

TOMORROW starts here.



Cisco *live!*

Routed Fast Convergence and High Availability

BRKRST-3363

Dennis Leung

AS Technical Advisor

Abstract

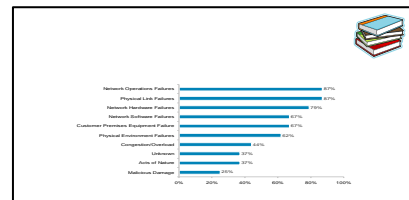
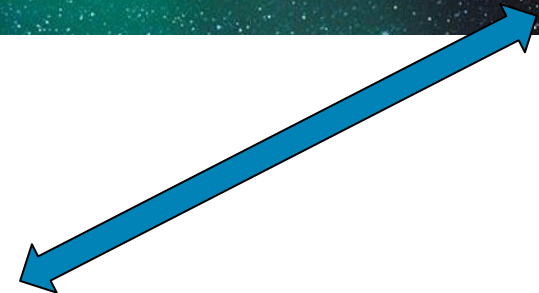
...This session discusses **various mechanisms** network engineers can use to **improve their network's convergence time and availability**, including nonstop forwarding, tuning for fast convergence.

Spoiler Alert!!

- Designing for fast convergence is more than tuning a few timers
- Think about *network* convergence, not just *routing protocol* convergence
- Look at layer 1 and layer 2 for failure detection properties and physical topology (shared-risk link groups)
- Failure detection is key to fast convergence
- OSPF – check your timers

Spoiler Alert!!

Slide Format Reference Slide



Agenda

- High Availability Overview
- IP Event Dampening
- Non Stop Forwarding and Graceful Restart
- Non Stop Routing
- Routed Convergence
- Summary



High Availability Overview

High Availability Overview Definitions

- Availability = $(\text{MTBF} - \text{MTTR}) / \text{MTBF}$
 - Useful definition for theoretical and practical
- MTBF is mean time between failure
 - What, when, why and how does it fail?
- MTTR is mean time to repair
 - How long does it take to fix?

What is High Availability?

Availability	DPM	Downtime Per Year (24x365)		
99.000%	10000	3 Days	15 Hours	36 Minutes
99.500%	5000	1 Day	19 Hours	48 Minutes
99.900%	1000		8 Hours	46 Minutes
99.950%	500		4 Hours	23 Minutes
99.990%	100			53 Minutes
99.999%	10			5 Minutes
99.9999%	1			30 Seconds



“High Availability”

The Culture of Availability

What's Your Availability Level?

Reactive?

Proactive?

Predictive?

Availability	DPM	Downtime Per Year (24x365)		
99.000%	10000	3 Days	15 Hours	36 Minutes
99.500%	5000	1 Day	19 Hours	48 Minutes
99.900%	1000		8 Hours	46 Minutes
99.950%	500		4 Hours	23 Minutes
99.990%	100			53 Minutes
99.999%	10			5 Minutes
99.9999%	1			30 Seconds

The Culture of Availability

What's Your Availability Level?



Reactive ~99%

- Few if any identified processes (except maybe to fix problems as reported by users)
- Significant number of Single Points of Failures
- Low tool utilisation
- Low level of consistency (HW, SW, config, design)
- No quality improvement processes

The Culture of Availability

What's Your Availability Level?



Proactive ~99.9%

- Good change management processes including what-if analysis and change validation
- Low number of Single Points of Failures
- Fault and configuration management tools
- Improved consistency (HW, SW, Config, design)
- Typically no quality improvement process

The Culture of Availability

What's Your Availability Level?



Predictive ~99.99+%

- Consistent processes for fault, configuration, performance and security
- No Single Points of Failures except at edge of network
- Fault, configuration, performance and workflow process tools
- Excellent consistency (HW, SW, config, design)

High Availability culture of quality improvement

Designing for Fast Convergence

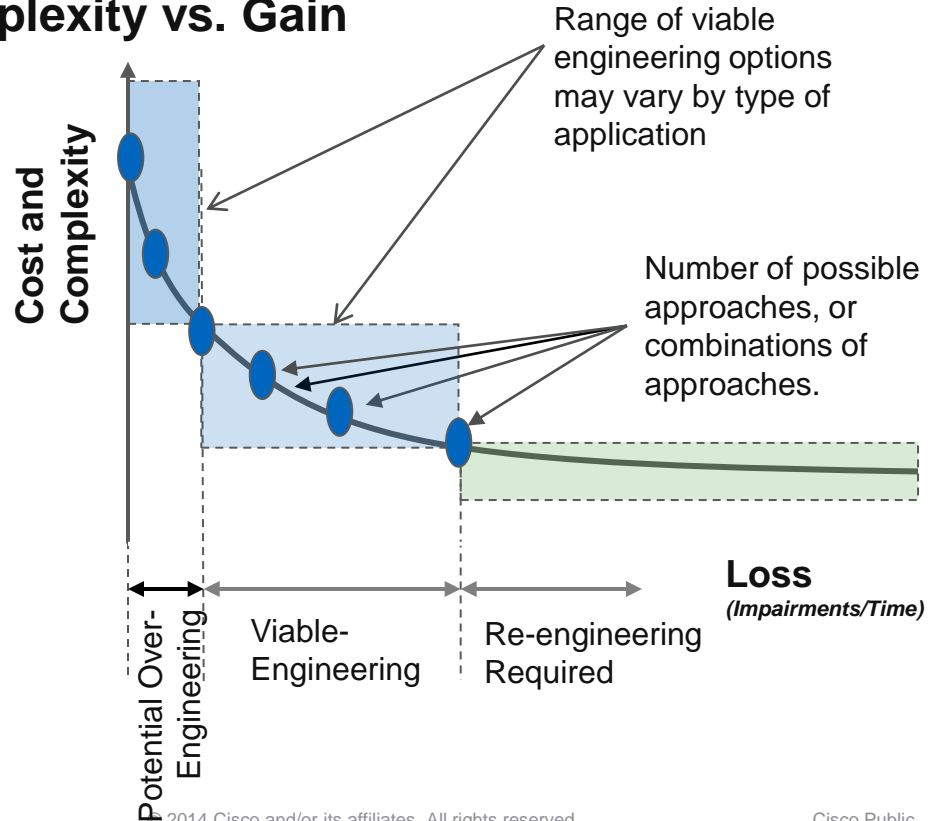
- Designing for FC is more than tuning a few timers
- Designers need to look at all network layers
 - Layer 1 and Layer 2 for failure detection properties and physical topology (shared-risk link groups)
 - Layer 3 protocol behaviour, interactions between different protocols
 - Layer 4-7 for application requirements and behaviour

Designing for Fast Convergence

- Only **3** numbers are interesting in the context of network convergence
 - 1) What is the *longest* the network could take to converge?
 - 2) What is the *average* amount of time the network could take to converge?
 - 3) *How long before applications running on the network lose their state?*
- If you know these three numbers you can tell...
 - 1) How well the network is going to perform
 - 2) Whether or not the network is meeting application requirements
 - 3) Whether or not to look for some way to make the network faster (or slower!)

Designing for Fast Convergence

Engineering Complexity vs. Gain



The Mistake in Fast Convergence

- Fast Convergence is doing the same things but faster
- Makes the routing protocol quicker to converge
- The mistake: we should not have been thinking about *routing protocol* convergence, but of *network* convergence
 - What can the network do to minimize loss in the event of a failure?
 - What can we do *outside* the routing protocol?



