Diagnostics

## Introduction

- **Event-based embedded network management** to help solve common diagnostics scenarios
- Collection of scripts with the following attributes:
  - Developed, fully tested, documented and supported by Cisco
  - Leverage Cisco IOS embedded capabilities: **Embedded Event Manager (EEM)** and **Embedded Resource Manager (ERM)**
  - User-friendly menu-driven installation and configuration through XML-based Menu Definition File (MDF) using IOS **Embedded Menu Manager (EMM)**
  - Diagnostic results notification through Syslogs / SNMP Traps / Email
  - User-customizable
  - FREE**
  - Digitally Signed
- Scripts available at:

<http://www.cisco.com/go/iossd>

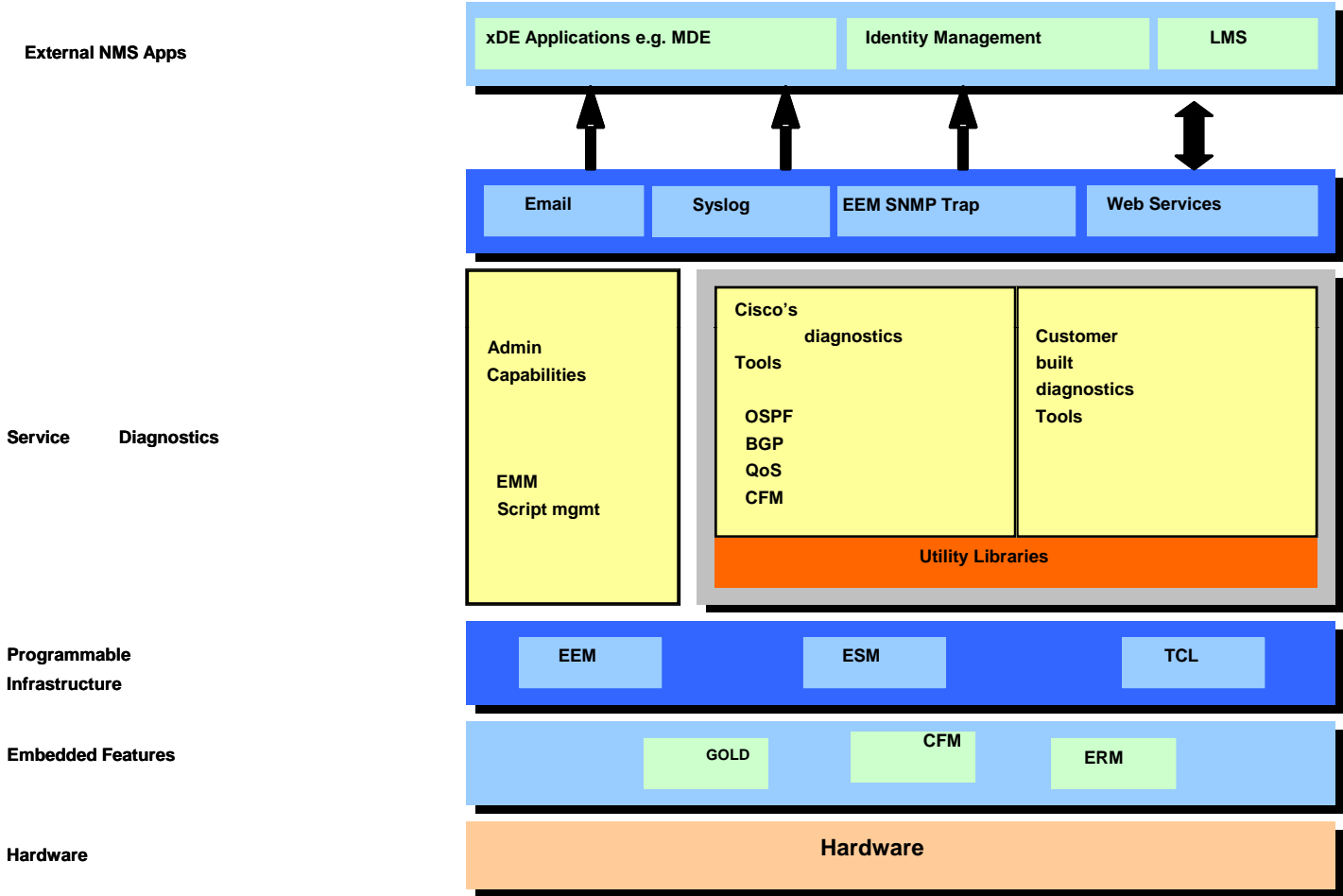
# Service Diagnostics

## Introduction (cont.)

- Scripts delivered in phases:
- Phase 1 – **Shipping**
  - Diagnostics scenarios related to OSPF, BGP, QoS and Memory/CPU-resources (12 scripts)
- Phase 2
  - Diagnostics scenarios related to Carrier Ethernet using IEEE 802.1ag (Connectivity Fault Management – CFM)
  - Beachhead platform: ME-3400 / ME-3400E
  - FCS inline with IOS 12.2(52)SE FCS (Q3CY2009)

# Service Diagnostics

## Feature Interaction



# CFM Scenarios Overview



# Service Diagnostics

## Carrier Ethernet Scenarios

- Diagnose ethernet service failures using IEEE 802.1ag (Connectivity Fault Management – CFM)
- Designed to support standard CFM IOS implementation  
Pre-standard CFM IOS implementation not supported
- Access / Demarcation customer premise equipment selected as beachhead for development and testing  
Cisco ME-3400 / ME-3400E

# Service Diagnostics

## Carrier Ethernet Scenarios (cont.)

- Scenarios broken into two (2) categories:

Diagnostics

Actions

- Value add:

Automates CFM troubleshooting and thus acting as a Level 1 NOC person

Eliminates chance for human errors while troubleshooting (e.g. mistyping MAC address)

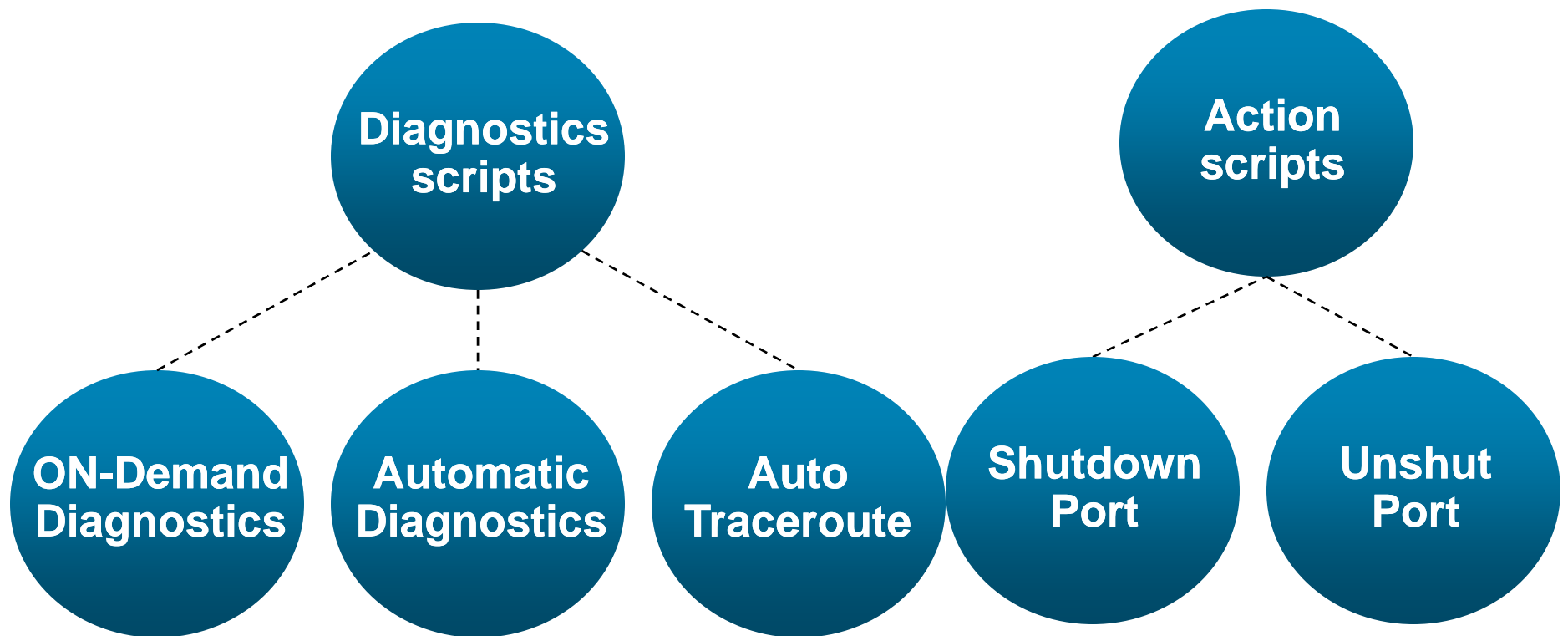
Makes use of traceroute cache and searches through an otherwise hard to sort output

