Section 3: High Availability

Patrick Warichet CCIE#14218
Topics

1. Designing High Available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   – SSO/NSF
4. Cisco IOS Modularity
   – ION
5. In Service Software Upgrade
   – ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   – GOLD – EEM
7. Bidirectional Forwarding Detection
   – BFD
High Available Campus Design

Essential Components of Enabling Highly Available Networks in the Enterprise

HA in the Campus – Structured and Resilient Network Design
What Is Availability?

Availability is a function of MTBF and MTTR

Availability Equation

\[
\text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}
\]

MTBF = Mean Time Between Failure
MTTR = Mean Time to Repair
Where Does MTBF Come From?

MTBF figures come from:
- History
  - Your history
  - Vendor’s history
- Projections based on history
- Projections based on testing

You can get MTBF figures for Cisco products through your account representative
- Inside Cisco—MTBF query tool
  - User enters part number/product family and predicted MTBF is provided
  - A system is a chassis populated with Field Replaceable Units (FRU) and software

Availability = \( \frac{MTBF}{MTBF + MTTR} \)

Higher is Better
Mean Time to Repair

Much of what we do from a network design and feature perspective has to do with reducing MTTR.

MTTR can be reduced or eliminated by:

- Process improvement
- Automation
- Eliminating single points of failure
- Fast convergence
- Reducing the service impact of downtime (redundancy)

$$\text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

Lower MTTR Equals Higher Availability
Examples of Network Reliability Modeling Using Reliability Block Diagrams

Reliability = 99.938% with Four Hour MTTR (325 Minutes/Year)

Reliability = 99.961% with Four Hour MTTR (204 Minutes/Year)

Reliability = 99.9999% with Four Hour MTTR (30 Seconds/Year)
Predicting and Modelling Availability:

Often performed at design time
Availability can be increased by decreasing MTTR or increasing MTBF or both
If service availability target is 99.999% calculated availability must be better than 99.999%
Components in series reduce availability, parallel (redundant) components increase availability
Complex networks require modelling tools to calculate engineered availability
Networks designed for high availability will tolerate any single failure
Failure probabilities are also important, downtime often occurs because of multiple failures
Measuring Availability

Simple PINGs
Agents
Synthetic traffic generators
  - Cisco IOS® IP SLA
NetFlow
All of these provide useful data
Automatic data collection and reporting
Designing High Availability Campus

!Not This!

Server Farm

WAN  Internet  PSTN
Link Redundancy and Spanning Tree
Spanning Tree Should Behave the Way You Expect

The root bridge should stay where you put it
- Loopguard and rootguard

Only end station traffic should be seen on an edge port
- Port Fast
- BPDU guard
- Port-Security

There is a reasonable limit to B-Cast and M-Cast traffic volumes
- Configure storm control on backup links to aggressively rate limit B-Cast and M-Cast
- Utilize Sup720 rate limiters or SupIV/V with HW queuing structure
Optimizing Spanning Tree
Complex Topologies Take Longer to Converge

Time to converge is dependent on the protocol implemented 802.1d, 802.1s or 802.1w

It is also dependent on:

- Size and shape of the L2 topology (how deep is the tree)
- Number of VLAN’s being trunked across each link
- Number of logical ports in the VLAN on each switch

Non-congruent Topologies Take Longer to Converge. Restricting the topology is necessary to reduce convergence times. Clear unnecessary VLANs from trunk configuration.
Flex Link
Design Considerations

Spanning tree is not involved in link recovery however the network is ‘not’ L2 loop free

Access switch blocks BPDU’s on both the active and the backup Flex Link ports

Spanning tree should still be configured on access and distribution switches

Follow best practice spanning tree configuration on all ports not configured as Flex Links

Flex Link reduces size of the spanning tree topology but does not make the network loop free
Link Redundancy and Flex Link

Link Redundancy (Back-Up Link)

Flex Link provides a backup interface for an access switch uplink.

On failure of the prime link the backup link will start forwarding.

Link failure detection is processed locally on the switch.

```plaintext
interface gigabitethernet1/1
switchport trunk encapsulation dot1q
switchport mode trunk
switchport backup interface gigabitethernet1/2

interface gigabitethernet1/2
switchport trunk encapsulation dot1q
switchport mode trunk
```
Link Redundancy
Link and Neighbour Failure Detection

Indirect link failures are harder to detect

With no direct HW notification of link loss or topology change, convergence times are dependent on SW notification

In certain topologies, the need for TCN updates or dummy multicast flooding (uplink fast) is necessary for convergence

Indirect failure events in a bridged environment are detected by Spanning Tree Hello’s

You should not be using hubs in an high availability design
Redundancy and Protocol Interaction
Layer 2 and 3—Why Use Routed Interfaces

Configuring L3 routed interfaces provides for faster convergence than a L2 switchport with an associated L3 SVI.

1. Link Down
2. Interface Down
3. Routing Update

~ 8 msec loss

1. Link Down
2. Interface Down
3. Autostate
4. SVI Down
5. Routing Update

~ 150-200 msec loss
Link Redundancy
Trunking

Trunking multiple VLAN’s over a single physical link allows for more complex logical topologies without having to build complex physical topologies

802.1q
- IEEE standard
- Dynamic Trunking Protocol (DTP) - supports both 802.1q and ISL
- Default is to ‘not’ encapsulate the native VLAN

Inter-Switch Link (ISL)
- Cisco pre-standard version
- Dynamic Inter-Switch Link Protocol (DISL)
- All VLAN’s are tagged by default
Link Redundancy—Trunking
DTP Dynamic Trunk Protocol

Automatic formation of trunked switch to switch interconnection
- **On**: always be a trunk
- **Desirable**: ask if the other side can/will
- **Auto**: if the other sides asks I will
- **Off**: don’t become a trunk

Negotiation of 802.1Q or ISL encapsulation
- **ISL**: try to use ISL trunk encapsulation
- **802.1q**: try to use 802.1q encapsulation
- **Negotiate**: negotiate ISL or 802.1q encapsulation with peer
- **Non-Negotiate**: always use encapsulation that is hard-set
Trunk Design Considerations
Trunk—Auto/Desirable Takes Some Time

DTP negotiation takes time to complete
Hard coding the trunk will improve link-up recovery times
Operational manageability vs convergence trade-off

Two Seconds of Delay/Loss Tuned Away

IOS(config-if)# switchport mode trunk
IOS(config-if)# switchport nonegotiate
Link Redundancy
EtherChannel

EtherChannel creates a logical link by bundling multiple physical links.