IP Event Dampening

IP Event Dampening

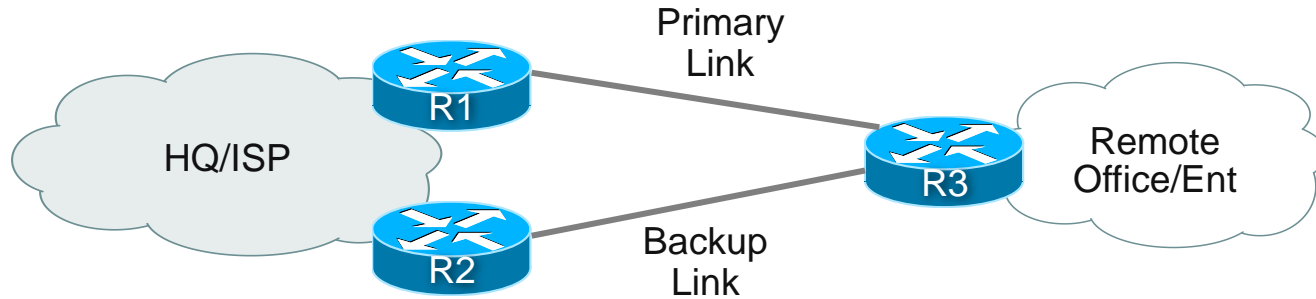
- Prevents routing protocol churn caused by constant interface state changes
- Supports all IP routing protocols
 - Static Routing, RIP, EIGRP, OSPF, IS-IS, BGP
 - In addition, it supports HSRP and CLNS routing
 - Applies on physical interfaces and can't be applied on sub-interfaces individually
- Available in IOS - 12.0(22)S, 12.2(13)T
- Also on IOS XR >= 2.0

IP Event Dampening Concept

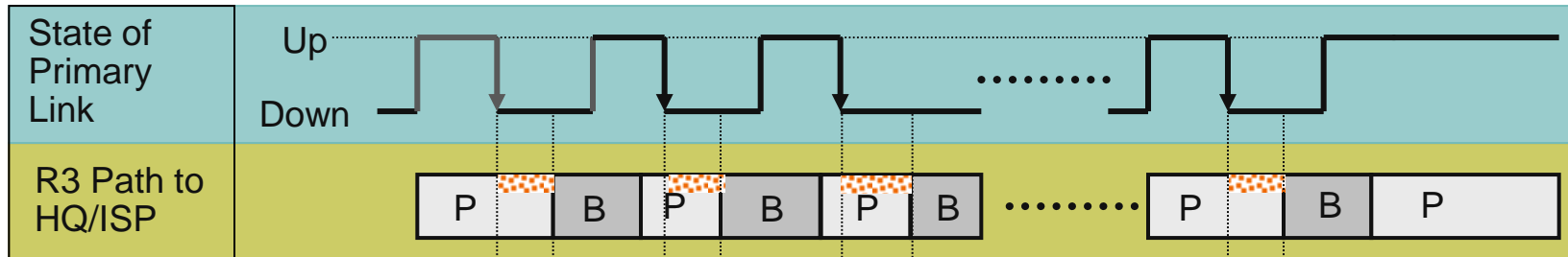
- Takes the concept of BGP route-flap dampening and applies it at the interface level, so all IP routing protocols can benefit
- Tracks interface flapping, applying a “*penalty*” to a flapping interface
- Puts the interface in “*down*” state from routing protocol perspective if the penalty is over a threshold tolerance
- Uses exponential decay algorithm to decrease the penalty over time and brings the interface back to “*up*” state

IP Event Dampening Deployment

Without IP Event Dampening



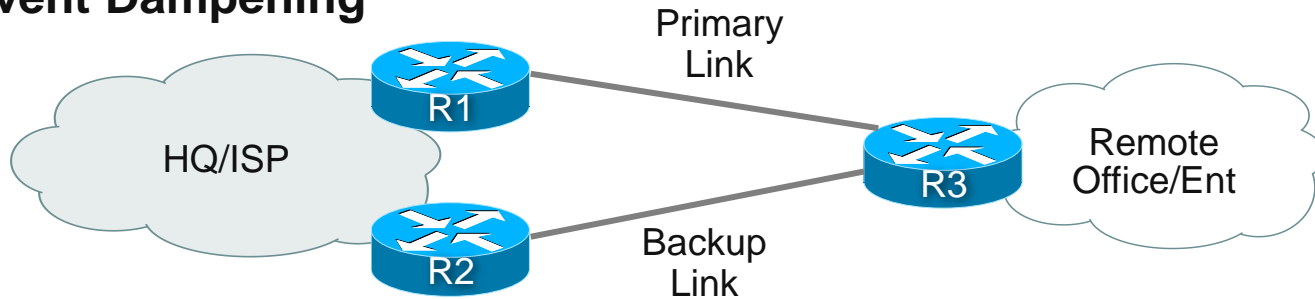
Link Flapping Causes Routing Reconvergence and Packet Loss