Reduce CE blackholing of CE traffic

“Brute Force” method to notify CE of EVC or remote UNI failures

# Service Diagnostics

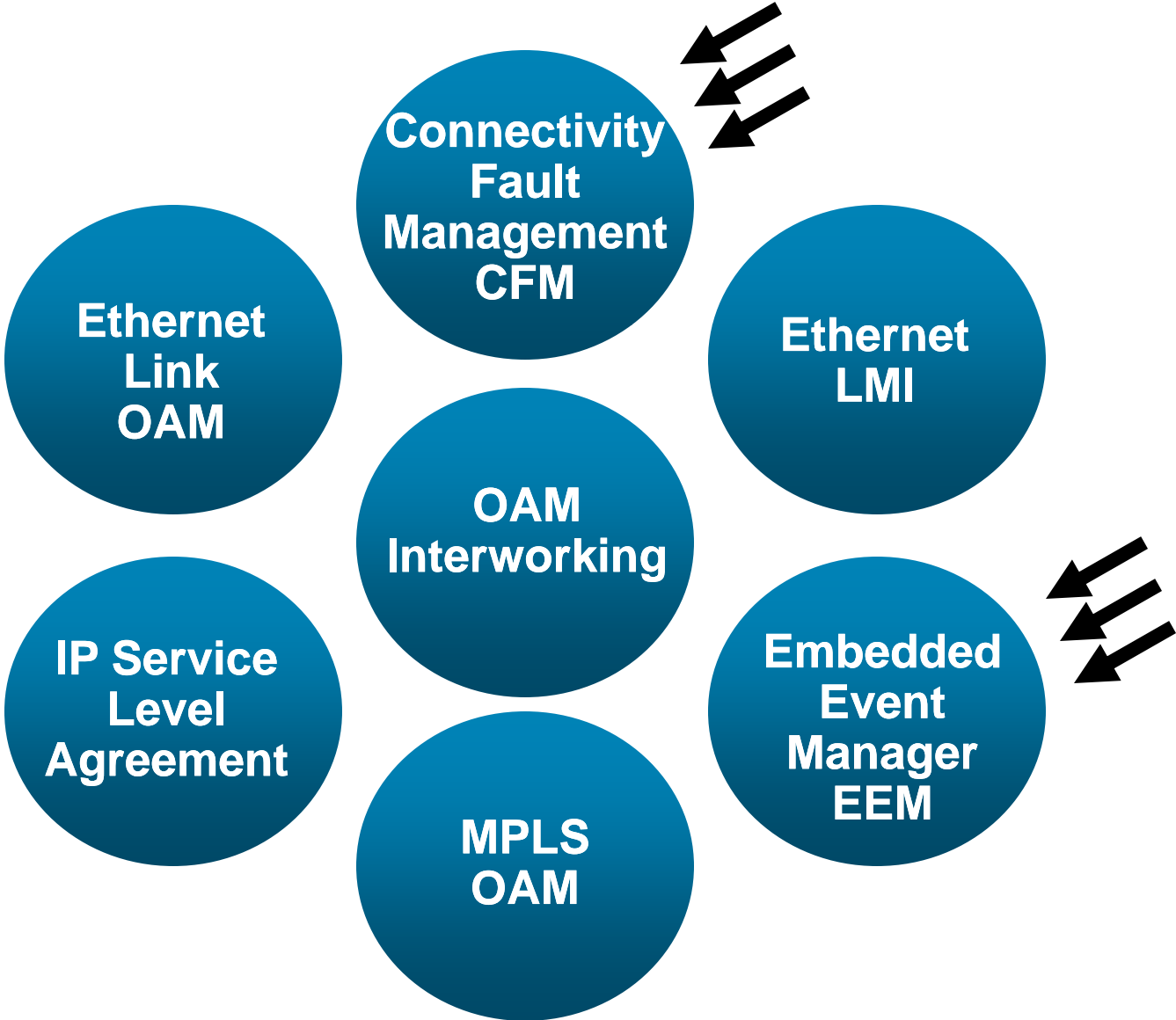
## Carrier Ethernet Scenarios (cont.)



# CFM Overview

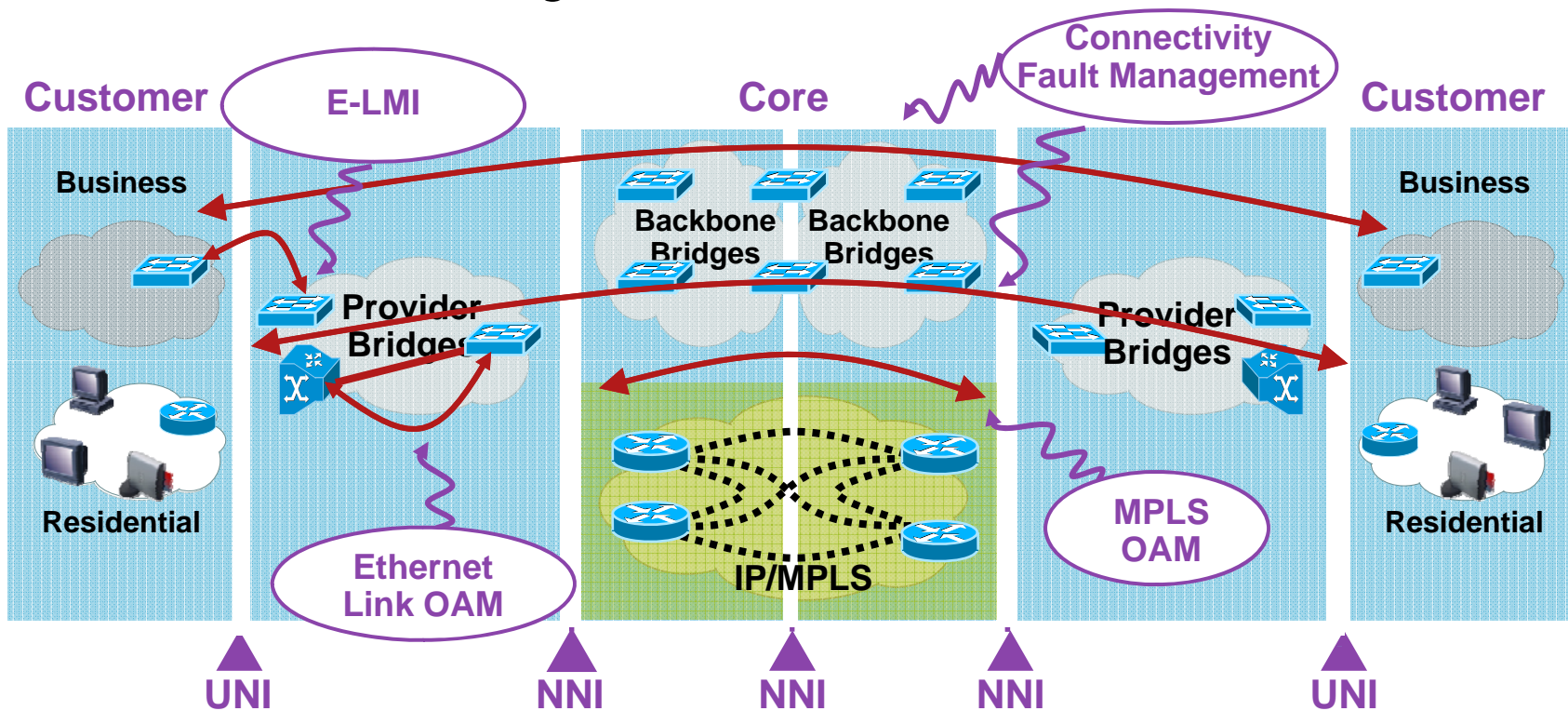


# Cisco Carrier Ethernet OAM



# Cisco Carrier Ethernet OAM

## Protocol Positioning



- E-LMI - User to Network Interface (UNI)
- Link OAM - Any point-point 802.3 link
- CFM - End-to-End UNI to UNI
- MPLS OAM - within MPLS cloud

# Connectivity Fault Management

## Overview

- IEEE standard (IEEE 802.1ag-2007)
- Family of protocols that provides capabilities to detect, verify, isolate and report end-to-end ethernet connectivity faults
- Employs regular Ethernet frames that travel in-band with the customer traffic

Devices that cannot interpret CFM Messages forward them as normal data frames

- Defines the following concepts:

Maintenance Domain (MD) [MD name, MD Level]

Maintenance Association (MA) [ShortMA name]

Maintenance Association End Points (MEP) [mpid]

Maintenance Domain Intermediate Points (MIP)

# Connectivity Fault Management

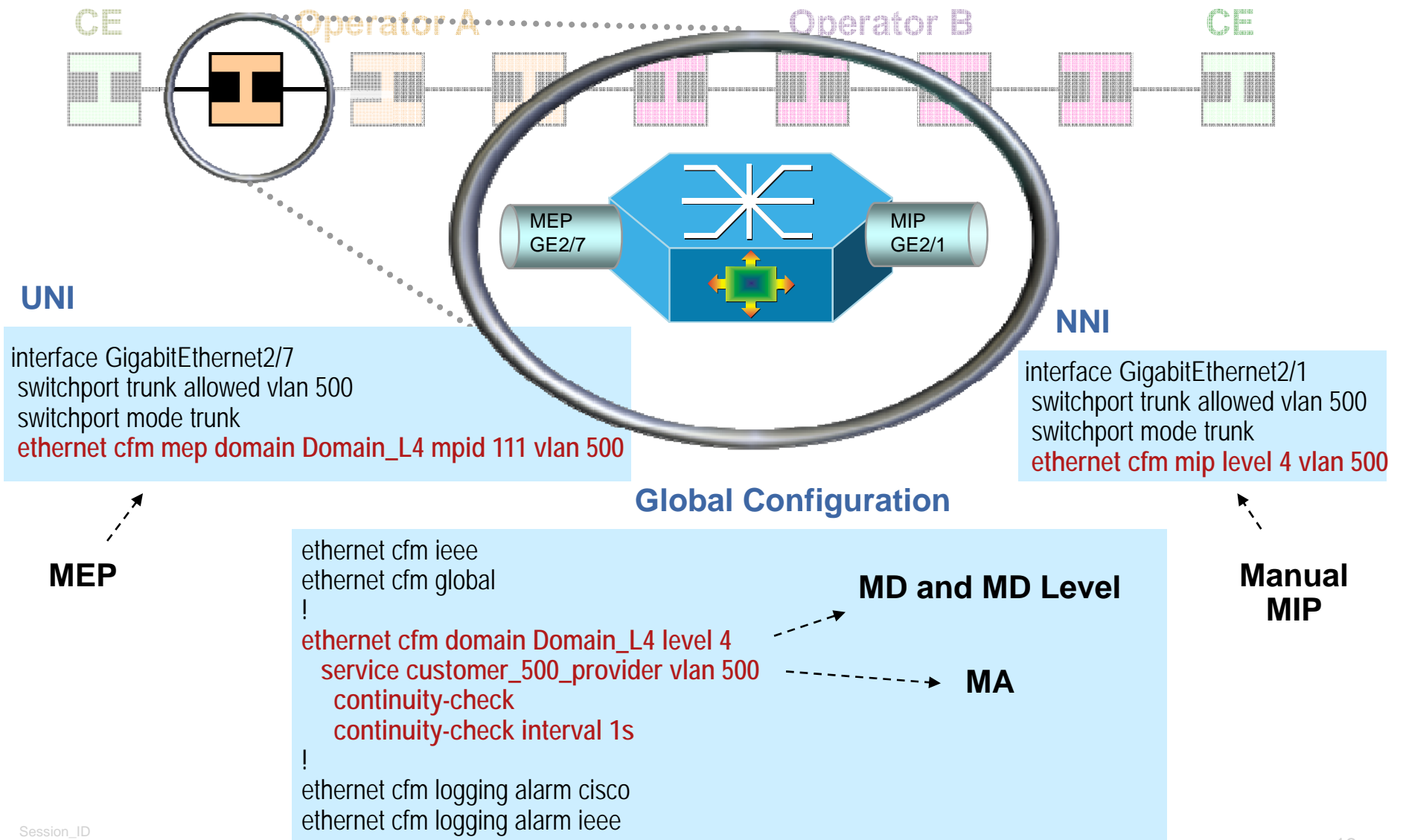
## Protocols

- There are three (3) protocols defined by CFM
- **Continuity Check** Protocol
  - Fault Detection
  - Fault Notification
- **Loopback** Protocol
  - Fault Verification
- **Linktrace** Protocol
  - Fault Isolation