It can provide for increased capacity as well as redundancy.

Failure of a link in a bundle will affect the spanning tree link cost and may result in a topology change.

Failure of a link in a bundle ‘may’ trigger a Layer 3 re-route.

Two to eight links in a bundle in the Cisco 6500
128 EtherChannels in a Cisco Catalyst 6500

In order to optimize the load balancing of traffic over multiple links deploy in powers of two (two, four, or eight)
EtherChannel Design Considerations

By default the Port-Channel interface associated with a physical EtherChannel bundle remains up as long as ‘one’ of the physical links is up.

When using LACP as the channel protocol it is possible to define how many links need to be active for the Port-Channel interface to remain up.

Balance the need to re-route vs. the need for the network capacity.

```
Sup720(config)# interface range gig 3/1 - 4
Sup720(config-if)# channel-protocol lacp
Sup720(config-if)# channel-group 5 mode on
Sup720(config)# interface port-channel 5
Sup720(config-if)# port-channel min-links 3
```
Link Redundancy—UDLD
Protecting Against One Way Communication

While 802.3z and 802.3ae link negotiation provides for L1 fault detection HW ASIC failures can still occur

UDLD provides an L2 based keep-alive mechanism that confirms bi-directional L2 connectivity

Each switch port configured for UDLD will send UDLD protocol packets (at L2) containing the port's own device/port ID, and the neighbor's device/port IDs seen by UDLD on that port

If the port does not see its own device/port ID echoed in the incoming UDLD packets the link is considered unidirectional and is shutdown
Default UDLD timers (15 second hellos) are intended to take down link prior to 802.1d spanning tree listening/learning transition.

Aggressive mode—after aging on a previously bi-directional link—tries eight times (once per second) to reestablish connection then err-disables port.

Aggressive mode protects against:

- One side of a link has a port stuck (both Tx and Rx)
- One side of a link remains up while the other side of the link has gone down

Example:

```
! Global configuration
udld aggressive
udld message time 7

! Interface
interface GigabitEthernet8/1
udld port aggressive
```
Layer 3 Link Redundancy
Layer 3 Indirect Neighbour Failure Detection

EIGRP, OSPF, IS-IS, mBGP all have native hello/dead mechanisms

Bidirectional Forwarding Detection (BFD)* provides a protocol independent mechanism
Layer 3 Link Redundancy
Equal Cost Multi-Path using Cisco Express Forwarding

In the recommended design the recovery from most component failures is based on L3 CEF ECMP recovery.

Behavior is deterministic independent of the number of links.

<table>
<thead>
<tr>
<th>Seconds of Lost Voice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>2.5</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>1.5</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

Path | Path & Switch

CEF Path Redundancy

CEF Path and Switch Redundancy
High Availability Campus Design
Redundancy and Protocol Interaction

Optimize the interaction of the physical redundancy with the network protocols

- Provide the necessary amount of redundancy
- Pick the right protocol for the requirement
- Optimize the tuning of the protocol

The network looks like this so that we can map the protocols onto the physical topology

We want to build networks that look like this

http://www.cisco.com/go/cvd
Topics

1. Designing High available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   – SSO/NSF
4. Cisco IOS Modularity
   – ION
5. In Service Software Upgrade
   – ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   – GOLD – EEM
7. Bidirectional Forwarding Detection
   – BFD
What is High Availability

Software Modularity
- In Service Software Upgrades (ISSU) for PSIRTs, Restartable Processes
- Fault Containment, Memory Protection
- Feature & Operational Consistency with IOS

Non-Stop Forwarding / Stateful Switch Over (NSF/SSO)
- Traffic continues flowing after a primary supervisor failure
- Sub-second recovery in L2 & L3 networks
- No line card reset

Physical Redundancy
- Redundant supervisors, power supplies, switch fabrics, and clocks

Generic Online Diagnostics (GOLD)
- Proactively detect and address potential hardware and software faults in the switch before they adversely impact network traffic
Catalyst 6500 – Physical Redundancy

- **Hot-Swap Fan Tray**
  - Includes Multiple Fans
  - Fans Maintain Proper Cooling Using Multiple Speeds and N+1 Redundancy

- **Chassis Backplane**
  - Provides Redundant Components

- **Redundant Supervisors**
  - Leverage Cisco IOS Software with Stateful Switchover and Non-Stop Forwarding

- **Redundant Power Supplies**
  - Provide Load Sharing 1:1 Power Redundancy

- **Hot-swap Interface Modules**

© 2007 Cisco Systems, Inc. All rights reserved.
Redundant Supervisors
Active – Standby Model

Active Supervisor
Control Plane
- Console access
- Manages Configurations
- Manages Chassis Environmentals
- L2 – L3 Protocols

Data Plane
- Hardware-based switching

Standby Supervisor
Not part of the active forwarding path

Multiple Redundancy modes
- COLD Standby
- HOT Standby

Synchronization
- CF – Checkpoint Facility
- RF – Redundancy Facility
Redundant Supervisors

Redundancy Modes

- **Stateful Switchover (SSO)**
  - Standby HOT
  - Stateful, forwarding tables & “HA aware” applications synchronize events and data structures between the two supervisors
  - IOS versions must be the same
  - 0-3 second outage after switchover