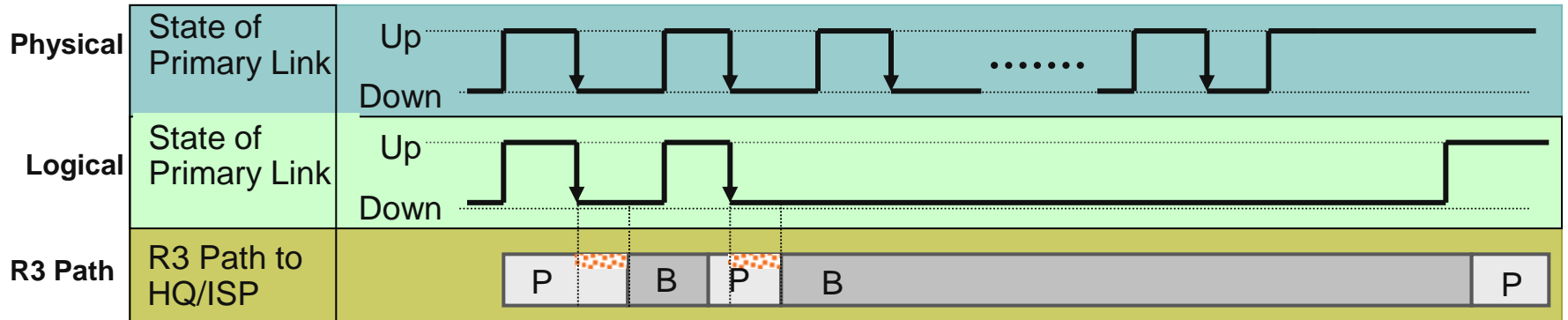
 Duration of Packet Loss

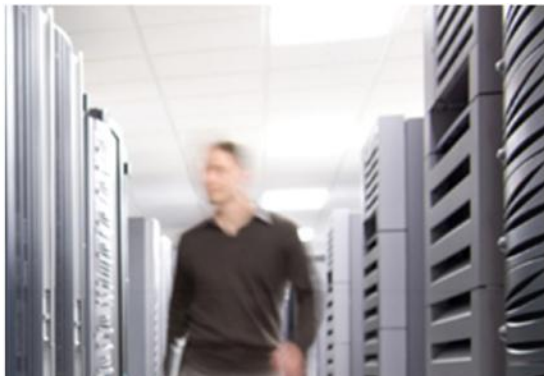
IP Event Dampening Deployment

With IP Event Dampening



IP Event Dampening Absorbs Link Flapping Effects on Routing Protocols

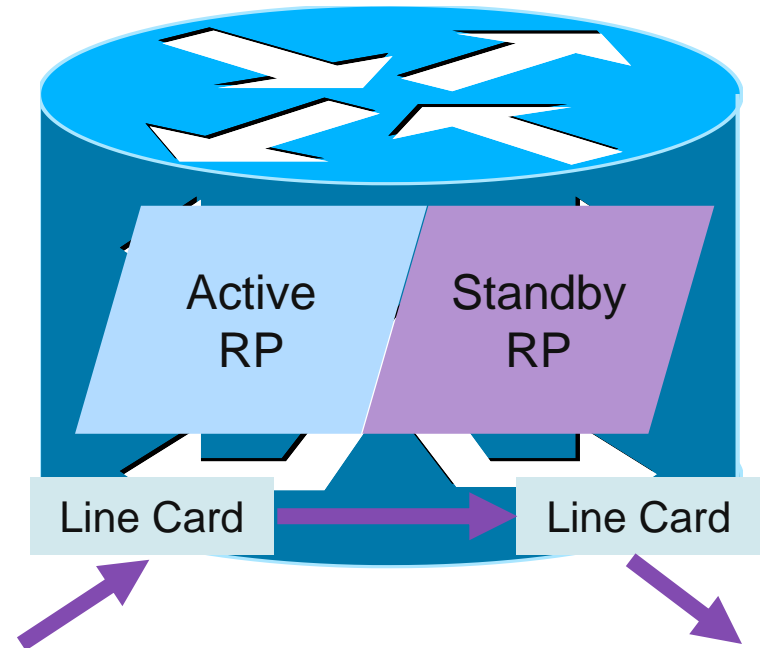




Non Stop Forwarding and Graceful Restart

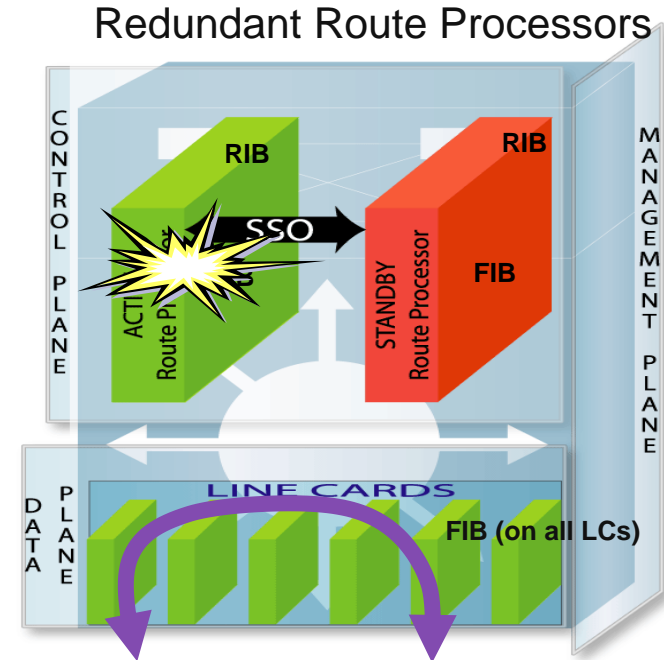
NSF/GR Frequently Used Terms

- **SSO** allows standby RP to take immediate control
- **NSF** continues to forward packets until route convergence is complete
- **GR** (graceful restart) reestablishes the routing information bases without churning the network



NSF/GR NSF/SSO

- Limit restart to be local event, not network wide
- Packet forwarding during switchover while routing is converging on Standby RP
- Non-stop forwarding of packets while control plane is reestablished and routing information is validated
 - Packet forwarding continues using current forwarding information base (FIB)
- Layer 3 (BGP, OSPF, IS-IS, EIGRP) recovers routing information from neighbours, rebuilds routing information base (RIB) and updates FIB



NSF/SSO Seeks to Preserve Traffic Forwarding

NSF/SSO Graceful Restart Relationship Building Process

NSF/SSO Capable



“I Can Preserve My Forwarding Table During Restart”

I Have Restarted

I Will Use Your Knowledge to Build My Database

Agreement

Restart Notification and Acknowledgement

Knowledge Transfer

Updates

NSF Aware Peer

During Restart

- 1) I Will Preserve Forwarding Table
- 2) I Will Not Declare you Dead
- 3) I Will Not Inform my Neighbours

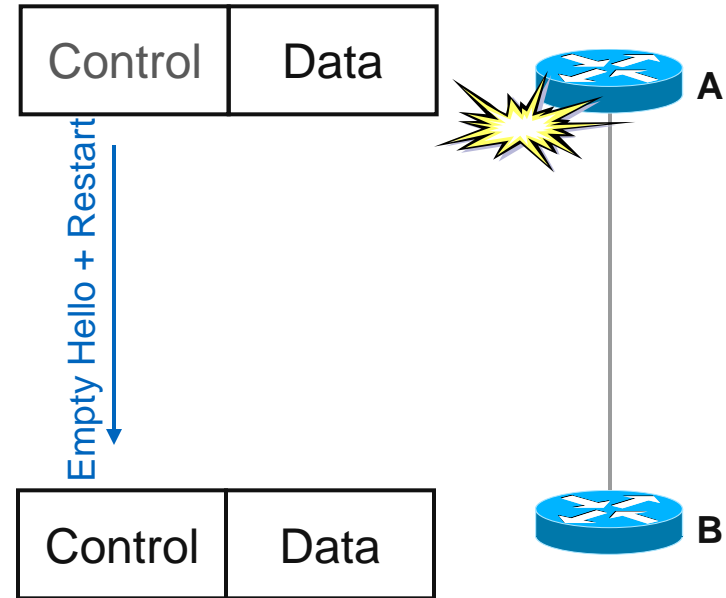
OK. I Acknowledge.
I Will Stick to My Agreement

This is my knowledge of the network

NSF/GR OSPF



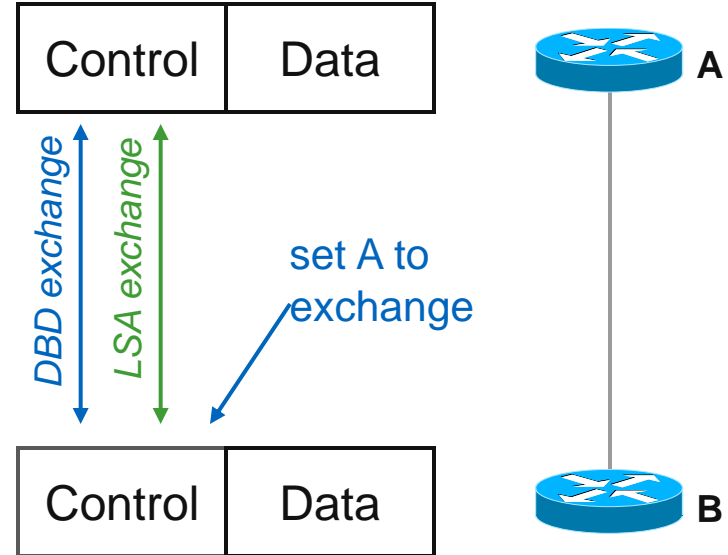
- OSPF uses an extension to the hello packets called link local signalling.
- The first hello **A** sends to **B** has an empty neighbour list; this tells **B** that something is wrong with the neighbour relationship.
- **A** sets the restart bit in its hello, which tells **B** that **A** is still forwarding traffic, and would like to resynchronise its database.





NSF/GR OSPF

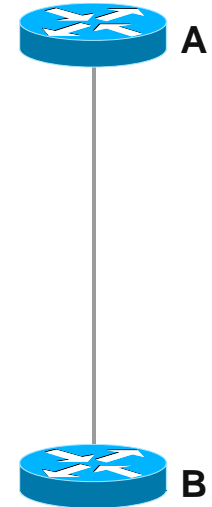
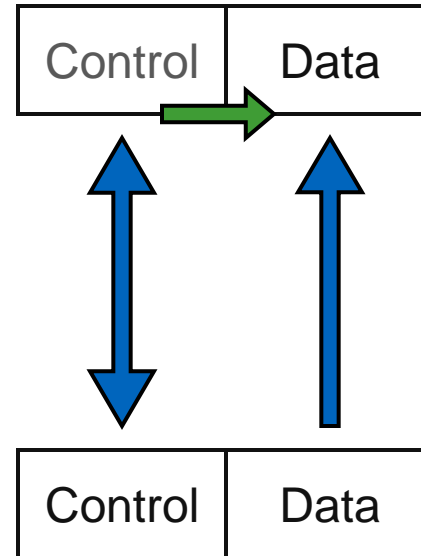
- **B** moves **A** into the exchange state, and uses out of band signalling (OOB) to resynchronise their databases.
- This process is the same as initial database synchronisation, but it uses different packet types.



NSF/GR OSPF



- When **A** and **B** have resynchronised their databases, they place each other in full state, and run SPF.
- After running SPF, the local routing table is updated, and OSPF notifies CEF.
- CEF then updates the forwarding tables, and removes all information marked as stale.