# Cisco CFM Implementation

## End-to-End CFM on Switchport Example

Standard CFM Implementation



# Diagnostic Scenarios



# Service Diagnostics

## Carrier Ethernet Scenarios - Diagnostics

- ON-DEMAND Diagnostics

Upon customer request to troubleshoot a given service, EEM script automatically verifies and isolates faults to remote MEPs impacted by service failure. It could also search traceroute-cache. Diagnostics report sent to NOC

- AUTOMATIC Diagnostics

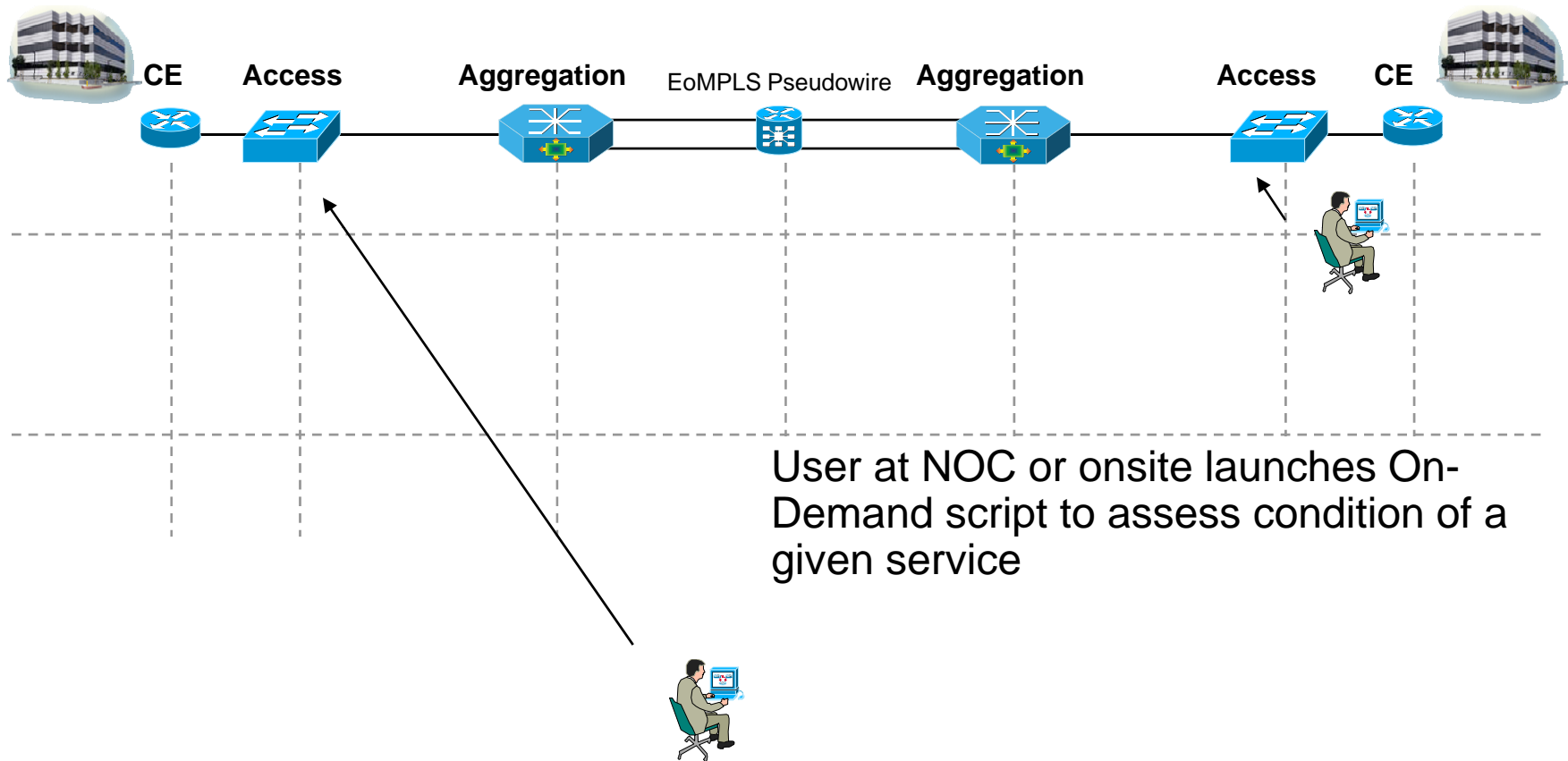
Upon a Service failure, EEM script automatically verifies and isolates faults to the remote MEPs impacted. Diagnostics report sent to NOC

- AUTO-TRACEROUTE

Upon timer expiration, EEM script automatically traceroutes to remote MEPs in CCDB (user may select the list of Services). Summary report sent to NOC

# Service Diagnostics

## ON-Demand Diagnostics



# Service Diagnostics

## ON-Demand Diagnostics (cont.)

EEM event detector: none  
Script variables:  
CFM MD name, shortMA name,  
remote mep id (opt)

### Parse Collect Data

Initialize these variables:  
• CFM Domain name  
• CFM Domain level  
• S-VLAN ID

### Path Discovery Isolate Failure (trace ethernet mac)

IF (remote mpid provided)  
THEN issue LTM towards it  
ELSE  
Issue LTM to all rmeps found in CCDB / Er  
Record results

### Inspect CFM CCM DB

IF (remote mpid provided)  
THEN examine MEP CCDB for entry logged against it  
ELSE  
Examine MEP CCDB for entries logged for the entire svc  
Record remote MAC addresses and remote IF states

### Inspect Traceroute cache

For failed traceroute operations:  
Inspect cache for last successful operation  
Record Missing hops

### Inspect CFM ErrorDB

IF (remote mpid provided)  
THEN look for errors logged against it  
ELSE  
look for errors logged for the entire service

### Generate Report

Place together the following information

- Number of active MEPs in CCDB with IF state == UP
- Details of active MEPs in CCDB with IF state <> UP
- Details of entries in Error DB
- Number of remote MEPs with verified connectivity
- Details of remote MEPs with connectivity problems
- Traceroute results for MEPs with connectivity problems
- Results of traceroute cache examination

### Connectivity / Failure Verification (ping ethernet mac)

IF (remote mpid provided)  
THEN issue LBM towards it and meps in ErrDB  
ELSE  
issue LBM to all rmeps found in CCDB and ErrorDB  
Record results

# Service Diagnostics

## ON-Demand Diagnostics Details



For Your Reference

EEM event detector: none  
 Script variables:  
 CFM MD name, shortMA name,  
 remote mep id (opt)

**Parse  
Collect Data**

show ethernet cfm domain zzz

**Inspect CFM  
CCM DB**

show ethernet cfm maintenace remote domain zzz  
 show ethernet cfm maintenace remote detail  
 mpid iii domain zzz vlan xxx

**Inspect CFM  
ErrorDB**

show ethernet cfm error domain-id zzz service www

**Connectivity /  
Failure Verification  
(ping ethernet mac)**

ping ethernet H.H.H domain zzz vlan xxx  
 ping ethernet multicast domain zzz vlan xxx

**Path Discovery  
Isolate Failure  
(trace ethernet mac)**

traceroute ethernet H.H.H  
 domain zzz vlan xxx

**Inspect  
Traceroute cache**

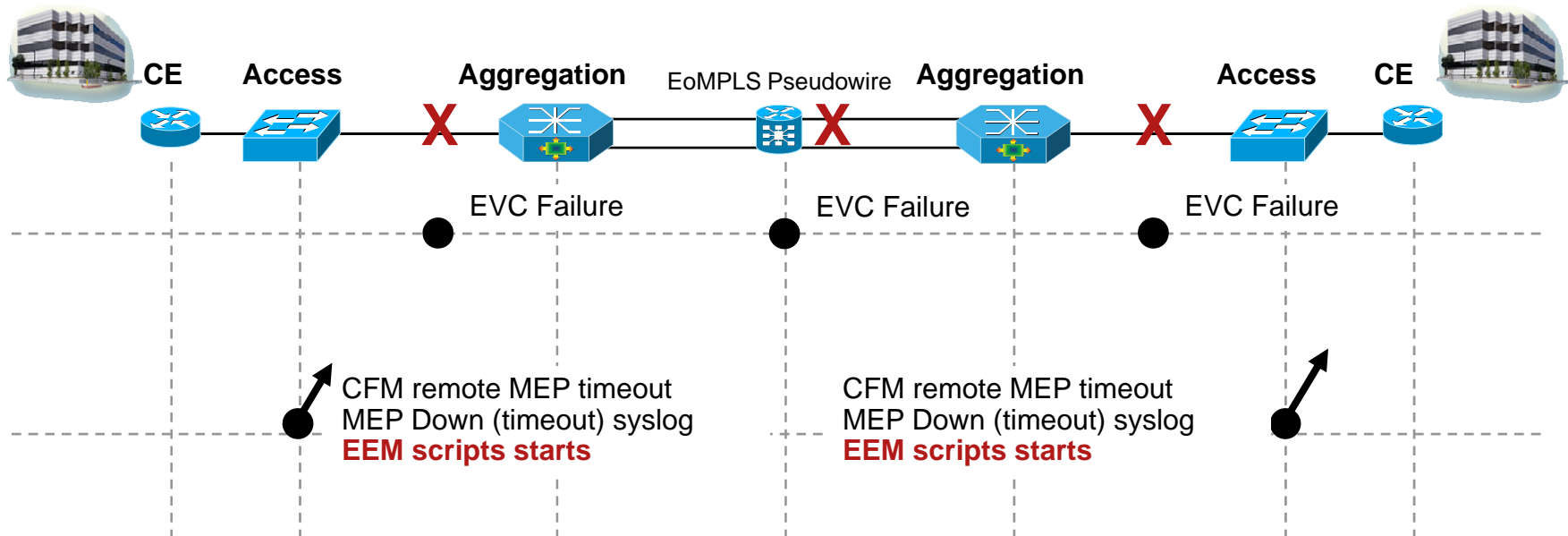
show ethernet cfm traceroute-cache

**Generate  
Report**

- Place together the following information
- Number of active MEPs in CCDB with port state == UP
  - Details of active MEPs in CCDB with port state <> UP
  - Details of entries in Error DB
  - Number of remote MEPs with verified connectivity
  - Details of remote MEPs with connectivity problems
  - Traceroute results for MEPs with connectivity problems
  - Results of traceroute cache examination