- **Route Processor Redundancy (RPR)**
  - Standby COLD
  - Startup configuration synchronized
  - IOS versions can be the same or different
  - 90+ second outage after switchover
Configuring Redundancy Mode

```
Cat6506E(config)#redundancy
Cat6506E(config-red)#mode ?
  rpr  Route Processor Redundancy
  sso  Stateful Switchover

Cat6506E(config-red)#mode sso
Cat6506E(config-red)#?
                   Redundancy configuration commands:
                   default              Set a command to its defaults
                   exit                 Exit from redundancy configuration mode
                   keepalive-enable     Enable/disable Redundancy keepalive
                   keepalive-threshold  Specify Redundancy keepalive threshold
                   keepalive-timer      Specify Redundancy keepalive timer (milliseconds)
                   main-cpu             Enter main-cpu mode
                   mode                 redundancy mode for this chassis
                   no                   Negate a command or set its defaults
                   notification-timer  Specify Redundancy notification timer (milliseconds)
                   policy               redundancy policy enforcement

Cat6506E(config-red)#
```
Verifying Redundancy Mode

Cat6506E#show redundancy
Redundant System Information :
----------------------------------
Available system uptime = 1 week, 3 days, 8 hours, 50 minutes
Switchovers system experienced = 0
  Standby failures = 2
  Last switchover reason = none

  Hardware Mode = Duplex
  Configured Redundancy Mode = rpr
  Operating Redundancy Mode = rpr
  Maintenance Mode = Disabled
  Communications = Up

Current Processor Information :
------------------------------
  Active Location = slot 6
  Current Software state = ACTIVE
  Uptime in current state = 1 week, 3 days, 8 hours, 49 minutes
  Image Version = Version 12.2(33)SXI, RELEASE SOFTWARE (fc2)
                  BOOT = sup-bootflash:s72033-ipbase-mz.122-33.SXI,1;
                  BOOTLDR = bootflash:
  Configuration register = 0x2102

Peer Processor Information :
----------------------------
  Standby Location = slot 5
  Current Software state = STANDBY COLD
  Uptime in current state = 10 minutes
  Image Version = Version 12.2(33)SXI, RELEASE SOFTWARE (fc2)
                  BOOT = sup-bootflash:s72033-ipbase-mz.122-33.SXI,1;
                  BOOTLDR = bootflash:
  Configuration register = 0x2102
Verifying Redundancy mode (cont.)

RPR

```
Cat6506E#show mod

<table>
<thead>
<tr>
<th>Mod</th>
<th>Ports</th>
<th>Card Type</th>
<th>Model</th>
<th>Serial No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>48</td>
<td>CEF720 48 port 10/100/1000mb Ethernet</td>
<td>WS-X6748-GE-TX</td>
<td>SAD073600VG</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>Supervisor Engine 720 (Cold)</td>
<td>WS-SUP720-BASE</td>
<td>SAD0752083T</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>Supervisor Engine 720 (Active)</td>
<td>WS-SUP720-BASE</td>
<td>SAD07230332</td>
</tr>
</tbody>
</table>
```

SSO

```
Cat6506E#show module

<table>
<thead>
<tr>
<th>Mod</th>
<th>Ports</th>
<th>Card Type</th>
<th>Model</th>
<th>Serial No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>48</td>
<td>CEF720 48 port 10/100/1000mb Ethernet</td>
<td>WS-X6748-GE-TX</td>
<td>SAD073600VG</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>Supervisor Engine 720 (Active)</td>
<td>WS-SUP720-BASE</td>
<td>SAD0752083T</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>Supervisor Engine 720 (Hot)</td>
<td>WS-SUP720-BASE</td>
<td>SAD07230332</td>
</tr>
</tbody>
</table>
```
SSO HA-Aware Redundancy Facility Clients

**Management & Services**
- EEM Server RF CLIENT
- SNMP HA RF Client
- Switch SPAN client
- MQC QoS
- Call-Home RF
- Port Security Client
- IKE RF Client
- IPSEC RF Client
- CRYPTO RSA
- IP Admission RF Client
- ISSU process
- LAN-Switch PAgP/LACP
- LAN-Switch Private V
- VLAN Mapping
- CTS HA

**L2 Services**
- Frame Relay
- HDLC
- LLDP
- PPP RF
- MPLS VPN HA Client
- LDP HA
- AToM manager
- Cat6k PAgP/LACP
- Spanning-Tree Protocol

**L3 Services**
- IPROUTING NSF RF ARP
- L3 Mobility Manager
- IP multicast RF Client
- Network RF Client
- HSRP
- GLBP
- BFD RF Client
- DHCP Snooping
- Cat6k MLS Multicast
- SLB RF Client

**Platform Specific**
- Cat6k Inline Power
- Car6k OIR
- Cat6k QoS Manager
- Cat6k Sip10g QoS Man
- Cat6k CWAN HA
- CWAN VLAN RF Client
- Cat6k Feature Manager
- Cat6k SPA TSM
- Cat6k Online Diag HA
- Cat6k Platform
- Config Sync RF client
- Cat6k Startup Config
- Cat6k Clear counter
- Cat6k HA Vpnsms

--- SNIP ---

**Partial List**

Cat6506E#show redundancy clients

- clientID = 0  clientSeq = 0  RF_INTERNAL_MSG
- clientID = 1319  clientSeq = 1  Cat6k Platform First
- clientID = 5030  clientSeq = 60  Redundancy Mode RF
Example, With Stateful Switchover

Supervisor switchover event occurs

SSO maintains SSO-aware applications, including L2 tables, L2 forwarding is maintained

Routing protocols will restart on the newly active Supervisor
  - L3 routes are purged

Routing neighbors lose adjacency with the restarting router
  - Routes to the lost neighbor are purged

Routing neighbors reestablish adjacencies, forwarding to and from non-directly connected L3 networks resumes
Non-Stop Forwarding

Non-Stop Forwarding, NSF, allows a router to continue forwarding data along routes that are already known, while the routing protocol information is being restored

- Supervisor switchover events (leverages SSO)
- Routing process restart in Catalyst 6500 with IOS Modularity

Implementations are unique to each routing protocol

- BGP, OSPF, EIGRP, ISIS, LDP

NSF router roles

- **NSF-aware router / NSF Helper** – A router running NSF-compatible software, capable of assisting a neighbor router perform an NSF restart
- **NSF-capable router** – A router configured to perform an NSF restart, therefore able to rebuild routing information from neighbor NSF-aware or NSF capable router
NSF/SSO Switchover Operation