NSF/GR OSPF

- Use the **nsf** command under the **router ospf** configuration mode to enable graceful restart.
- **Show ip ospf** can be used to verify graceful restart is operational.

```
router ospf 100
nsf
....
```

```
router ospf 100
nsf
....
```

```
router#sh ip ospf
Routing Process "ospf 100" with ID 10.1.1.1
```

```
....
```

```
Non-Stop Forwarding enabled, last NSF restart 00:02:06 ago (took 44 secs)
```

```
router#show ip ospf neighbor detail
Neighbor 3.3.3.3, interface address 170.10.10.3
```

```
....
```

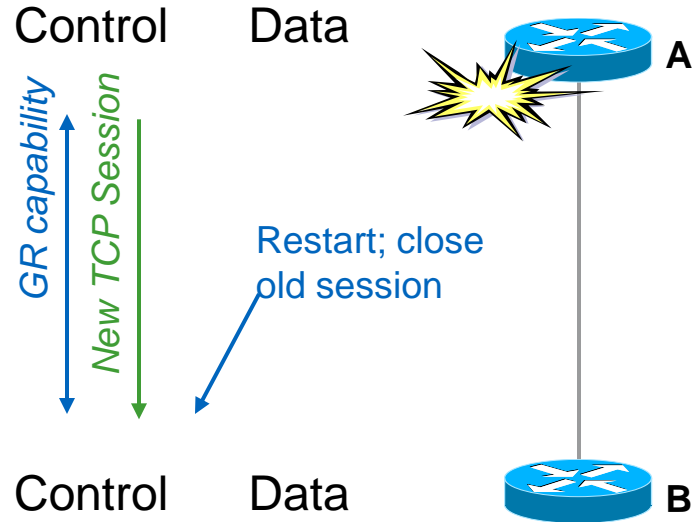
```
Options is 0x52
LLS Options is 0x1 (LR), last OOB-Resync 00:02:22 ago
```





NSF/GR BGP

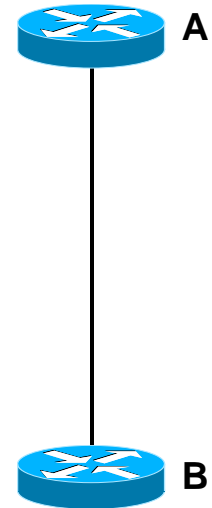
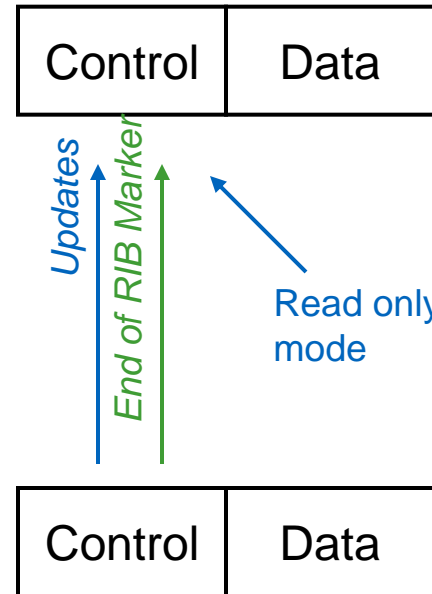
- When the BGP peering session is brought up, the graceful restart capability is negotiated. If both peers state they are capable of GR, it is enabled on the peering session.
- When **A** restarts, it opens a new TCP session to **B**, using the same router ID.
- **B** interprets this as a restart, and closes the old TCP session.



NSF/GR BGP



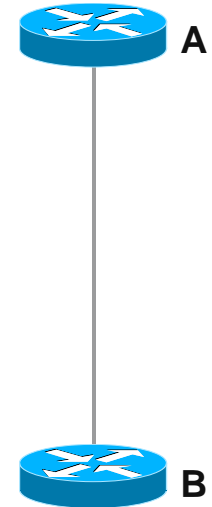
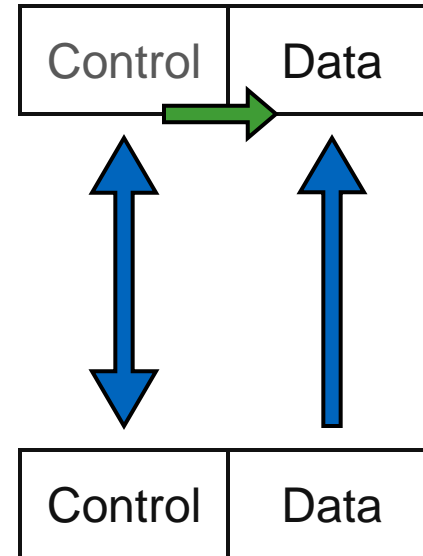
- **B** transmits updates containing its BGP table (it's local RIB out).
- **A** goes into read only mode, and does not run the bestpath calculations until its **B** has finished sending updates.
- When **B** has finished sending updates, it sends an end of RIB marker, which is an update with an empty withdrawn NLRI TLV.



NSF/GR BGP



- When A receives the end of RIB marker, it runs bestpath, and installs the best routes in the routing table.
- After the local routing table is updated, BGP notifies CEF.
- CEF then updates the forwarding tables, and removes all information marked as stale.



NSF/GR BGP

- Use the **bgp graceful-restart** command under the **router bgp** configuration mode to enable graceful restart.
- **Show ip bgp neighbours** can be used to verify graceful restart is operational.

```
router bgp 65000  
bgp graceful-restart  
....
```

```
router bgp 65501  
bgp graceful-restart  
....
```

```
router#show ip bgp neighbors x.x.x.x  
....  
Neighbor capabilities:  
....  
Graceful Restart Capability:advertised and received  
Remote Restart timer is 120 seconds  
Address families preserved by peer:  
IPv4 Unicast, IPv4 Multicast
```

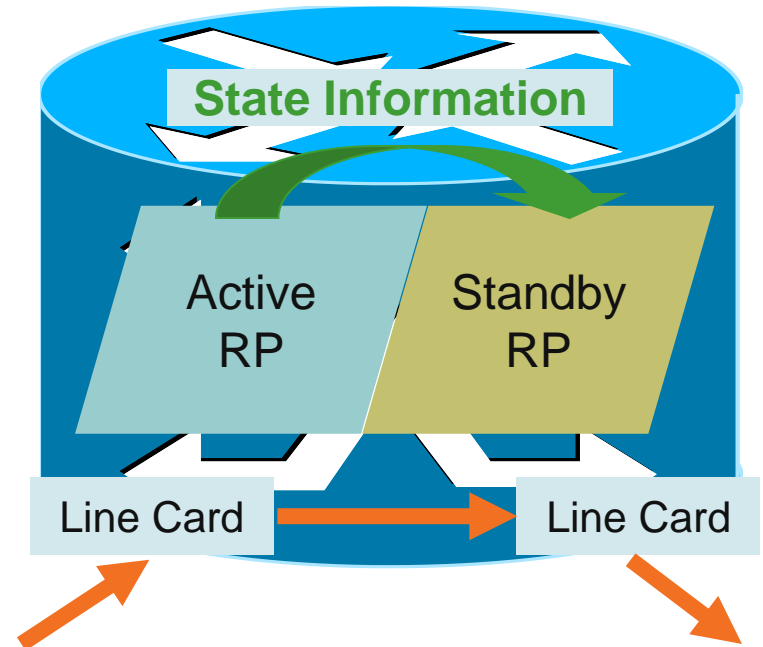




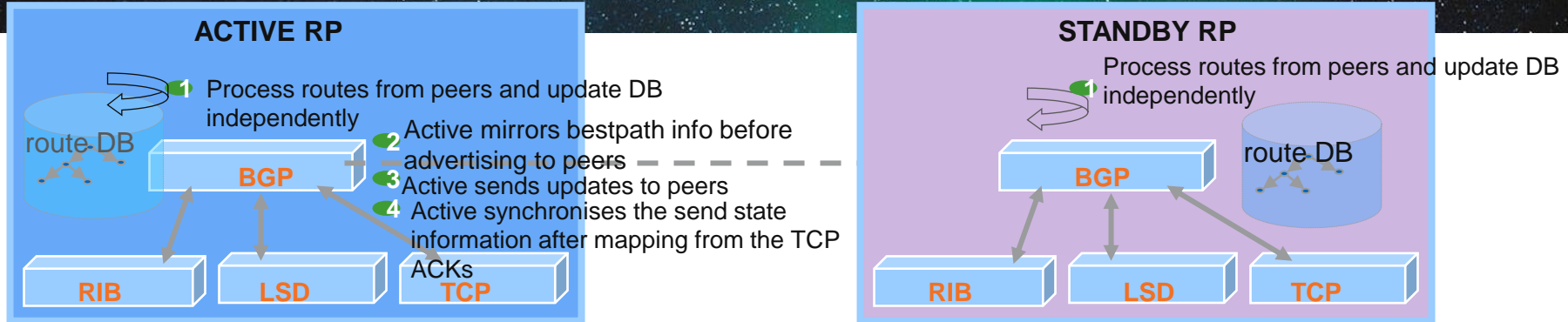
Non Stop Routing (NSR)

Non Stop Routing (NSR)

- **SSO** allows standby RP to take immediate control
- **NSF** continues to forward packets until route convergence is complete
- **GR** (graceful restart) reestablishes the routing information bases without churning the network
- **NSR** (Non-Stop Routing) Continue forwarding packets and maintain routing state