# Service Diagnostics

## Automatic Diagnostics

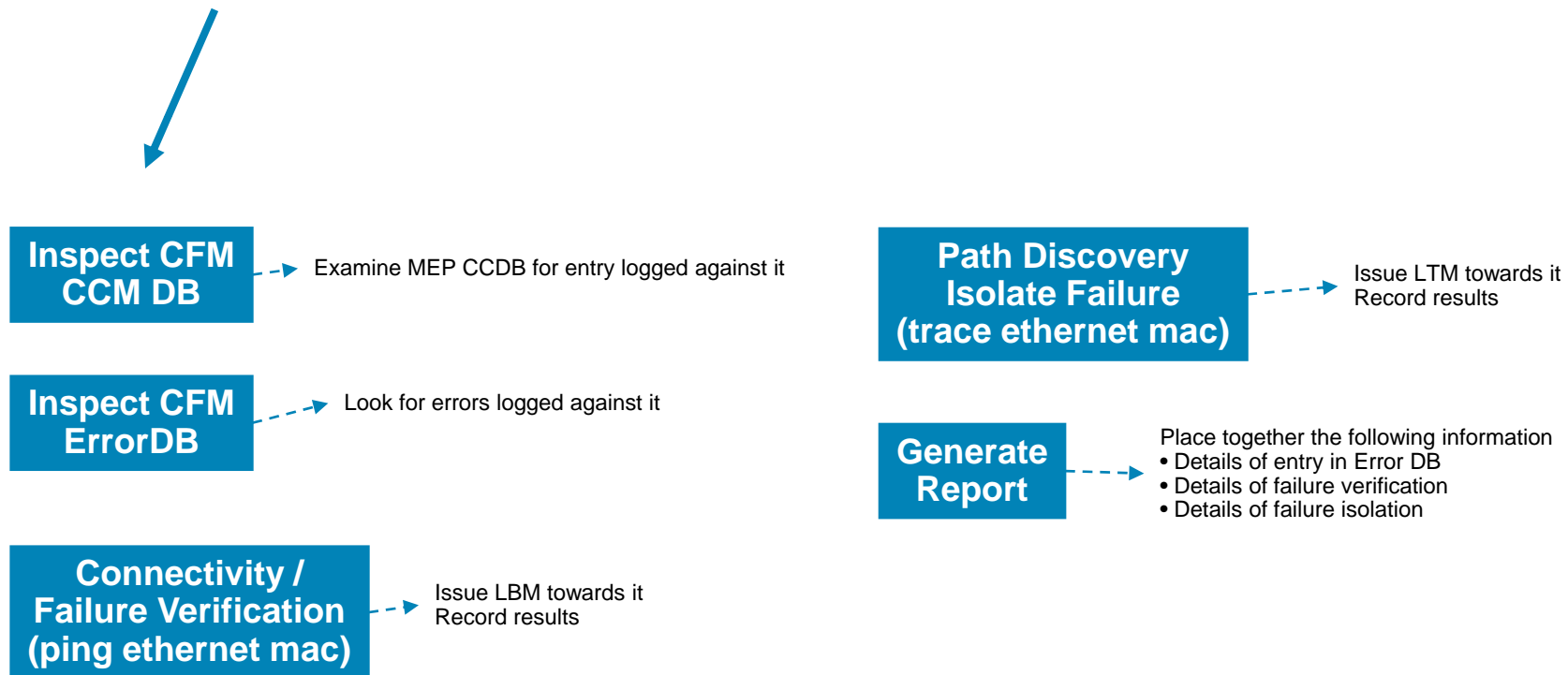


NO User intervention. Script runs when a remote MEP timeout is detected on selected services

# Service Diagnostics

## Automatic Diagnostics (cont.)

EEM event detector: CFM MEP Down (timeout) syslog<sup>1</sup>  
Script variables:  
Input file with list of shortMA names



# Service Diagnostics

## Automatic Diagnostics Details



EEM event detector: CFM MEP Down (timeout) syslog<sup>1</sup>  
Script variables:  
Input file with list of shortMA names

**Inspect CFM  
CCM DB**

show ethernet cfm maintenace remote detail  
mpid iii domain zzz vlan xxx

**Inspect CFM  
ErrorDB**

show ethernet cfm error domain-id zzz service www

**Connectivity /  
Failure Verification  
(ping ethernet mac)**

ping ethernet H.H.H domain zzz vlan xxx

**Path Discovery  
Isolate Failure  
(trace ethernet mac)**

traceroute ethernet H.H.H  
domain zzz vlan xxx

**Generate  
Report**

Place together the following information

- Details of entry in Error DB
- Details of failure verification
- Details of failure isolation

# Service Diagnostics

## Auto-Traceroute

EEM event detector: timer expiration  
Script variables:  
Input file with list of shortMA names  
Timer value (default e.g. 6 hours)

Verify if cache  
is enabled

Verify if traceroute cache is enabled  
Verify the size and the hold-time of traceroute cache

Parse  
CFM CCM DB

Initialize these variables for the desired MAs:  
• cfm Domains  
• rmep mpids  
• rmep MAC addresses  
• vlan number

Path Discovery  
(trace ethernet mac)

Issue LTM for each of the rmeps  
discovered

Generate  
Report

Place together the following information  
• Total number of traceroute operations performed  
• rmep details

# Service Diagnostics

## Auto-Traceroute Details



For Your Reference

EEM event detector: timer expiration  
Script variables:  
Input file with list of shortMA names  
Timer value (default e.g. 6 hours)

Verify if cache is enabled

show ethernet cfm traceroute-cache

Parse CFM CCM DB

show ethernet cfm domain brief  
show ethernet cfm domain <domain>  
show ethernet cfm maintenance-points remote  
show ethernet cfm maintenance-points remote domain <domain>

Path Discovery (trace ethernet mac)

traceroute ethernet H.H.H domain zzz vlan xxx

Generate Report

Place together the following information

- Total number of traceroute operations performed
- rmep details

# Action Scenarios



# Service Diagnostics

## Carrier Ethernet Scenarios - Actions

- Shutdown Port

In point-to-point environments with NON-ELMI capable CEs, reduce CE blackholing of traffic when EVC or remote UNI failures occur by performing Shutdown of UNI interface associated with an UP MEP

- UNshut Port

Upon EVC or remote UNI recovery, bring UP an interface (no shut) previously shutdown by the Shutdown Port script