1. Active Supervisor Fails

2. Database Synchronization

3. Data Plane

4. Control Plane

5. Control Path

6. ARP Table

7. Restart Notification

8. Hello. I Am NSF Aware

9. OSPF Process

10. EIGRP Process

11. IS-IS Process

12. BGP Process

Cisco IOS CEF Tables

Global Epoch = 1

FIB Table
- Prefix: 10.2
- Next Hop: 10.1.1.1
- Interface: Vlan 10
- Epoch: 0

- Prefix: 192.168.1.1
- Next Hop: 192.168.1.1
- Interface: Vlan 192
- Epoch: 0

Adjacency Table
- Next Hop: 10.1.1.1
- MAC: AA-BB-
- Epoch: 0

- Next Hop: 192.168.1.1
- MAC: EE-DD-
- Epoch: 0

Routing Information Base

© 2007 Cisco Systems, Inc. All rights reserved. Course acronym vx.x—#-38
Non-Stop Forwarding (cont)

Configuration

```
router eigrp 1
    nsf
!
router ospf 1
    nsf ietf
!
router isis 1
    nsf cisco
!
```

```
Cat6506E#show ip ospf nsf
Routing Process "ospf 1"
IETF Non-Stop Forwarding enabled
    restart-interval limit: 120 sec
IETF NSF helper support enabled
IETF NSF helper strict-lsa-checking enabled
Cisco NSF helper support enabled
    OSPF restart state is NO_RESTART
    Handle 2162698, Router ID 1.1.1.1, checkpoint Router ID 1.1.1.1
Config wait timer interval 10, timer not running
Dbase wait timer interval 120, timer not running
```

Configuration is required to enable “NSF Capable”
No configuration required to enable “NSF Helper” with default settings

- The NSF Helper feature is enabled by default with the routing protocol
- NSF Helper options do require configuration (helper options available with IETF mode)
Now, With NSF/SSO

Supervisor switchover event occurs

SSO maintains SSO-aware applications, including L2 tables, L2 forwarding is maintained

NSF-capable router signals NSF-aware routing peers of a routing protocol restart

NSF-aware routers detect the restarting router
  - Assist in re-establishing full adjacency
  - Maintain forwarding to and from the restarting router

NSF restart complete, traditional L3 convergence event is avoided
Non-Stop Forwarding with Stateful Switchover

NSF Awareness and Capability Support

- NSF is supported for EIGRP, OSPF, ISIS and BGP
- **Catalyst 6500**: Sup720 (all models) and Sup32 (all models) both support NSF capability and awareness
- **VSS supports** inter-chassis NSF/SSO
Topics

1. Designing High available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   - SSO/NSF
4. Cisco IOS Modularity
   - ION
5. In Service Software Upgrade
   - ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   - GOLD – EEM
7. Bidirectional Forwarding Detection
   - BFD
Cisco IOS Software
New Architecture with Software Modularity

- Routing
- IPFS
- TCP
- UDP
- CDP
- EEM
- INETD
- IOS-BASE

High Availability Infrastructure
Network Optimized Microkernel
Cisco Catalyst 6500 Hardware Data Plane
Cisco IOS Software Modularity Benefits
Minimize Unplanned Downtime

If an Error Occurs in a Modular Process
HA subsystem determines the best recovery action
- Restart a modular process
- Switchover to standby supervisor
- Remove the system from the network

Process restarts with no impact on the data plane
- Utilizes Nonstop Forwarding (NSF) even with a single Supervisor with NSF-Aware neighbors
- State checkpointing allows quick process recovery

Traffic Forwarding Continues During Unplanned Process Restarts
Cisco IOS Software Modularity Benefits

Simplify Software Changes

If the Software Needs to Be Patched:

The change can be made available as an individual patch which reduces code certification time.

Subsystem In-Service Software Upgrade (ISSU)* allows the change to be applied with no service disruption.

Traffic Forwarding Continues During Unplanned Process Restarts

*For All Processes Except Cisco IOS-BASE
Deploying Cisco IOS Software Modularity

Modes of Operation
Either Mode Has its Own Characteristics:

- **Recommended**
  - Installed Image
  - All benefits if run as a binary image
  - Allows patching of individual subsystems
  - Dynamically allocates memory for modules
  - File name contains “-vm”
  - File system with Cisco IOS needs to be available at all times
Deploying Cisco IOS Software Modularity
Install Process

Swmod#show version
Cisco Internetwork Operating System Software
IOS (tm) s72033_rp Software (s72033_rp-IPSERVICES_WAN-VM), Version 12.2(18)SXF4,
RELEASE SOFTWARE (fc1)

Patching is not available since the system is not running from an installed
image. To install please use the "install file" command

Swmod#install file disk0:s72033-ipservices_wan-vz.122-18.SXF4 disk0:/sys
Swmod(config)#no boot system flash disk0:/s72033-ipservices_wan-vz.122-18.SXF4
Swmod(config)#install bind disk0:/sys

Just Specify the Top
Level Directory

Swmod#copy running-config startup-config
Swmod#show running-config | include boot
boot system disk0:/sys/s72033/base/s72033-ipservices_wan-vm
Swmod#reload

Swmod#show version
Cisco Internetwork Operating System Software
IOS (tm) s72033_rp Software (s72033_rp-IPSERVICES_WAN-VM),

System restarted at 16:00:24 PDT Mon Apr 10 2006
System image file is "disk0:/sys/s72033/base/s72033-ipservices_wan-vm"

System is currently running from installed software
For further information use "show install running"
Cisco IOS Software Modularity
Tags and Rollback

Not only applying but also rolling back of patches is important

“Tags” are used for the rollback operation to define to what state the system should revert to

A “user tag” can be set to identify a specific point in the patching history

There are three pre-defined system tags:

- CISCO_BASE
- CISCO_LATEST
- CISCO_LATEST_ACTIVATE
Cisco IOS Software Modularity
Patching History—System Tags

install file...
install file...
install activate...

install rollback...
...CISCO_LATEST

install rollback...
...CISCO_LATEST

install rollback...
...CISCO_LATEST_ACTIVATE

install activate...