Non-Stop Routing (NSR) Operation



- Unlike GR, NSR is a self-contained solution to maintain the routing topology across HA events
- TCP connections and the routing protocol sessions are migrated from the active RP to standby RP without letting the peers knowing about the switchover
- Does not depend on any protocol extensions—relies on forwarding-plane's NSF capability
- Neighbours/protocol peers and rest of the network do not notice that an OSPF/LDP/BGP process went through a restart
 - Minimal LSA/Route information re-flooded during NSR recovery
 - Overall CPU usage greatly reduced during NSR recovery
 - Improves reliability of the overall system**

Non Stop Routing (NSR)



- IOS XE
 - OSPFv2, OSPFv3, ISIS, MPLS LDP, BGP
- IOS XR
 - OSPFv2, OSPFv3, ISIS, MPLS LDP, BGP



Routed Convergence

Agenda

- High Availability Overview
- IP Event Dampening
- Non Stop Forwarding and Graceful Restart
- Non Stop Routing
- **Routed Convergence**
 - Overview
 - Event Detection at Layer 3
 - BFD
 - Event Propagation - OSPF/ISIS
 - Event Processing – OSPF/ISIS
 - BGP
- Summary

Agenda

- Routed Convergence
 - Overview
 - Event Detection at Layer 3
 - BFD
 - Event Propagation - OSPF/ISIS
 - Event Processing – OSPF/ISIS
 - EIGRP (reference)
 - BGP



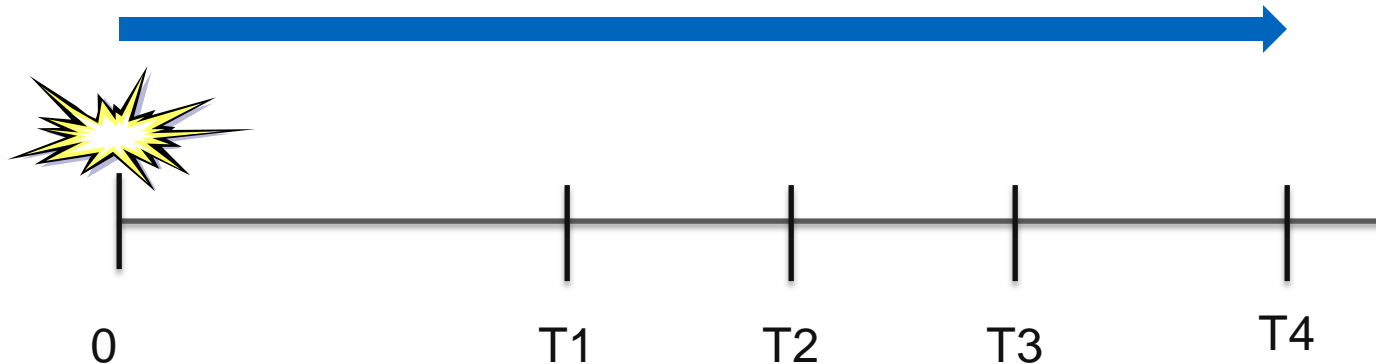
Network Convergence: Overview, L3 Event Detection, BFD

Network Convergence Overview

- Network convergence is the time needed for traffic to be rerouted to the alternative or more optimal path after the network event
- Network convergence requires all affected routers to process the event and update the appropriate data structures used for forwarding
- Network convergence is the time required to:
 - Detect the event
 - Propagate the event
 - Process the event
 - Update the routing table/FIB

Network Convergence Overview

- Network Convergence is the time required to:
 - Detect the event has occurred T1
 - Propagate the event T2
 - Process the event T3
 - Update related forwarding structures T4



*“The **detection** of network failures **consumes most of the convergence time budget** in typical designs”*

*“**Event driven detection** of link or neighbour failures is almost always going to be faster than **polled detection** of these failures.*

For instance, detecting the loss of carrier on a point-to-point Ethernet link is always faster than detecting the loss of three “hello” or “status” packets no matter how fast those hello packets are transmitted, received, and processed.”

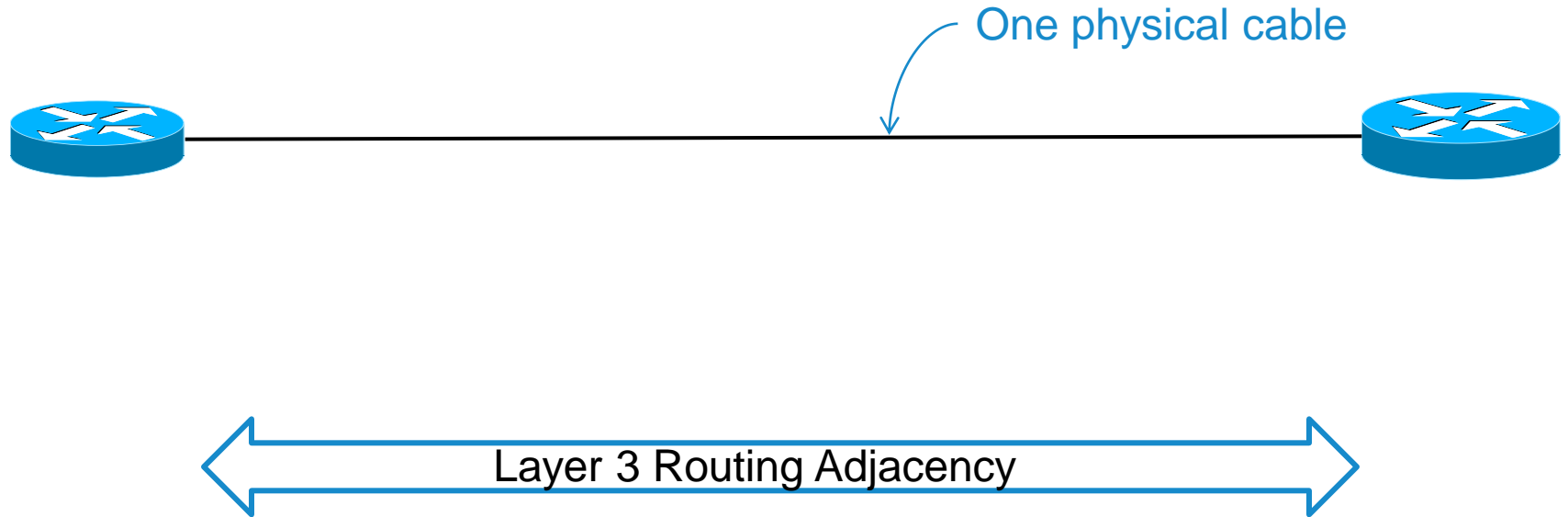
Russ White, Mosaddaq Turabi
CCDE Quick Reference