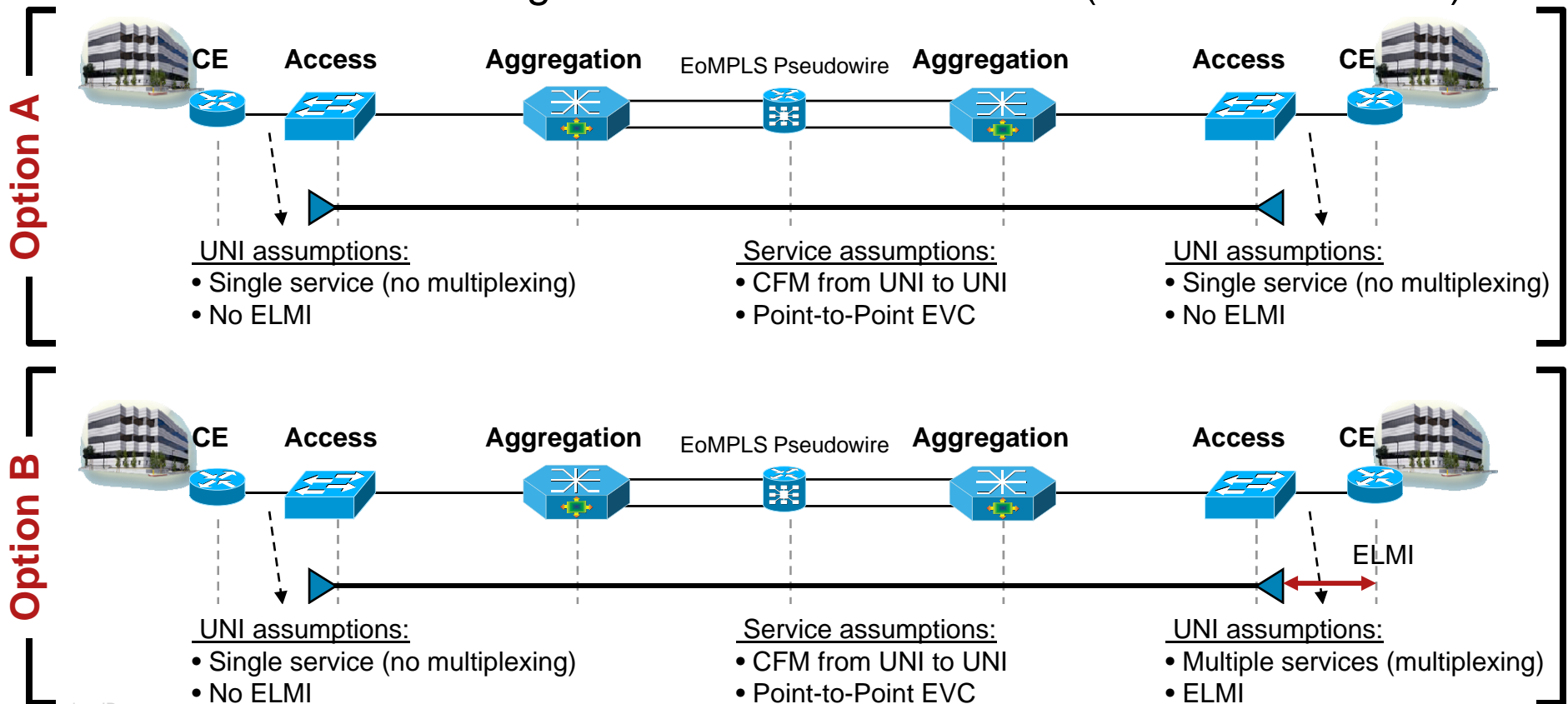
# Service Diagnostics

## Shutdown Port

- TWO scenarios to be considered

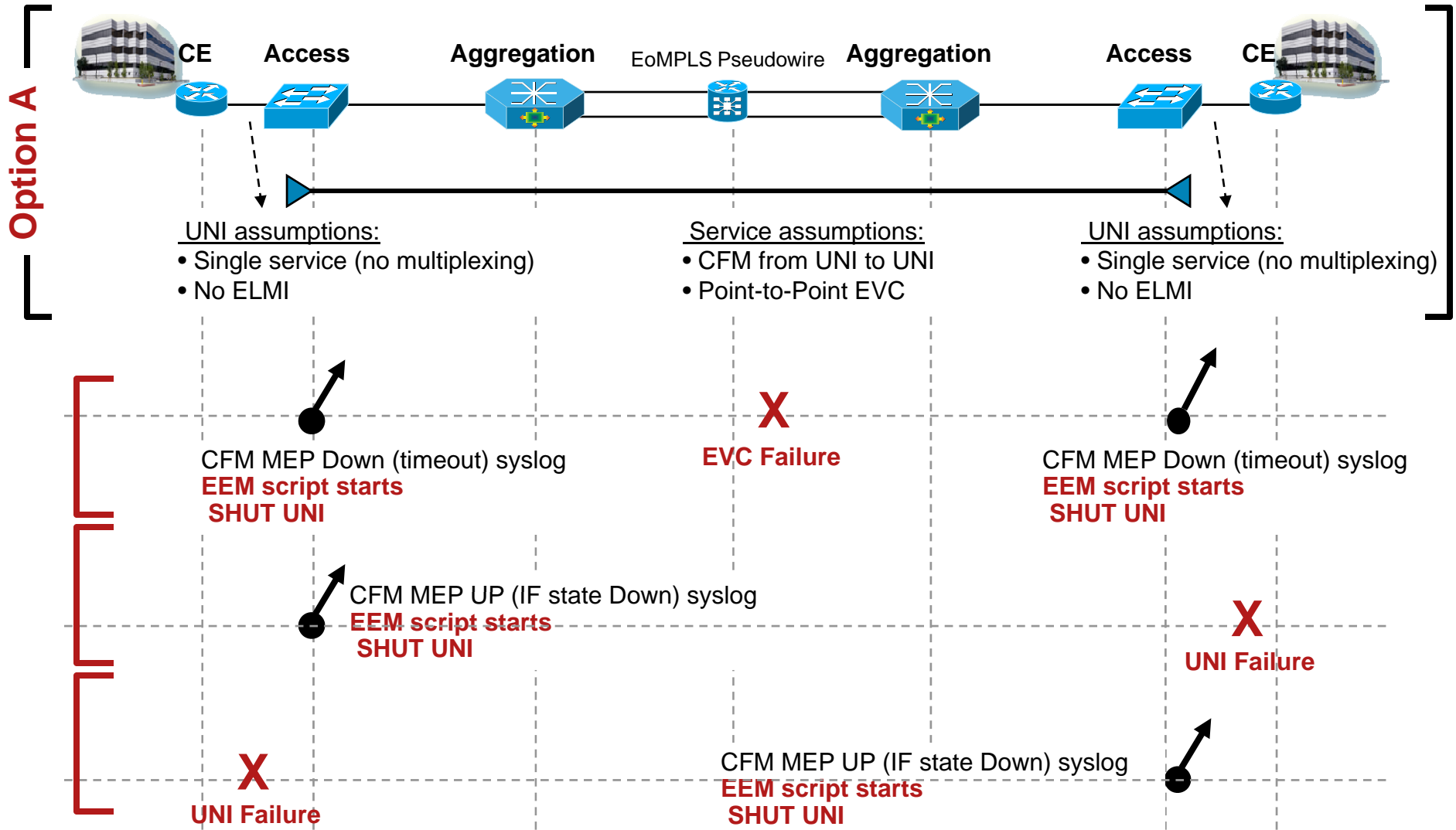
No ELMI and single service UNI on both sides

No ELMI and single service UNI on one side (ELMI on the other)



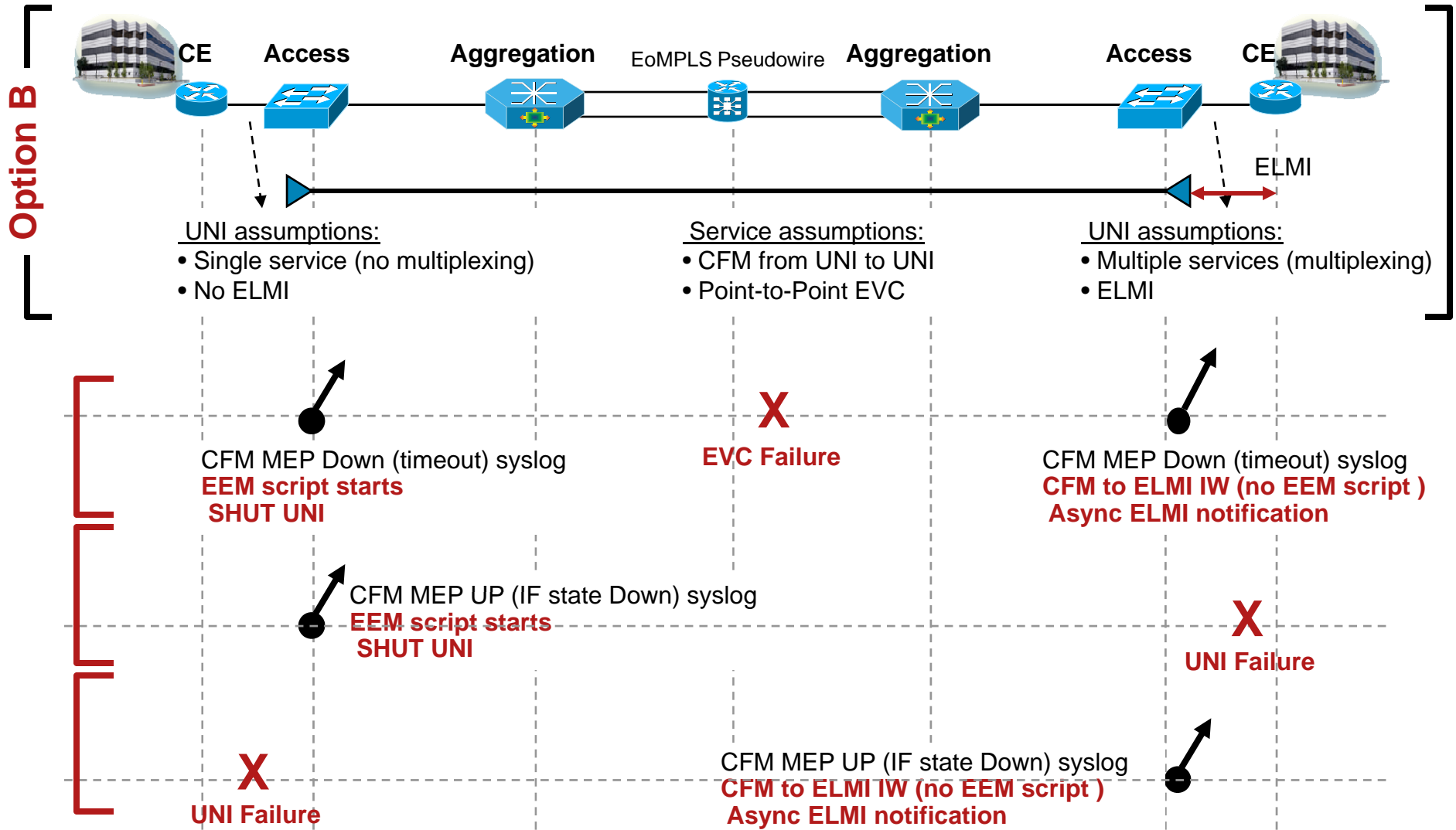
# Service Diagnostics

## Shutdown Port (cont.)



# Service Diagnostics

## Shutdown Port (cont.)



# Service Diagnostics

## Shutdown Port (cont.)

EEM event detector: CFM MEP Down (timeout) syslog (1)  
CFM MEP Up (port state Down) syslog (2)

Script variables:  
Input file with list of shortMA names

**Parse  
Collect Data**

Initialize these variables:  
• CFM Domain name  
• S-VLAN ID

**Generate  
Report**

Send message indicating the MA name and corresponding Interface that was shutdown by the script

**Inspect CFM  
CCM DB**

IF MEP UP IF state syslog  
THEN verify if IF State == DOWN  
continue next step  
Note: if remote IF state == AdminDown do nothing

IF MEP Down (timeout) syslog  
THEN continue next step

**Inspect CFM  
Local DB**

Examine CFM maintenance point local DB  
to determine the interface that holds the mep associated  
with the service under monitoring

**SHUT  
Port**

Shutdown Port

**Record  
MA and  
Interface shut  
for recovery**

Record remote mpid / shortMA name / MD name / Vlan information  
This will be used later to perform UNshut during recovery

# Service Diagnostics

## Shutdown Port Details



EEM event detector: CFM MEP Down (timeout) syslog (1)  
CFM MEP Up (port state Down) syslog (2)  
Script variables:  
Input file with list of shortMA names

**Parse  
Collect Data**

**Generate  
Report**

Send message indicating the MA name and corresponding Interface that was shutdown by the script

**Inspect CFM  
CCM DB**

show ethernet cfm maintenance-points remote  
detail mpid <mpid> domain <zzz> vlan <xxx>

**Inspect CFM  
Local DB**

show ethernet cfm maintenance-points local mep domain <zzz>

**SHUT  
Port**

conf t  
interface gigx/y  
shutdown

**Record  
MA and  
Interface shut  
for recovery**

Record remote mpid / shortMA name / MD name / Vlan information  
This will be used later to perform UNshut during recovery

# Service Diagnostics

## Unshut Port

EEM event detector: CFM MEP Up (new) syslog (1)  
CFM MEP Up (returning) syslog (2)  
CFM MEP Up (port state UP) syslog (3)

Script variables:  
Interfaces previously shut by SHUTDOWN script

**Parse  
Collect Data**

Initialize these variables:  
• CFM Domain name  
• S-VLAN ID

**Inspect CFM  
CCM DB**

Verify if IF State == UP or ADMIN DOWN  
IF yes THEN continue next step  
  
Confirm interface ID of local MEP and verify its state  
IF shutdown THEN continue next step

**NO SHUT  
Port**

Proceed to bring UP (unshut) the corresponding interface

**Generate  
Report**

Verify that interface is UP  
  
Send message indicating the MA name and corresponding  
Interface that was unshut by the script

# Service Diagnostics

## Unshut Port Details



For Your Reference

EEM event detector: CFM MEP Up (new) syslog (1)  
CFM MEP Up (returning) syslog (2)  
CFM MEP Up (port state) syslog (3)

Script variables:  
Interfaces previously shut by SHUTDOWN script

Parse  
Collect Data

Inspect CFM  
CCM DB

```
show ethernet cfm maintenance-points remote  
detail mpid <mpid> domain <zzz> vlan <xxx>
```

```
show ethernet cfm maintenance-points local mep domain <zzz>
```

```
show interface <id>
```