© 2007 Cisco Systems, Inc. All rights reserved.
Deploying IOS Software Modularity
New Functionality, with Investment Protection

What’s new?

- Enhancement to commands to take advantage of new functionality and architecture
- Images supporting software modularity end in “-vz” or “-vm”

What stays the same?

- Existing Cisco IOS functionality remains the same
- No new training required
- From a configuration perspective the same troubleshooting rules apply as for Cisco IOS so far
- Runs on existing Sup720 and/or Sup32 supervisors
- Software licensing, software release cycle, maintenance rebuilds and release numbering remain the same
Cisco IOS Software Modularity
Patch Navigator

Cisco.com offers a tool that allows downloading maintenance packs and patches: www.cisco.com/go/pn
Topics

1. Designing High available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   - SSO/NSF
4. Cisco IOS Modularity
   - ION
5. In Service Software Upgrade
   - ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   - GOLD – EEM
7. Bidirectional Forwarding Detection
   - BFD
What is EFSU

Enhanced Fast Software Upgrade is the initial phase of In Service Software Upgrade (ISSU) without Minimum Disruptive Restart (MDR) feature.

EFSU on cat6k is supported starting from 12.33(SXI).

EFSU is built on the Cisco ISSU framework.

EFSU uses the same ISSU CLI and processes.

EFSU minimizes traffic outage by pre-downloading Line Cards with the new image.

MDR will be introduced in the future releases and may require new Line Cards.
What is EFSU (cont.)

Provides ability to upgrade/downgrade IOS/ION images on a redundant switch

Extends the SSO/NSF concepts by removing the limitation of active and standby supervisors running the same image

Supports 18-month rolling window for image upgrade or downgrade

Applicable across maintenance releases and rebuilds of maintenance releases on the same software train
Cat6k software upgrade *pre* EFSU Release
Cat6k software upgrade *with* EFSU

1. SSO upgrade
2. Standby upgrade
3. SSO Switchover
4. (old) active upgrade

Performed with the ISSU Process
EFSU process

ISSU commands
- abortversion: abort ISSU process
- acceptversion: accept new IOS version on new Active
- commitversion: commit IOS version on Standby
- loadversion: load new IOS version on Standby
- runversion: run new IOS version on Standby and make it Active
Behavior on the linecards

Linecards that support ‘pre-download’ will be soft-reset at RunVersion

Pre-download reduces traffic outage time

Line cards that do not support pre-download during LoadVersion will be powercycled at RunVersion

Standby uplinks are stateful
Linecards that support pre-download

Pre-download is currently supported only on certain linecards

- 67xx
- SIP-200
- SIP-400
- FW-II only

Linecards must have at least 512 MB of memory to support pre-download.

Linecard must be powered-on and up (online) when the standby is booting after issuing a ‘loadversion’.
Verify linecards behavior

Verify that all linecard are up and running before issuing the “loadversion” (*show module*)

<table>
<thead>
<tr>
<th>Slot #</th>
<th>Card Type</th>
<th>MDR Mode</th>
<th>Max Outage Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>48 port 10/100 mb RJ-45 ethernet</td>
<td>RELOAD</td>
<td>360 secs</td>
</tr>
<tr>
<td>2</td>
<td>SFM-capable 16 port 1000mb GBIC</td>
<td>RELOAD</td>
<td>360 secs</td>
</tr>
<tr>
<td>3</td>
<td>48 port 10/100 mb RJ-45 ethernet</td>
<td>RELOAD</td>
<td>360 secs</td>
</tr>
<tr>
<td>4</td>
<td>CEF720 48 port 10/100/1000mb Ethernet</td>
<td>WARM_RELOAD</td>
<td>300 secs</td>
</tr>
<tr>
<td>8</td>
<td>CEF720 48 port 10/100/1000mb Ethernet</td>
<td>RELOAD</td>
<td>360 secs</td>
</tr>
<tr>
<td>9</td>
<td>Intrusion Detection System</td>
<td>RELOAD</td>
<td>360 secs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Slot #</th>
<th>Reason</th>
<th>Error Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>PLATFORM_INIT</td>
<td>3</td>
</tr>
</tbody>
</table>
Supported Software & Hardware

12.2SXI is the first release to support EFSU
EFSU works only if both images support EFSU
EFSU works only between same-type images

<table>
<thead>
<tr>
<th>New Running</th>
<th>adventerprisek9</th>
<th>adventerprisek9_wan</th>
<th>ipbasek9</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>adventerprisek9</td>
<td>Yes</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>adventerprisek9_wan</td>
<td>NO</td>
<td>Yes</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>ipbasek9</td>
<td>NO</td>
<td>NO</td>
<td>Yes</td>
<td>NO</td>
</tr>
<tr>
<td>...</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>Yes</td>
</tr>
</tbody>
</table>
# IOS Versus ION

<table>
<thead>
<tr>
<th></th>
<th>New</th>
<th></th>
<th></th>
<th>IOS</th>
<th>ION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running</td>
<td></td>
<td>IOS</td>
<td>ION</td>
<td>Yes</td>
<td>NO</td>
</tr>
<tr>
<td>IOS</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>ION</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
# ION Installed Versus ION Non Installed

<table>
<thead>
<tr>
<th></th>
<th>New</th>
<th>ION installed</th>
<th>ION non installed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running</td>
<td>ION installed</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>ION non installed</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

© 2007 Cisco Systems, Inc. All rights reserved.
Expected Traffic Outage

It all depends of the configuration and the type of hardware.