Event Detection at Layer 3



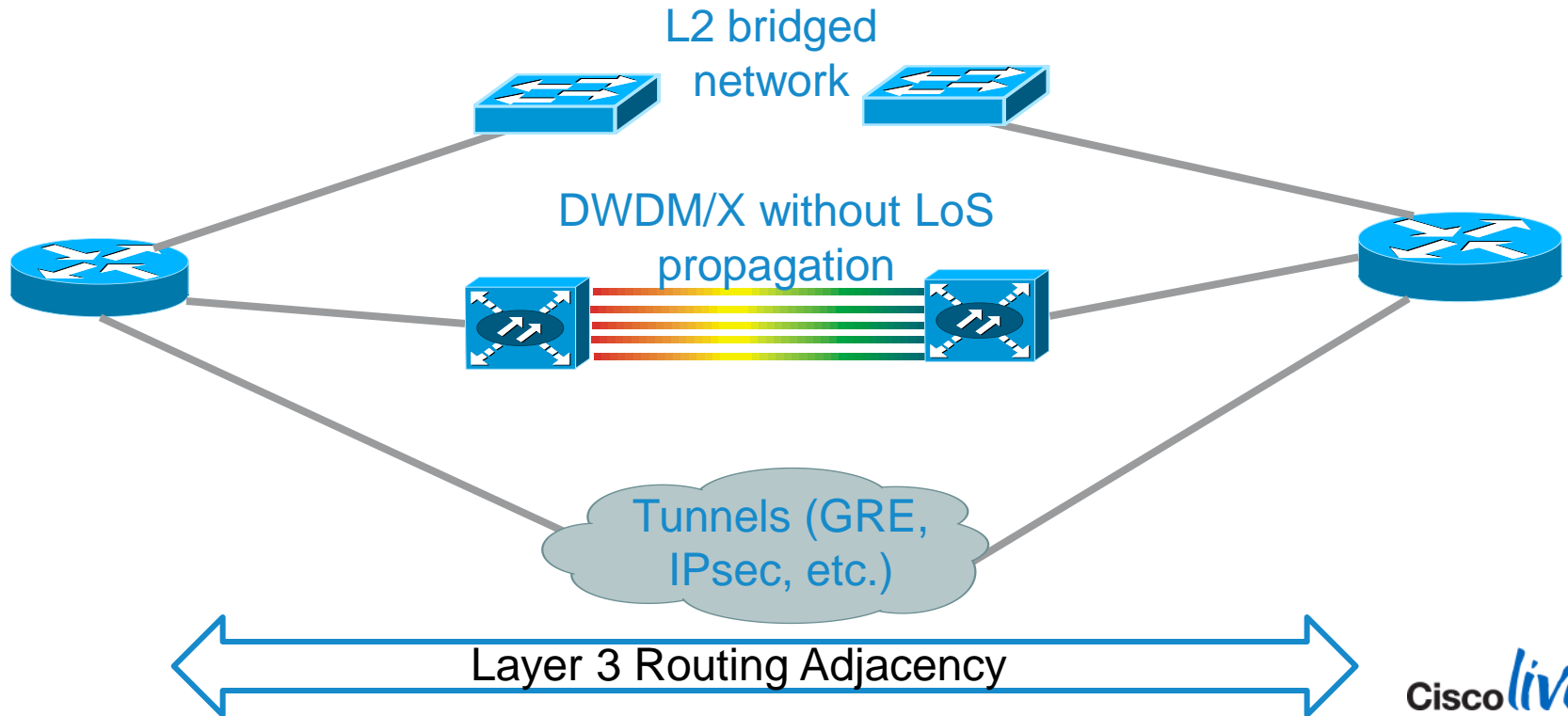
Event Detection at Layer 3

- In some environments, some or all failures can be directly tied to the physical



Event Detection at Layer 3

- In some environments, some or all failures require checks at Layer 3 (i.e. IP)



Event Detection at Layer 3

- All IGPs (EIGRP, OSPF and ISIS) use HELLOs to maintain adjacencies and to check neighbour reachability
 - Hello/Hold timers can be tuned down (“Fast Hellos”), however it is not recommend doing so because
 - This does not scale well to larger number of neighbours
 - Not a robust solution, high CPU load can cause false-positives
 - Having said this: Works reasonably well in small & controlled environments, for example Campus networks
- ➔ We need another solution: Use BFD!

BFD (Bi-directional Forwarding Detection)

- Lightweight Hello protocol
- Simple protocol, low overhead
- BFD is able to run distributed (ex: on “intelligent” linecards) and is able to scale to lower hello intervals and higher number of sessions
- Any “interested application” (OSPF, BGP, EIGRP, HSRP, TE FRR, etc.) registers with BFD and is notified as soon as BFD recognises a neighbour loss

BFD (Bi-directional Forwarding Detection)



- In normal scenarios if a failure occurred in the L2 bridged network, the layer 3 routers would have to rely on their IGP/BGP timers to detect the failures
- With BFD failure can be detected in less than a second

BFD (Bi-directional Forwarding Detection)



```
P4# router ospf 1
P4(config-router)# bfd all-interfaces
```

If you don't want to enable on all the interface you can use

```
P4(config-if)# ip ospf bfd <disable>
```

Following configuration may be needed on on an interface

```
[no] bfd interval <50-999> min_rx <1-999> multiplier <3-50>
```

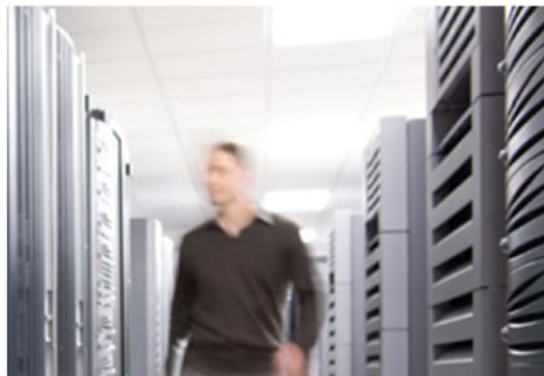


BFD (Bi-directional Forwarding Detection)

- Described in
 - RFC 5880, RFC 5881
 - RFC 5883 (Multihop)
- Similar in concept to HELLO or heartbeat-type protocol based on 3-way handshake—**BFD is Protocol Independent**
- Neighbours exchange unicast hello packets at negotiated regular intervals
- A neighbour is declared down when expected hello packets don't show up
- BFD payload packets are sent using encapsulation of each protocol/connection you want to monitor (IPv4, IPv6, 802.3, etc)
- BFD control packets encapsulated in UDP datagram (src port 3784)
- BFD multi-hop control packets encapsulated in UDP datagram (src port 4784)
- BFD echo packets – UDP datagram (dst port 3785)

Event Detection - Summary

- Failure Detection is **Key** to Fast Convergence!!
- BFD should be used wherever available
 - Higher interval can be considered when good Layer 1 failure detection is available (i.e. SONET/SDH), as a safety net



Network Convergence: Event Propagation/Event Processing OSPF

Agenda

- Network Convergence
 - Overview
 - Event Detection at Layer 3
 - BFD
 - Event Propagation – OSPF
 - Event Processing – OSPF
 - BGP

SPF and LSA Generation Throttling

Throttling is the general process of **slowing down** responses to the frequently oscillating events such as link flaps.

The general idea is to reduce resource wastage in unstable situations and wait till the situations calm down.....The general idea is as follows.

When an event occurs, e.g. a link goes down or new LSA arrives, **do not respond to it immediately**, e.g. by generating an LSA or running SPF, but wait some time, hoping to accumulate more similar events, e.g. waiting for the link to go back up, or more LSAs arriving.

This could potentially save a lot of resources, by reducing the number of SPF runs or amount of LSAs flooded.

The question is – how long should we hold or throttle the responses?

Petr Lapukhov

<http://blog.ine.com/2009/12/31/tuning-ospf-performance/>

Network Convergence Improvements

- OSPF Convergence Times

- Convergence =

Failure Detection + Event Propagation + SPF + FIB Update



Neighbour Down: Dead Time vs. BFD

Event Propagation: LSA generation throttle vs. LSA generation tuning

Event Processing: SPF throttle vs. SPF tuning

Event Propagation: OSPF

- Initial LSA Generation Delay
 - `OSPF_LSA_DELAY_INTERVAL`
 - Only Router and Network LSA Generation Delayed
- Recurring LSA Origination Delay
 - `MinLSInterval`
 - The minimum time between distinct originations of any particular LSA.
- Fast LSA Generation
- Repeated events increase regeneration delay
- Configuration:

```
timers throttle lsa all <lsa-start> <lsa-hold> <lsa-max>
timers lsa arrival <lsa-min-arrival>
```

Event Processing: OSPF

- SPF-DELAY and SPF-HOLDTIME protect the router as the cost of convergence time
- Configuration:
 - timers throttle spf <spf-start> <spf-hold> <spf-max>

OSPF: Default Timers



IOS

```
brisbane#sh ip ospf 100
Routing Process "ospf 100" with ID 30.0.0.13
Initial SPF schedule delay 5000 msec ← Initial SPF delay (5 secs)
Minimum hold time between two consecutive SPFs 10000 msec ← (10 secs)
Maximum wait time between two consecutive SPFs 10000 msec ← (10 secs)
Incremental-SPF disabled
Minimum LSA interval 5 secs. Minimum LSA arrival 1 sec ← Min LSA arrival
                                     ← Min LSA interval
```

OSPF: Default Timers



IOS XE

```
darwin#sh ip ospf 100
Routing Process "ospf 100" with ID 30.0.0.14
```

```
.
```

```
Initial SPF schedule delay 5000 msec ← Initial SPF delay (5 secs)
Minimum hold time between two consecutive SPFs 10000 msec ← (10 secs)
Maximum wait time between two consecutive SPFs 10000 msec ← (10 secs)
Incremental-SPF disabled
Minimum LSA interval 5 secs ← Minimum LSA arrival 1 sec ← Min LSA arrival
Min LSA interval
```