NO SHUT  
Port

```
conf t  
interface gigx/y  
no shutdown
```

Generate  
Report

```
show interface <id>
```

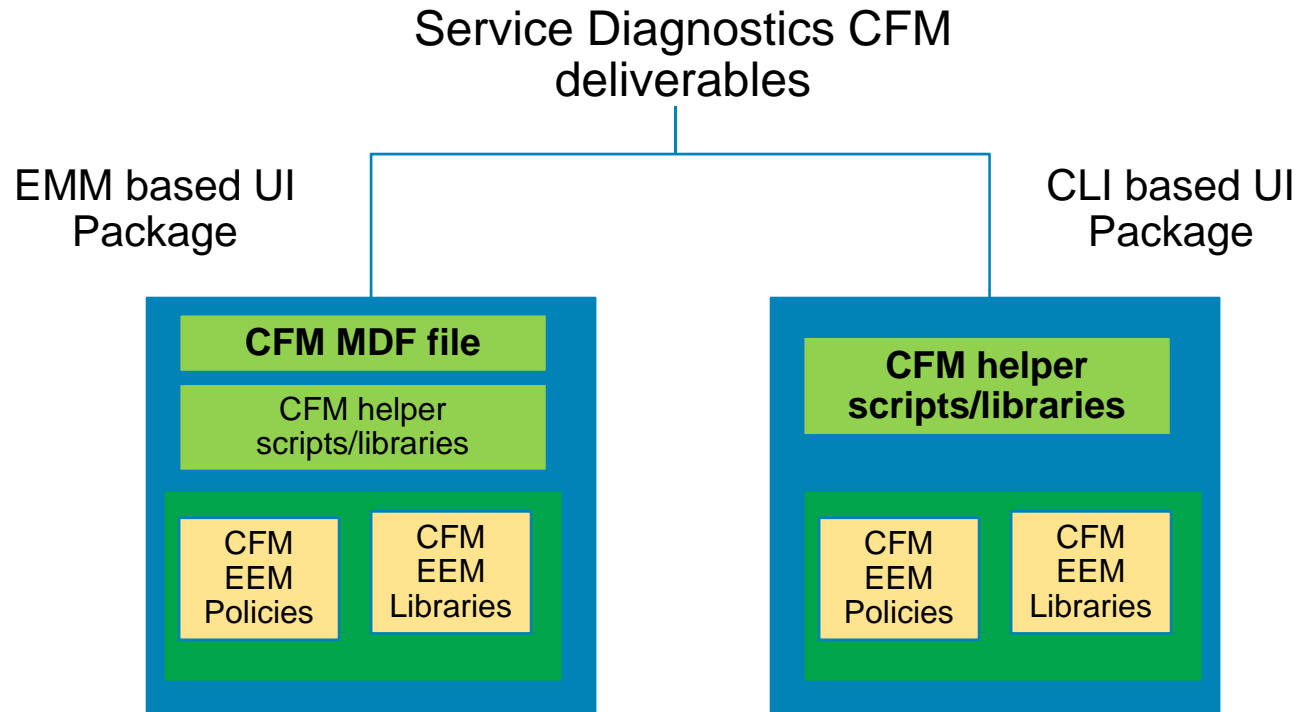
Send message indicating the MA name and corresponding Interface that was unshut by the script

# Deliverables



# Types of Deliverables

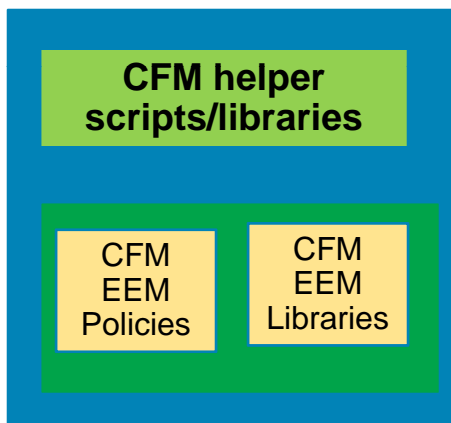
Based on User Interface (UI)



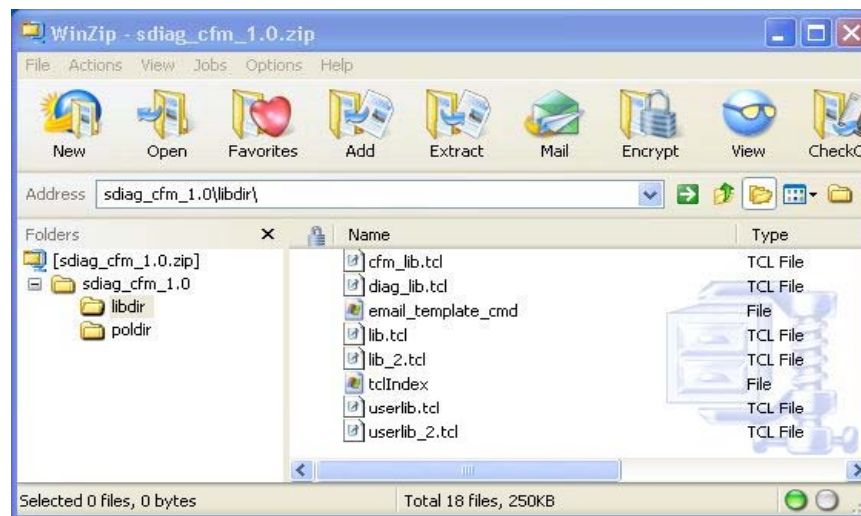
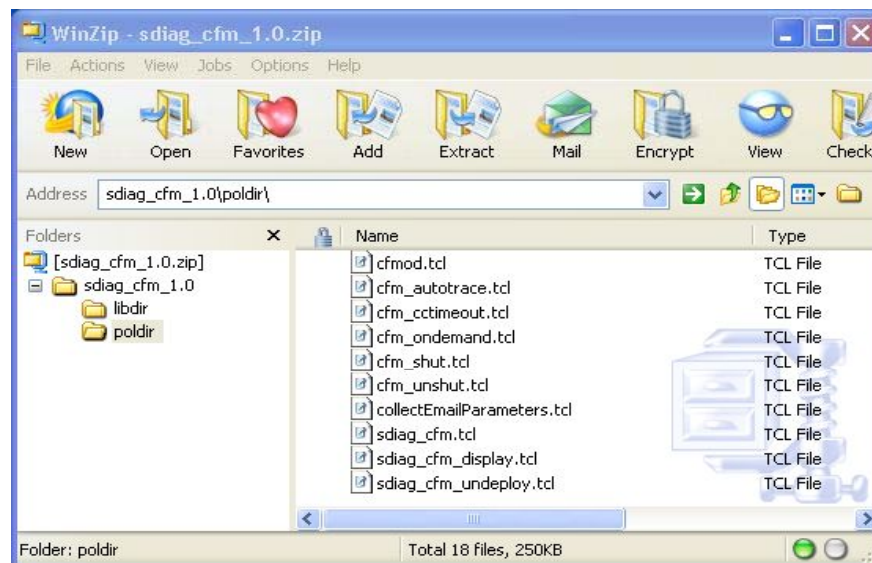
# CLI Based UI package contents

Policy directory  
(Helper scripts + CFM EEM policies)

CLI based UI Package



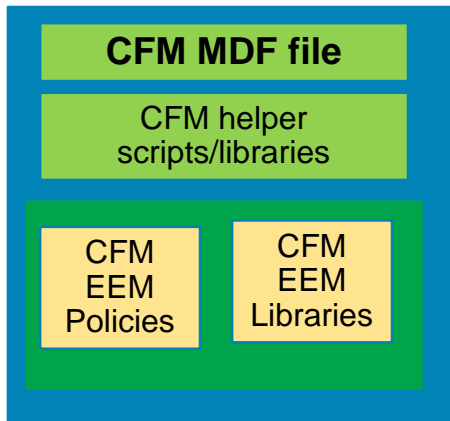
Library directory  
(Helper libraries + CFM EEM libraries)



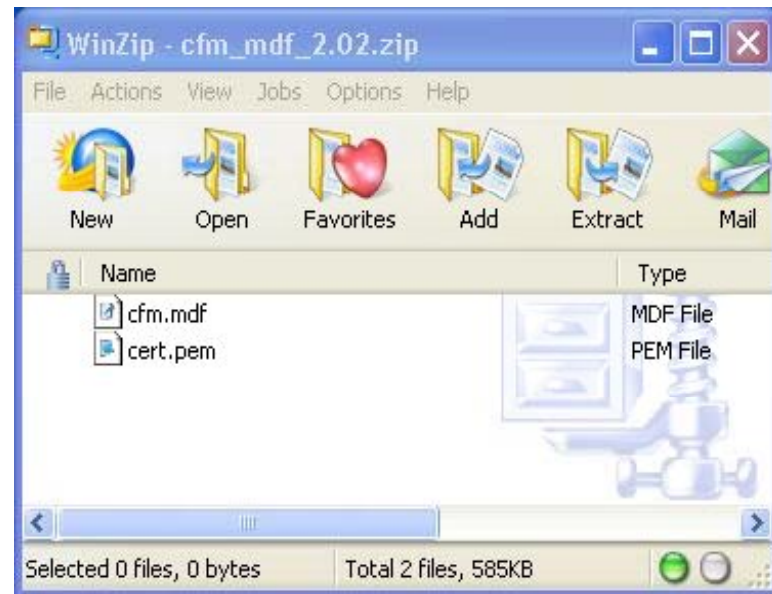
# EMM Based UI package contents

## CFM MDF

EMM based UI Package



Contents of the EMM based UI package



# CFM helper scripts

- `sdiag_cfm.tcl`
- `sdiag_cfm_display.tcl`
- `sdiag_cfm_undeploy.tcl`
- `cfmod.tcl`
- `collectEmailParameters.tcl`

# CFM helper scripts contd.

## sdiag\_cfm.tcl

Deploy CFM policies (register with EEM), configure notifications (Email, syslog and SNMP) and enable additional troubleshooting options like command history, event history and original syslog prepend with Service diagnostic message.

### Usage:

```
tclsh <disk#:>/<dir_name>/sdiag_cfm.tcl <policy_name> <notification>  
<configuration history> <event history> <command history> <prepend  
trigger message> <user_pol_dir> <user_lib_dir> <policy_arguments>  
....
```

For policy detailed parameters: `tclsh <disk#:>/<dir_name>/sdiag_cfm.tcl <policy_name> -help`

# CFM helper scripts contd.

sdiag\_cfm\_display.tcl

Display deployed CFM policy and corresponding policy specific parameters.

Usage :

```
tclsh <disk#>:/<dir_name>/sdiag_cfm_display.tcl <policy_name>
```

# CFM helper scripts contd.

sdiag\_cfm\_undeploy.tcl

Undeploy a particular CFM policy and all its related parameters.

Usage:

```
tclsh <disk#>:./<dir_name>/sdiag_cfm_undeploy.tcl <policy_name>
```

# CFM helper scripts contd.

## collectEmailParameters.tcl

Configure email parameters to be used by Service Diagnostics policies for email notification.