Some numbers for reference:

- Classic modules 50 seconds
- 65xx series cards 65 seconds
- 67xx series with pre-download CFC (512 MB) 80 seconds
- 67xx series no pre-download DFC (512 MB) 164 seconds
EFSU with IOS Modularity

Two types of patch

<table>
<thead>
<tr>
<th>Reload Patch</th>
<th>Restart Patch</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Patch that require a reload of the system to be activated.</td>
<td>• Patch that can be activated by just restarting one or several processes.</td>
</tr>
<tr>
<td>• Example: Patching ios-base or a non restartable process.</td>
<td>• Example: Patching iprouting process or any other restartable process.</td>
</tr>
</tbody>
</table>
EFSU With ION - Patching Steps

The steps are the same for a full installed image upgrade and a patch upgrade.

Install the patch/image on active and standby supervisor.

- The patch should be installed in the search root of the currently running installed image.

Use the ISSU CLI

- #issu loadversion disk0:/sys
- %issu loadversion executed successfully, Standby is being reloaded
Target of the patch

A running cat6k is running several types of software

- RP image
- SP image
- LC image (different depending on the linecard)

As of today, the following target are patchables:

- RP
- SP
- Lan LC 67XX and 65XX + DFC

WAN Line Card and any other Line Card are not patchables
The behavior of each patch types

<table>
<thead>
<tr>
<th>LC content</th>
<th>Reload Patch</th>
<th>Restart Patch</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC content</td>
<td>LC Reset</td>
<td>No LC Reset</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>No traffic impact</em></td>
</tr>
<tr>
<td>No LC content</td>
<td>In service Patching</td>
<td>No LC Reset</td>
</tr>
<tr>
<td></td>
<td><em>No traffic impact</em></td>
<td><em>No traffic impact</em></td>
</tr>
</tbody>
</table>
Topics

1. Designing High available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   – SSO/NSF
4. Cisco IOS Modularity
   – ION
5. In Service Software Upgrade
   – ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   – GOLD – EEM
7. Bidirectional Forwarding Detection
   – BFD
What Is GOLD?

GOLD defines a common framework for diagnostics operations across Cisco platforms running Cisco IOS Software.

Goal: check the health of hardware components and verify proper operation of the system data plane and control plane at run-time and boot-time.

- Bootup tests (includes online insertion)
- Health monitoring tests (background non-disruptive)
- On-Demand tests (disruptive and non-disruptive)
- User scheduled tests (disruptive and non-disruptive)
- CLI access to data via management interface
How Is GOLD Different From Other Forms Diagnostics?

GOLD performs functional tests typically using diagnostic packets switching through the system, as well ASIC memory testing

- Can be performed during runtime
- Typically use the same hardware path and IOS software drivers and user traffic

Power On Self Test (POST) occurs early on in the IOS initialization

- Focused on the CPU subsystem and memory components
Cat6K Online Diagnostic Methodology

Boot-up diagnostics touch every single ASIC/memory device in the data path and control path.

Perform Functional Testing combined with components monitoring to detect fault in passive components (connector, solder joint etc.) and active components (ASICs, PLDs etc.).

Tests are written using run-time driver routines to catch SW defects.

Non-disruptive tests are used as HA triggers.

Both disruptive and non-disruptive tests are available on-demand as trouble shooting tools for CA/TAC.

Root cause analysis and corrective actions are performed upon test failure.

EEM will be used for configurable corrective action. (Tcl based)
How Does GOLD Work?

Diagnostic packet switching tests verify that the system is operating correctly:

- Is the supervisor control plane and 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?

Other types of diagnostics tests including memory and error correlation tests are also available.
Generic Online Diagnostics
Diagnostic Operation

Boot-Up Diagnostics

Switch(config)#diagnostic bootup level complete

Runtime Diagnostics

Health-Monitoring

Switch(config)#diagnostic monitor module 5 test 2
Switch(config)#diagnostic monitor interval module 5 test 2 00:00:15

On-Demand

Switch#diagnostic start module 4 test 8
Module 4: Running test(s) 8 may disrupt normal system operation
Do you want to continue? [no]: y
Switch#diagnostic stop module 4

Scheduled

Switch(config)#diagnostic schedule module 4 test 1 port 3 on Jan 3 2005 23:32
Switch(config)#diagnostic schedule module 4 test 2 daily 14:45

Run During System Bootup, Line Card OIR or Supervisor Switchover
Makes Sure Faulty Hardware Is Taken out of Service

Non-Disruptive Tests Run in the Background
Serves as HA Trigger

All Diagnostics Tests Can Be Run on Demand, for Troubleshooting Purposes. It Can Also Be Used As A Pre-deployment Tool

Schedule Diagnostics Tests, for Verification and Troubleshooting Purposes

© 2007 Cisco Systems, Inc. All rights reserved.
Generic Online Diagnostics
View the GOLD Tests and Attributes

Switch#show diagnostic content mod 5
Module 5: Supervisor Engine 720 (Active)

<table>
<thead>
<tr>
<th>ID</th>
<th>Test Name</th>
<th>Attributes</th>
<th>Testing Interval (day hh:mm:ss.ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TestScratchRegister</td>
<td>M<em><strong>N</strong><strong>A</strong></em></td>
<td>000 00:00:30.00</td>
</tr>
<tr>
<td>2</td>
<td>TestSPRPInbandPing</td>
<td>M<em><strong>N</strong><strong>A</strong></em></td>
<td>000 00:00:15.00</td>
</tr>
<tr>
<td>3</td>
<td>TestTransceiverIntegrity</td>
<td>M**PD**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>4</td>
<td>TestActiveToStandbyLoopback</td>
<td>M**PD**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>5</td>
<td>TestLoopback</td>
<td>M**PD**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>6</td>
<td>TestNewIndexLearn</td>
<td>M**N**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>7</td>
<td>TestDontConditionalLearn</td>
<td>M**N**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>8</td>
<td>TestBadBpduTrap</td>
<td>M**D**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>9</td>
<td>TestMatchCapture</td>
<td>M**D**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>10</td>
<td>TestProtocolMatchChannel</td>
<td>M**D**<strong>I</strong>*</td>
<td>not configured</td>
</tr>
<tr>
<td>11</td>
<td>TestFibDevices</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>TestIPv4FibShortcut</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>TestL3Capture2</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>TestIPv6FibShortcut</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>TestMPLSFibShortcut</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>TestNATFibShortcut</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>TestAclPermit</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>TestAclDeny</td>
<td>M**N**<strong>I</strong>*</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>TestQoSTcam</td>
<td>M**D**<strong>I</strong>*</td>
<td></td>
</tr>
</tbody>
</table>