OSPF: Default Timers



IOS XR

```
RP/0/5/CPU0:rangers#sh ospf
```

Routing Process "ospf 100" with ID 30.0.0.4

Initial SPF schedule delay 50 msec	←	Initial SPF delay	(5/100ths sec)
Minimum hold time between two consecutive SPF	←		(2/10ths sec)
Maximum wait time between two consecutive SPF	←		(5 secs)
Initial LSA throttle delay 50 msec	←	Initial LSA delay	(5/100ths sec)
Minimum hold time for LSA throttle	←		(2/10ths sec)
Maximum wait time for LSA throttle	←		(5 secs)
Minimum LSA interval 200 msec. Minimum LSA arrival 100 msec	←	Min LSA arrival	
	←	Min LSA interval	

OSPF: Default Timers



N7K

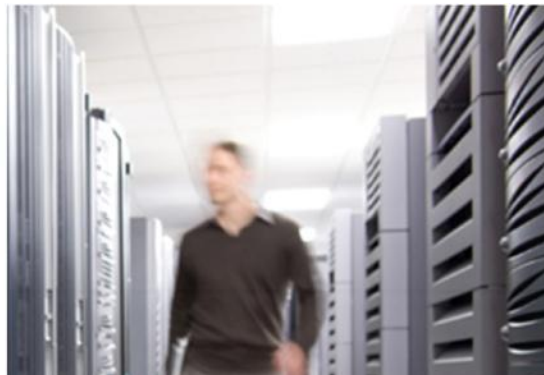
```
N7KC1# sh ip ospf
Routing Process 100 with ID 10.1.1.1 VRF default
```

```
...
...
```

```
Initial SPF schedule delay 200.000 msec, ← Initial SPF delay (2/10ths sec)
minimum inter SPF delay of 1000.000 msec, ← (1 sec)
maximum inter SPF delay of 5000.000 msec ← (5 secs)
Minimum hold time for Router LSA throttle 5000.000 ms ← (5 secs)
Minimum hold time for Network LSA throttle 5000.000 ms ← (5 secs)
Minimum LSA arrival 1000.000 msec ← Min LSA arrival
```

```
..
..
```

Initial LSA delay is (0 secs)



Network Convergence: BGP

BGP

- BGP and IGP Convergence tuning have a different focus
 - IGP Convergence - Rebuild the topology quickly following an event
 - BGP Convergence - Transfer large amounts of prefix information very quickly
- The magnitude of time involved is different
 - IGP - Sub-Second
 - BGP - Seconds to Minutes
- Fast IGP Convergence plays a role in maintaining availability for BGP prefixes
 - Often topological changes can result in no BGP changes, the IGP updates the next-hop information for BGP prefixes

BGP

Faster Convergence

- Typically two scenarios where we need faster convergence
- Single route convergence
 - A bestpath change occurs for one prefix
 - How quickly can BGP propagate the change throughout the network?
 - How quickly can the entire BGP network converge?
 - Key for VPNs and voice networks
- Router startup or “clear ip bgp *” convergence
 - Most stressful scenario for BGP
 - CPU may be busy for several minutes
 - Limiting factor in terms of scalability
 - Key for any router with a full Internet table and many peers

BGP

Initial Convergence

Initial convergence is limited by
the *amount of work* that needs to be done and
the router/network's ability to do this *fast and efficiently*

BGP

Initial Convergence

Initial convergence is limited by the *amount of work* that needs to be done and the router/network's ability to do this *fast and efficiently*

– ***The number of packets required to transfer the entire BGP database***

- The number of routes
- The number of peers
- The ability of BGP to pack routes into a small number of packets
- The number of peer specific policies

– ***TCP transport issues***

- How often does TCP go into slow start?
- How much can TCP put into one packet?

– ***Router Specific***

- Horsepower of your CPU, Code version, Outbound Interface Speed

BGP

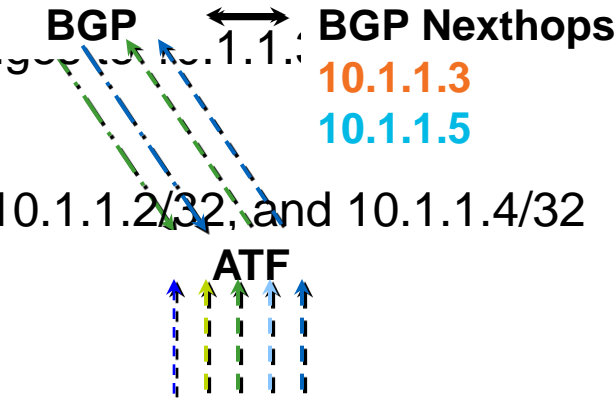
Converging the BGP Nexthops


- Every 60 seconds the BGP scanner recalculates bestpath for all prefixes
- Changes to the IGP cost of a BGP nexthop will go unnoticed until scanner's next run
 - IGP may converge in less than a second
 - BGP may not react for as long as 60 seconds ☹️
- Need to change from a polling model to an event driven model to improve convergence
 - Polling model – Check each BGP next hop's IGP cost every 60 seconds
 - Event driven model – BGP is informed by a 3rd party when the IGP cost to a BGP nexthop changes

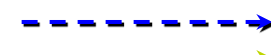
BGP Next Hop Tracking
Enabled by default

BGP

Address Tracking Filter

- BGP tells ATF to let us know about any changes to BGP Nexthops


The diagram shows a BGP router connected to BGP Nexthops. The nexthops are 10.1.1.1 (grey), 10.1.1.3 (orange), and 10.1.1.5 (blue). A double-headed arrow connects the BGP router to the nexthops. Dashed arrows point from the nexthops to the ATF. The ATF is shown with five arrows pointing up to the RIB: blue (10.1.1.1/32), yellow (10.1.1.2/32), green (10.1.1.3/32), grey (10.1.1.4/32), and blue (10.1.1.5/32). The ATF is also shown with three arrows pointing right to the RIB: blue (10.1.1.1/32), yellow (10.1.1.2/32), and green (10.1.1.3/32). The RIB is shown with five entries: 10.1.1.1/32 (grey), 10.1.1.2/32 (yellow), 10.1.1.3/32 (orange), 10.1.1.4/32 (grey), and 10.1.1.5/32 (blue). The GP is shown with a grey arrow pointing to the RIB.
- ATF filters out any changes for 10.1.1.1/32, 10.1.1.2/32, and 10.1.1.4/32


The diagram shows three dashed arrows pointing right: blue (10.1.1.1/32), yellow (10.1.1.2/32), and green (10.1.1.4/32).
- Changes to 10.1.1.3/32 and 10.1.1.5/32 are passed to the RIB


The diagram shows two dashed arrows pointing right: orange (10.1.1.3/32) and blue (10.1.1.5/32).

BGP

Next Hop Tracking

- BGP registers all nexthops with ATF
- ATF will let BGP know when a route change occurs for a nexthop
- ATF notification will trigger a lightweight “BGP Scanner” run
 - Bestpaths will be calculated
 - None of the other “Full Scan” work will happen

BGP

Minimum Route Advertisement Interval (MRAI)

“...determines the minimum amount of time that must elapse between an advertisement and/or withdrawal of routes to a particular destination by a BGP speaker to a peer. This rate limiting procedure applies on a per-destination basis, although the value of MinRouteAdvertisementIntervalTimer is set on a per BGP peer basis.”