Usage:

```
tclsh flash:poldir/collectEmailParameters.tcl <fromID> <toID>  
<ccID> <email_server_name>
```

# CFM helper scripts contd.

## cfmod.tcl

Run the On Demand policy along with optionally changing the domain name, service name, mep id and notification.

### Usage:

```
tclsh cfmod.tcl ma_name domain_name [-m mep_id] [-n email |  
syslog | snmp | all]
```

# CFM helper libraries

- email\_template\_cmd
- userlib.tcl
- diag\_lib.tcl
- lib.tcl
- lib\_2.tcl

# CFM MDF

## cfm.mdf

- XML based file run using Embedded Menu Manager(EMM).
- Provides an alternate Menu based user interface to deploy, display and undeploy CFM policies.
- Can be used as an alternative to helper scripts (note: helper scripts are required for cfm.mdf).
- cfm.mdf contains all the required scripts, policies and libraries and these can be extracted onto to the device.

- Usage:

**emm mdf** *mdf-url*

- Example:

Switch# **emm mdf** flash:poldir/cfm.mdf

# CFM Scenarios



# CFM EEM policies

## Scenario Name

On-Demand Diagnostics



Automatic Diagnostics



Auto-Traceroute



Shutdown Port



Unshut Port



## Policy Name

cfm\_ondemand.tcl

cfm\_cctimeout.tcl

cfm\_autotrace.tcl

cfm\_shut.tcl

cfm\_unshut.tcl

# CFM EEM libraries

- userlib\_2.tcl
- lib.tcl
- lib\_2.tcl
- tclIndex
- cfm\_lib.tcl



# Usage for CFM helper scripts

## sdiag\_cfm.tcl

Usage: tclsh <disk#:>/<dir\_name>/sdiag\_cfm.tcl **cfm\_ondemand.tcl**  
<notification> <configurationHistory> <EventHistory> <CommandHistory>  
<prepend trigger message> <user\_pol\_dir> <user\_lib\_dir>  
<domain\_name> <ma\_name> <optional mep-id>

```
me3400-1#tclsh flash:poldir/sdiag_cfm.tcl cfm_ondemand.tcl email true  
false true true flash:poldir flash:libdir operator-A service-100 63
```

Usage: tclsh <disk#:>/<dir\_name>/sdiag\_cfm.tcl **cfm\_cctimeout.tcl**  
<notification> <configurationHistory> <EventHistory> <CommandHistory>  
<prepend trigger message> <user\_pol\_dir> <user\_lib\_dir> <ma\_list>

```
me3400-1#tclsh flash:poldir/sdiag_cfm.tcl cfm_cctimeout.tcl email true  
false true true flash:poldir flash:libdir flash:inputfile
```

Usage: tclsh <disk#:>/<dir\_name>/sdiag\_cfm.tcl **cfm\_autotrace.tcl**  
<notification> <configuration history> <event history> <commandHistory>  
<prepend trigger message> <user\_pol\_dir> <user\_lib\_dir> <ma\_list>  
<timer>

```
me3400-1#tclsh flash:poldir/sdiag_cfm.tcl cfm_autotrace.tcl snmp true true  
true true flash:poldir flash:libdir flash:allinputfile 300
```

# Usage for CFM helper scripts

## sdiag\_cfm.tcl (cont.)

Usage: tclsh <disk#:>/<dir\_name>/sdiag\_cfm.tcl **cfm\_shut.tcl**  
<notification> <configuration history> <event history> <command history> <prepend trigger message> <user\_pol\_dir> <user\_lib\_dir>  
<ma\_list> <p2p>

```
me3400-1#tclsh flash:poldir/sdiag_cfm.tcl cfm_shut.tcl syslog true true  
true true flash:poldir flash:libdir flash:inputfile true
```

Usage: tclsh <disk#:>/<dir\_name>/sdiag\_cfm.tcl **cfm\_unshut.tcl**  
<notification> <configuration history> <event history> <command history> <prepend trigger message> <user\_pol\_dir> <user\_lib\_dir>

```
me3400-1#tclsh flash:poldir/sdiag_cfm.tcl cfm_unshut.tcl syslog true  
true true true flash:poldir flash:libdir
```

# Usage for CFM helper scripts

sdiag\_cfm\_display.tcl

```
tclsh <disk#>:/<dir_name>/sdiag_cfm_display.tcl [cfm_ondemand.tcl |  
cfm_cctimeout.tcl | cfm_shut.tcl | cfm_unshut.tcl | cfm_autotrace.tcl]
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_display.tcl cfm_ondemand.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_display.tcl cfm_cctimeout.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_display.tcl cfm_autotrace.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_display.tcl cfm_shut.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_display.tcl cfm_unshut.tcl
```

# Usage for CFM helper scripts

sdiag\_cfm\_undeploy.tcl

```
tclsh <disk#>:./<dir_name>/sdiag_cfm_undeploy.tcl [cfm_ondemand.tcl |  
cfm_cctimeout.tcl | cfm_shut.tcl | cfm_unshut.tcl | cfm_autotrace.tcl]
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_undeploy.tcl cfm_ondemand.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_undeploy.tcl cfm_cctimeout.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_undeploy.tcl cfm_autotrace.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_undeploy.tcl cfm_shut.tcl
```

```
me3400-1#tclsh flash:poldir/sdiag_cfm_undeploy.tcl cfm_unshut.tcl
```

# Usage for CFM helper scripts

## collectEmailParameters.tcl

```
tclsh flash:poldir/collectEmailParameters.tcl <fromID> <toID> <ccID>  
<email_server_name>
```

```
me3400-1#tclsh flash:poldir/collectEmailParameters.tcl  
user@cisco.com user@cisco.com user@cisco.com my_email_server
```

# Usage for CFM helper scripts

## cfmod.tcl

```
tclsh cfmod.tcl ma_name domain_name [-m mep_id] [-n email | syslog | snmp | all]
```

```
me3400-1#tclsh flash:poldir/ tclsh cfmod.tcl service-100 operator-A -m 61  
-n email
```

```
me3400-1#tclsh flash:poldir/ tclsh cfmod.tcl service-100 operator-A
```

```
me3400-1#tclsh flash:poldir/ tclsh cfmod.tcl service-100 operator-A -n  
snmp
```

```
me3400-1#tclsh flash:poldir/ tclsh cfmod.tcl service-100 operator-A -m 65
```

# Usage for CFM MDF

## Main Menu

```
=====
                          Connectivity Fault Management Diagnostics
                               Enter ? for help or ?# for item help
-----
1. Install Diagnostic Scripts
2. Set Global Variables (email parameters)
3. Deploy CFM Scenarios
4. Display Registered Policies
5. Display Environment Variables
6. Remove Diagnostic Policies
7. Exit

Enter selection [7]: █
```

# Usage for CFM MDF

## Deploy Menu

```
=====
                          Connectivity Fault Management Scenario Deployment
                                          Enter ?# for item help
-----
1. Deploy CFM CC-Timeout Script
2. Deploy CFM On-Demand Script
3. Deploy Action Shut Script
4. Deploy CFM Action Unshut Script
5. Deploy CFM Action Auto Trace Script
6. Return to main CFM menu

Enter selection [6]: █
```

# Usage for CFM MDF

## Display Menu

```
Please select diagnostic scenario to display from the running-configuration
```

1. CFM CC-Timeout
2. CFM On-Demand
3. CFM Action Shut
4. CFM Action Unshut
5. CFM Auto Trace

```
Enter choice: █
```

# Usage for CFM MDF

## Undeploy Menu

```
Please select diagnostic scenario to remove from the running-configuration
```

1. CFM CC-Timeout
2. CFM On-Demand
3. CFM Action Shut
4. CFM Action Unshut
5. CFM Auto Trace
6. ALL CFM Policies

```
Enter choice: █
```

# Types of notification

- Syslog notification
- SNMP trap notification
- Email notification
- Combination of any two of the above
- All notifications

# Syslog notification

## On-Demand scenario

```
me3400-1#  
Aug 17 11:45:28.341 PDT: %HA_EM-6-LOG: cfm_ondemand.tcl: TRIGGERED_BY "Ondemand script run manu  
ally": SERV-DIAG_CFM_ondemand: Diagnostics for vlan 660, CFM MD operator-A, rmep:all. CCDB: 0  
found, 0 with UP IF state; ErrorDB: 2 found; Reasons: RxAIS, Timeout; Ping: 1/1 total/fail;  
me3400-1#
```

# SNMP trap notification

## On-Demand scenario

```
ceemHistoryEventEntry.2.1 = 131
ceemHistoryEventEntry.3.1 = 0
ceemHistoryEventEntry.4.1 = 0
ceemHistoryEventEntry.5.1 = 0
ceemHistoryEventEntry.6.1 = flash:/poldir/cfm_ondemand.tcl
ceemHistoryEventEntry.7.1 = script: cfm_ondemand.tcl
ceemHistoryEventEntry.9.1 = 0
ceemHistoryEventEntry.10.1 = 0
ceemHistoryEventEntry.11.1 = SERV-DIAG_CFM_ondemand: Diagnostics for vlan 660, CFM MD operator
-A, rmep:all. CCDB: 0 found, 0 with UP IF state; ErrorDB: 2 found; Reasons: RxAIS, Timeout; Pi
ng: 1/1 total/fail;
```

# Email notification

## On-Demand scenario

```

From: user 1
To: user 1
Cc: user 1
Subject: From router me3400-1:
Sent: Mon 8/17/2009 4:06 PM

TRIGGERED_BY "Ondemand script run manually"::: SERV-DIAG_CFM_ondemand: Diagnostics for vlan 660, CFM MD Operator-A, rmep:all. CCDB: 0 found, 0 with UP IF state; ErrorDB: 2 found; Reasons: RxAIS, Timeout; Ping: 1/1 total/fail;

DETAILED MESSAGE:
Diagnostics for vlan 660, CFM MD Operator-A, rmep:all, level = 6 .
(1) No CCDB entry was found for the remote MEPs.
(2) Details of remote MEP collected from Error DB
(a) MEP ID - 63
(b) MAC Address - 0021.a1dc.6000
(c) Reason - "Lifetime Timer Expired"

(a) MEP ID - -
(b) MAC Address - 0021.a1dc.9280
(c) Reason - "Receive AIS"

(3) Output of loopback operation performed:
Ping to "0021.a1dc.6000" failed.
Number of successful pings: 0
Number of failed pings: 1

(4) Output of traceroute operation to remote MEP: Failed.
TRACEROUTE OUTPUT:
Type escape sequence to abort. TTL 64. Linktrace Timeout is 5 seconds Tracing the route to 0021.a1dc.6000 on Domain Operator-A, Level 6, vlan 660 Path to 0021.a1dc.6000 via FastEthernet0/3 is down. Traceroute terminated.
me3400-1#

(5) Previous successful trace route from cache is:
Traceroute to 0021.a1dc.6000 on Domain Operator-A, Level 6, vlan 660 issued at 5w5d path found via MPDB

B = Intermediary Bridge
! = Target Destination
* = Per hop Timeout

-----
Hops  Host          MAC          Ingress    Ingr Action  Relay Action
      Host          Forwarded    Egress     Egr Action   Previous Hop
-----
! 1          0021.a1dc.6000 Fa0/4       IngOk       RlyHit:MEP

THE CONFIGURATION HISTORY is: !
!Contextual Config Diff:
+no logging console
+event manager environment _email_from gm event manager environment
+_email_to gm event manager environment _email_cc gm event manager
+directory user policy "flash:/poldiremm"