---SNIP---

Diagnostics test suite attributes:

- M/C/* - Minimal bootup level test / Complete bootup level test / NA
- B/* - Basic ondemand test / NA
- P/V/* - Per port test / Per device test / NA
- D/N/* - Disruptive test / Non-disruptive test / NA
- S/* - Only applicable to standby unit / NA
- X/* - Not a health monitoring test / NA
- F/* - Fixed monitoring interval test / NA
- E/* - Always enabled monitoring test / NA
- A/I - Monitoring is active / Monitoring is inactive
- R/* - Power-down line cards and need reset supervisor / NA
- K/* - Require resetting the line card after the test has completed / NA
- T/* - Shut down all ports and need reset supervisor / NA
An Example: Supervisor Data path Coverage

Monitors forwarding path between the switch processor, route processor and forwarding engine

Runs periodically every 15 seconds after system is online (configurable)

Ten consecutive failures is treated as fatal and will result in supervisor switchover or supervisor reset

Switch(config)#diagnostic monitor module 5 test 2
Switch(config)#diagnostic monitor interval module 5 test 2 00:00:15
Switch#show diagnostic result mod 7
Current bootup diagnostic level: complete
Module 7: CEF720 24 port 1000mb SFP

Overall Diagnostic Result for Module 7: MINOR ERROR
Diagnostic level at card bootup: complete

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
---------------------------------------------------------------

2) TestLoopback:

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
---------------------------------------------------------------
. . . . . . . . . . . . . . F . . . . . . . . . . . . . .

3) TestScratchRegister ------------->

4) TestSyncedFabChannel ------------->

---SNIP---
Reducing Downtime Thru Automation
GOLD Integration With EEM and Call Home

Automates problem diagnosis and information gathering
- EEM applets and scripts can initiate GOLD tests

Automates corrective actions and notifications
- GOLD events can trigger EEM scripts
- Beginning in release 12.2(33)SXH GOLD corrective actions are configured via EEM scripts

Automates result notification
- GOLD events are monitored by Call Home diagnostics profile group
Embedded Event Manager Supports Event Detector for GOLD

EEM can be used to track and perform corrective actions for GOLD. Beginning in release 12.2(33)SXH all GOLD corrective actions are scripted using EEM.

```
Cat6506E#show event manager policy available
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_fabsync.tcl
7  system Thu Feb 7  06:28:15 2036  Mandatory.go_intlpbk.tcl
8  system Thu Feb 7  06:28:15 2036  Mandatory.go_ltlmemconsty.tcl
9  system Thu Feb 7  06:28:15 2036  Mandatory.go_mac.tcl
10 system Thu Feb 7  06:28:15 2036  Mandatory.go_nondislp.tcl
---SNIP---
```
Call Home Service Monitors GOLD Status

- Automates the notification process
- Allows customization via profiles
  - Severity levels
  - Who gets notified
  - Which transport method
- Supported starting with IOS 12.2(33)SXH

```
call-home
  alert-group configuration
    alert-group diagnostic
    alert-group environment
    alert-group inventory
    alert-group syslog
    profile "CiscoTAC-1"
    no active
    no destination transport-method http
    destination transport-method email
    destination address email CallHome@cisco.com
    destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService
    subscribe-to-alert-group diagnostic severity minor
    subscribe-to-alert-group environment severity minor
    subscribe-to-alert-group syslog severity major pattern ".*"
    subscribe-to-alert-group configuration periodic monthly 8 16:34
    subscribe-to-alert-group inventory periodic monthly 8 16:19
```
Recommendations

Bootup diagnostics:
- Set level to complete

On demand diagnostics:
- Use as a pre-deployment tool: run complete diagnostics before putting hardware into production environment
- Use as a troubleshooting tool when suspecting hardware failure

Scheduled diagnostics:
- Schedule key diagnostics tests periodically
- Schedule all non-disruptive tests periodically

Health-monitoring diagnostics:
- Key tests running by default
- Enable additional non-disruptive tests for specific functionalities enabled in your network: IPv6, MPLS, NAT
Generic Online Diagnostics Summary

Provides a common framework to configure, view and schedule diagnostics across Cisco IOS based switches and routers.

GOLD functional tests verify both the data path and control path of the device, can be run during bootup and during runtime.

When combined with other features such as Embedded Event Manager and Call Home the MTTR, mean time to repair, can be dramatically lowered via process automation.
Topics

1. Designing High available Campus Network
2. Physical Redundancy
3. Stateful Switch Over - Non Stop Forwarding
   - SSO/NSF
4. Cisco IOS Modularity
   - ION
5. In Service Software Upgrade
   - ISSU/EFSU
6. Online Diagnostics – Embedded Event Manager
   - GOLD – EEM
7. Bidirectional Forwarding Detection
   - BFD
High Availability – Link Liveliness Detection

Multiple L3 protocols with individual hello timers
- BGP, OSPF, EIGRP, IS-IS, HSRP

Sub second hold timers/dead timers are not supported

Limited tuning options for different class devices
- The same timers are applied across all devices
- Timers applied in both directions
- Packets are processed by the control plane
Protocol Timer Tuning

OSPF
- Default 10s hello / 40s dead timer
- Minimum 1s hold time, multiplier value determines sub second hello interval

EIGRP
- Default 5s hello / 15s hold (high speed interfaces)
- Minimum 1s hello time / 3s hold time (recommended hold time 3x hello interval)

BGP
- Default 60s keepalive / 180s hold (recommended hold time 3x keepalive interval)
- Hold timer values below 20s are not recommended

IS-IS
- Default 10s hello / 30s hold timer
- Minimal 1s hold time, multiplier value determines sub second hello interval