RFC 4271

Section 9.2.1.1

BGP

Minimum Route Advertisement Interval (MRAI)

- MRAI timers are maintained per peer
 - iBGP – 0 seconds by default
 - eBGP – 30 seconds by default
 - neighbor x.x.x.x advertisement-interval <0-600>
- Pros
 - Promotes stability by batching route changes
 - Improves update packing in some situations
- Cons
 - May **drastically** slow convergence
 - One flapping prefix can slow convergence for other prefixes

BGP

Minimum Route Advertisement Interval (MRAI)

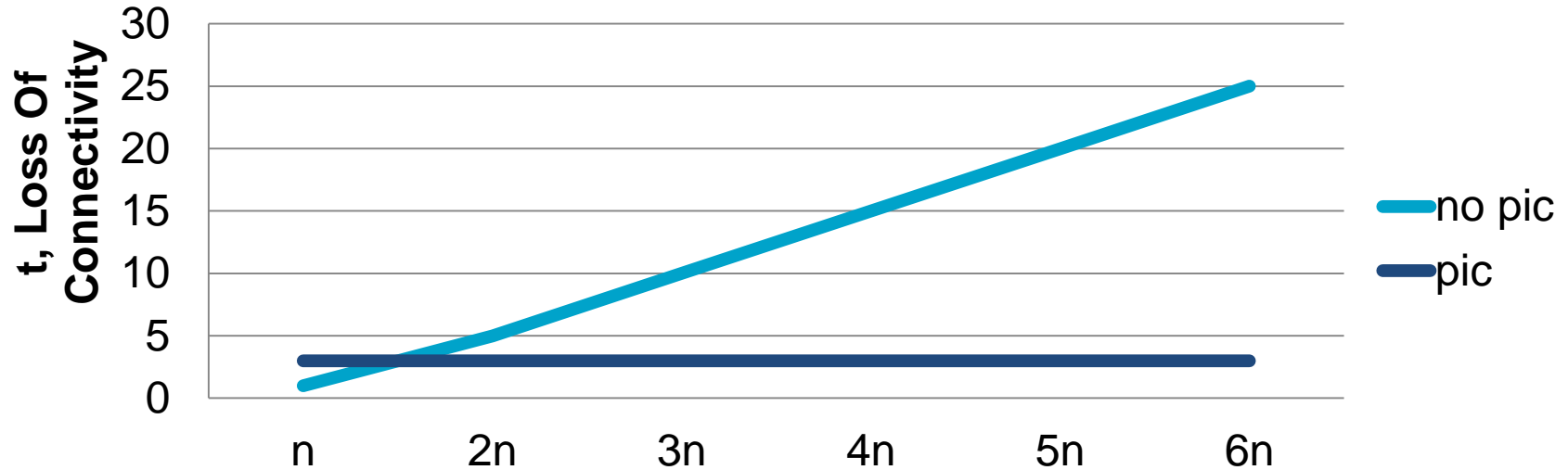
- BGP is not a link state protocol, but instead is path vector based
- May take several “rounds/cycles” of exchanging updates & withdraws for the network to converge
- MRAI must expire between each round!
- The more fully meshed the network and the more tiers of Autonomous Systems, the more rounds required for convergence
- Think about
 - The many tiers of Autonomous Systems that are in the Internet
 - The degree to which peering can be fully meshed



A Couple of Additional Options to Explore:
BGP PIC, BGP AddPath, LFAs

BGP Prefix Independent Convergence (PIC)

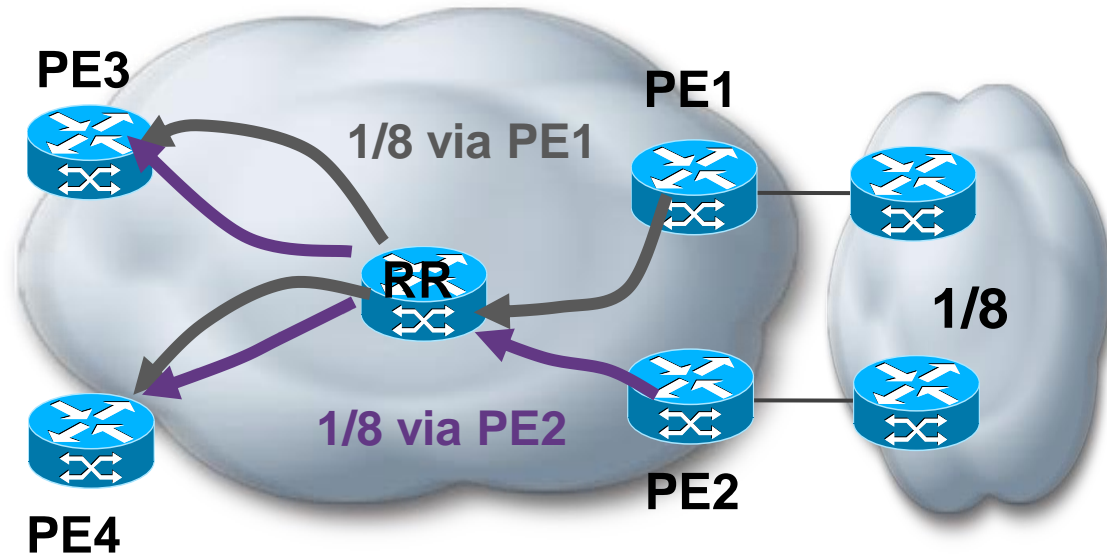
- What is it, and why?



- PIC is the ability to restore forwarding without resorting to per prefix operations.
- Loss Of Connectivity does not increase as my network grows (one problem less).

BGP AddPath

- New Capability to allow a BGP speaker to advertise more than one path (“The holy grail”)
- Available in IOS-XR 4.0, IOS-XE 3.7, 15.2(4)S, 15.3T
- Requires support for this functionality on RR and PEs



And Then There Were LFAs

- Loop Free Alternate: a routing protocol calculates the next-best hop should a link fail (per-link LFA) or a particular prefix become unreachable (per-prefix LFA)
- EIGRP's concept of Feasible Successor is per-prefix LFA
- Recent LFA technologies simply apply FS logic to link-state protocol

LFA in Link-State Protocols

- WWLSPD (What Would Link-State Protocols Do?)
- OSPF and ISIS can apply the same concept as EIGRP's FS
 - Calculate the second-best NH in the event of a failure
 - ...variants can handle link or node failure
- Cannot work in all topologies
 - Same as EIGRP; not all topos have a FS



Q & A

Complete Your Online Session Evaluation

Give us your feedback and receive a Cisco Live 2014 Polo Shirt!

Complete your Overall Event Survey and 5 Session Evaluations.

- Directly from your mobile device on the Cisco Live Mobile App
- By visiting the Cisco Live Mobile Site www.ciscoliveaustralia.com/mobile
- Visit any Cisco Live Internet Station located throughout the venue

Polo Shirts can be collected in the World of Solutions on Friday 21 March 12:00pm - 2:00pm



Learn online with Cisco Live!

Visit us online after the conference for full access to session videos and presentations.

www.CiscoLiveAPAC.com



CISCO TM

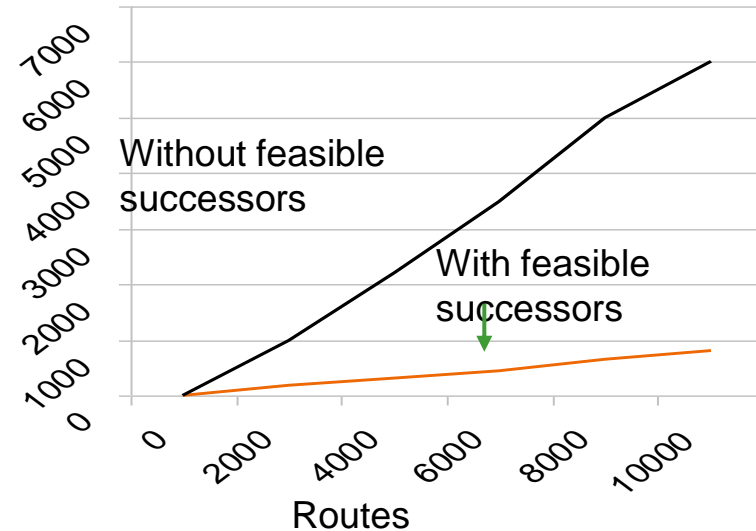


Network Convergence: EIGRP



Feasible Successors

- Whether an alternate path is a feasible successor or not makes a large difference in convergence.
- In this test, switching from the best path to a feasible successor takes less than 1 second; switching to some other neighbour takes about 6 seconds.
- It's important to consider not only the best paths through an EIGRP network, but also the feasible successors.





Feasible Successor

- Whether the next best path is considered loop free by EIGRP (a feasible successor) or not has a large impact on convergence times.
- Don't just consider the best path from every point in your network, but also the next best path.
- Determine how best to set up your path metrics to improve convergence performance.
- Always use the delay metric to engineer your routing, never the bandwidth metric!

EIGRP



- Design to have feasible successors if you can
- Always use the delay metric to engineer your routing, never the bandwidth metric!
- Bound the query domain!



CISCO™