```

# Deployment Considerations



# Pre-Requisites

- CFM (standards-compliant - d8.1) MD, MA and MEPs configuration
- CFM alarm syslogs enabled (“ethernet cfm logging”) for CFM EEM policies (cfm\_cctimeout.tcl, cfm\_shut.tcl, cfm\_unshut.tcl)
- CFM Traceroute cache enabled for CFM EEM policies (cfm\_autotrace.tcl, cfm\_ondemand.tcl)
- AIS alarm-suppression disabled
- SNMP community / manager configuration
- EEM SNMP traps enabled
- Domain name (“ip domain-name abc”) (for email notification)
- CFM MA input file for CFM EEM policies (cfm\_cctimeout.tcl, cfm\_autotrace.tcl, cfm\_shut.tcl)
- EMM feature (for cfm.mdf)

# Deployment considerations

- [cfm\_ondemand.tcl, cfm\_cctimeout.tcl, cfm\_autotrace.tcl] CFM EEM policy's execution time is directly proportional to the number of remote MEPs and corresponding operations (e.g. ping/traceroute) performed

Ethernet ping / traceroute timeout is 5 seconds

- [cfm\_cctimeout.tcl, cfm\_shut.tcl, cfm\_unshut.tcl] In case of concurrent syslogs triggering the same or different policies, the execution is always done in sequence
- [cfm\_cctimeout.tcl, cfm\_shut.tcl] Automatic Diagnostics and Shutdown Port policies have a common trigger (network failure event ("remote mepTimeout"))

This may cause both policies to trigger at the same time if input MA files have common entries

In this case, the order of policy deployment determines the execution order. Thus, it is recommended to always deploy first the Shutdown Port policy

## Deployment considerations contd.

- [cfm\_shut.tcl, cfm\_unshut.tcl] Unshut Port policy recovers interfaces previously brought down by the Shutdown Port policy. Thus, Shutdown Port policy must always be deployed for the Unshut Port policy to operate as designed
- [cfm\_shut.tcl, cfm\_unshut.tcl] The list of interfaces shutdown by the Shutdown Port policy is not persistent across reloads
- [cfm\_ondemand.tcl, cfm\_autotrace.tcl] On-demand Diagnostics policy parses out the content of CFM traceroute cache. Therefore, it is recommended to deploy and run the Autotrace policy in order to maintain the traceroute cache populated
- Syslog and SNMP notifications are faster compared to Email

# Deployment considerations contd.

- Potential enhancements:
- EEM Class-based Scheduling
  - Allows parallel execution of policies at the same class level
  - Allows prioritized execution of policies at different levels
  - Requires EEM 2.4
- EEM Turbo-TCL Policies
  - Provide an order-of-magnitude increase in event handling
  - Requires EEM 3.0

# Service Diagnostic CFM scenario Demonstration



# Service Diagnostic Demonstration

## Objectives

- **CLI-based** operation

  - CFM EEM policy deployment using CFM Helper script

  - CFM EEM policy display using CFM Helper script

  - CFM EEM policy execution (e.g. `cfm_ondemand.tcl`)

  - CFM EEM policy un-deployment using CFM Helper script

- **MDF-based** operation

  - CFM EEM policy deployment using CFM MDF

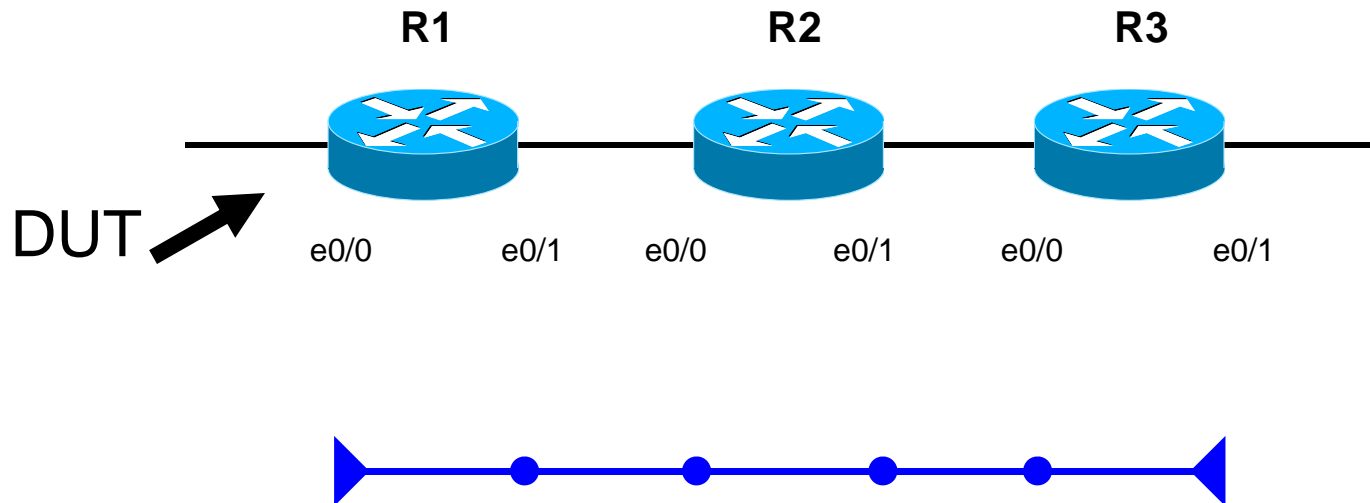
  - CFM EEM policy display using CFM MDF

  - CFM EEM policy execution (e.g. `cfm_cctimeout.tcl`)

  - CFM EEM policy un-deployment using CFM MDF

# Service Diagnostic Demonstration

## Topology



MIP ●  
MEP ►

# Troubleshooting



# Troubleshooting

## Show commands

- Show event manager policy registered - for checking the registered policies

```
No. Class Type Event Type Trap Time Registered Name
1 script user none Off Mon Jul 27 15:26:25 2009 cfm_ondemand.tcl
policyname {cfm_ondemand.tcl} sync {yes}
nice 0 queue-priority normal maxrun 600.000
```

- Show event manager directory user policy/library - for registered policy/library directory information

```
Router#show event manager directory user policy/library
flash:/poldir or - flash:/libdir
```

- Show event manager history events - for the chronology of policy triggers

```
No. Job Id Status Time of Event Event Type Name
1 1 success Mon Jul27 15:35:48 2009 none script: cfm_ondemand.tcl
2 2 success Mon Jul27 15:36:08 2009 none script: cfm_ondemand.tcl
```

# Troubleshooting

Show commands contd.

- Show event manager environment - for checking the configured environment variables

No.	Name	Value
1	_svcdiag_cfmOndemandNotif	SYSLOG
2	_svcdiag_cfmOndemandConfig	FALSE
3	_svcdiag_cfmOndemandEvent	FALSE
4	_svcdiag_cfmOndemandCommand	TRUE
5	_svcdiag_cfmOndemandPrepend	TRUE
6	_svcdiag_cfmOndemandMAname	MA-L6-1
7	_svcdiag_cfmOndemandDomain	Domain-L6-1
8	_svcdiag_cfmOndemandMpid	63

# Troubleshooting

## Show commands contd.

- **Show run | inc event** - for the entire set of event manager configurations

```
snmp-server enable traps event-manager
event manager environment _svcdiag_cfmOndemandNotif SYSLOG
event manager environment _svcdiag_cfmOndemandConfig FALSE
event manager environment _svcdiag_cfmOndemandEvent FALSE
event manager environment _svcdiag_cfmOndemandCommand TRUE
event manager environment _svcdiag_cfmOndemandPrepend TRUE
event manager environment _svcdiag_cfmOndemandMAname MA-L6-1
event manager environment _svcdiag_cfmOndemandDomain Domain-L6-1
event manager environment _svcdiag_cfmOndemandMpid 63
event manager directory user policy "flash:/poldir"
event manager directory user library "flash:/libdir"
event manager policy cfm_ondemand.tcl type user
```

# Troubleshooting

## Debug commands

- Debug event manager tclsh
- Debug event manager server
- Debug event manager policydir
- Debug event manager all

# Types of Problems

- Error in CFM EEM policy.
- Error in CFM Helper scripts.
- Error in MDF file.
- EEM policy not executed (or results not generated) for a given CFM trigger.
- Wrong EEM policy executed for a given CFM trigger.
- EEM policy executed unexpectedly.

# Error in CFM EEM policy

The reason for this could be:

- Tcl syntax error for the policy.
- Command/variable used in the policy is wrong or image/platform dependent output.
- The scenario is not covered by the policy.

# Error in CFM Helper scripts

The reason for this could be:

- Tcl syntax error for the helper script.
- Command/variable used in the script is wrong or image/platform dependent output.
- The underlying infrastructure required by Service Diagnostics is absent on the image.

## Error in MDF file

The reason for this could be:

- Tcl syntax error within MDF file.
- Command/variable used in the MDF file is wrong or image/platform dependent output.
- The underlying infrastructure required by Service Diagnostics is absent on the image.

# EEM policy not executed for a given CFM trigger

The reason for this could be:

- Command/variable used in the policy is wrong (image/platform) dependent output.
- The scenario is not covered by the policy.
- The underlying infrastructure required by Service Diagnostics is absent on the image.

# Wrong EEM policy executed for a given CFM trigger

The reason for this could be:

- Command/variable used in the policy is wrong (image/platform) dependent output.
- The scenario is not covered by the policy.

# EEM policy executed unexpectedly

The reason for this could be:

- Command/variable used in the policy is wrong (image/platform) dependent output.
- The scenario is not covered by the Service Diagnostics policy.
- Delays between the time when policy was triggered and when the policy was actually executed.

Q&A



Thank You!

# Additional Resources

- E-OAM site - <http://wwwin.cisco.com/ios/tech/ethernet/eoam/>
- EEM site - <http://wwwin.cisco.com/ios/tech/mgmt/eem/>
- Service Diagnostics site - <http://www.cisco.com/go/iossd>
- Aliases: cs-eem / cs-sdiag / cs-ether-oam