HSRP
- Default 3s hello / 10s hold timer
- Millisecond timers are supported, minimum values are 15ms hello / 45ms hold
Bi-Directional Forwarding Detection

- Provides a lightweight hello protocol to detect faults in the path between two forwarding engines
  - Enhanced Control Plane scalability
  - Sub-second fault detection
- Operates independently from other protocols, therefore it can service multiple protocols at once.
  - OSPF, EIGRP, BFD, HSRP
- Flexible tuning options
  - Timers are negotiated per direction
BFD Modes

Asynchronous Mode
- Periodic Control packets are sent in each direction
- If packets are not received by the hold time expiration the link is declared down
- Asynchronous mode is supported in current Cisco Catalyst 6500 implementations

Demand Mode
- Once the session is established Control packets may cease, and can be sent again if the system needs to verify connectivity
- Demand mode is not supported in current Cisco Catalyst 6500 implementations

Echo mode
- Can be used in both Asynchronous mode or Demand Mode
- Echo packets are sent at a periodic rate with the destination IP address set to match the source, i.e. packets are looped back
- Verifies the full forwarding patch including the forwarding engine
- Echo packets are not seen by the neighbor device control plane, i.e. less round-trip jitter is induced, allows for faster rates
- Echo mode is supported beginning in the 12.2(33)SXH Cisco Catalyst 6500 release
BFD Configuration Example

```
BFD OSPF Global Configuration Example

bfd slow-timers 5000
!
interface GigabitEthernet2/1
ip address 192.168.32.0 255.255.255.254
bfd interval 500 min_rx 500 multiplier 3
!
interface GigabitEthernet2/2
ip address 192.168.96.0 255.255.255.254
bfd interval 500 min_rx 500 multiplier 3
!
router ospf 1
router-id 1.1.1.1
network 192.168.0.0 0.0.255.255 area 0
bfd all-interfaces
```

- Sets the interval for control packets in Echo Mode
- Multiplier value used to determine hold time
- Sets the Echo packet interval when in Echo mode
- Interval not used in Echo mode
- Applies BFD to all OSPF interfaces
BFD Session Monitoring

Catalyst6509E#show bfd neighbors detail

<table>
<thead>
<tr>
<th>NeighAddr</th>
<th>LD/RD</th>
<th>RH/RS</th>
<th>State</th>
<th>Int</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.32.1</td>
<td>4/1</td>
<td>Up</td>
<td>Up</td>
<td>Gi2/1</td>
</tr>
</tbody>
</table>

Session state is UP and using echo function with 50 ms interval.

OurAddr: 192.168.32.0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 5000000, MinRxInt: 5000000, Multiplier: 3
Received MinRxInt: 5000000, Received Multiplier: 3
Holddown (hits): 0(3), Hello (hits): 5000(923948)
Rx Count: 21711, Rx Interval (ms) min/max/avg: 1/5000/4387 last: 2948 ms ago
Tx Count: 21687, Tx Interval (ms) min/max/avg: 1/5000/4395 last: 2248 ms ago
Elapsed time watermarks: 4 4 (last: 4)
Registered protocols: OSPF
Uptime: 1w3d
Last packet: Version: 1
  State bit: Up
  Poll bit: 0
  Multiplier: 3
  My Discr.: 1
  Min tx interval: 5000000
  Min Echo interval: 500000
  - Diagnostic: 0
  - Demand bit: 0
  - Final bit: 0
  - Length: 24
  - Your Discr.: 4
  - Min rx interval: 5000000

Control packet interval derived from bfd slow-timer value
Echo packet interval derived from bfd minTX timer value
BFD is a protocol intended to detect faults in the bidirectional path between two forwarding engines, including physical interfaces, subinterfaces, data link(s), and to the extent possible the forwarding engines themselves, with potentially very low latency. It operates independently of media, data protocols, and routing protocols. An additional goal is to provide a single mechanism that can be used for liveness detection over any media, at any protocol layer, with a wide range of detection times and overhead, to avoid a proliferation of different methods.[1]

http://www.ietf.org/html.charters/bfd-charter.html
# Cisco Catalyst 6500 High Availability

## For Planned Upgrades

**Cisco ISSU Phase 1 (In-Service-Software-Upgrade)**
- Greatly reduces SW upgrade maintenance window (however, will still require linecard reload)
- VSS & ISSU will provide ~200msec upgrade window (assuming dual-homed access devices)

## For Planned/Unplanned Events

**Cisco IOS Software Modularity**
- Subsystem In-Service Software Upgrades
- Stateful Process Restarts
- Fault Containment, Memory Protection

**Generic On-Line Diagnostics (GOLD)/ Embedded Event Manager (EEM)**
Proactively detect and address potential hardware and software faults in the switch before they adversely impact network traffic

## For Unplanned Events

**Non-Stop Forwarding (NSF) and Stateful Switchover (SSO)**
- Traffic continues flowing after a primary supervisor failure
- Sub-second recovery in L2 and L3 networks
- No line card reset

**Physical Redundancy**
Redundant supervisors, power supplies, switch fabrics, and clocks
Catalyst 6500 BFD Frequently Asked Questions and Recommendations

Q. Can I enable BFD on SVIs (VLAN interfaces)?
A. No, BFD is only supported on routed interfaces.

Q. Is BFD a modularized process in IOS Software Modularity images?
A. No, BFD currently runs in the iosbase.process (the blob).

Q. Does BFD run in a distributed or centralized mode?
A. On the Catalyst 6500 BFD runs in a centralized fashion from the MSFC.

Q. Is BFD SSO-aware?
A. No, BFD is not SSO-aware. In the SXI release BFD implements an Admin Down State prior to a Supervisor switchover event thus allowing the routing protocols to leverage graceful restart technologies.

Q. What are the minimum timers recommended for the Catalyst 6500?
A. The Catalyst 6500 supports 50ms hello intervals with a multiplier value of 3, up to 100 sessions. DFC-only mode is recommended for systems with sub 1-second hold times (hello interval X multiplier value) otherwise Online Insertion and Removal events may cause the sub-second hold times to expire and break the session.