Fast Failover: Marketing and Reality

Ivan Pepelnjak (ip@ipSpace.net)
Network Architect

ipSpace.net AG

Who is Ivan Pepelnjak (@ioshints)

Past

- Kernel programmer, network OS and web developer
- Sysadmin, database admin, network engineer, CCIE
- Trainer, course developer, curriculum architect
- Team lead, CTO, business owner

Present

- Network architect, consultant, blogger, webinar and book author

Focus

- SDN and network automation
- Large-scale data centers, clouds and network virtualization
- Scalable application design
- Core IP routing/MPLS, IPv6, VPN

CISCO CCIE EMERITUS

vmware vEXPERT

Service Restoration Target: 50 msec

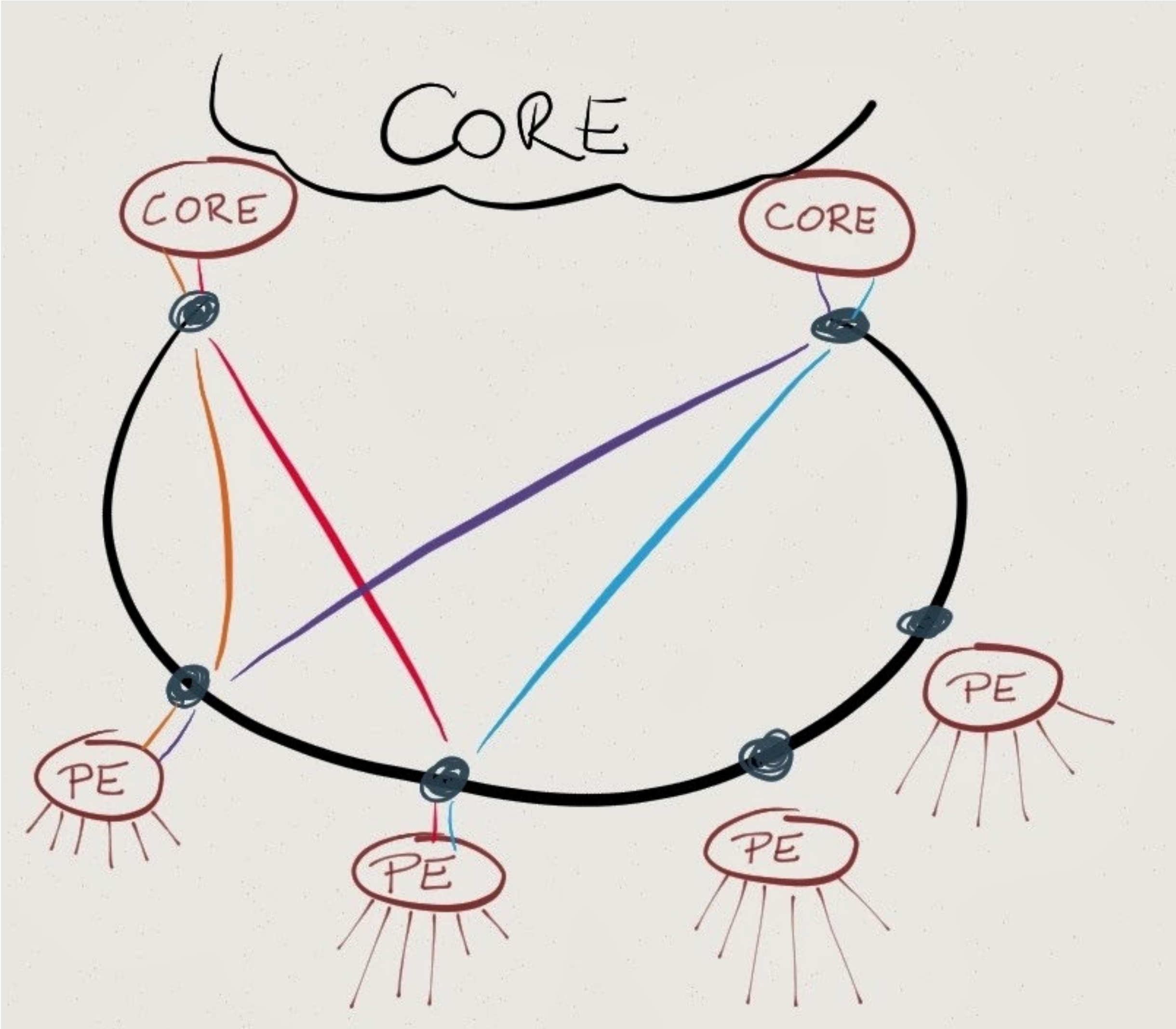
Yeah, No Big Deal

Moment of Truth: DT Terastream Presentation (PLNOG 11 / 2014)

Ian Farrer

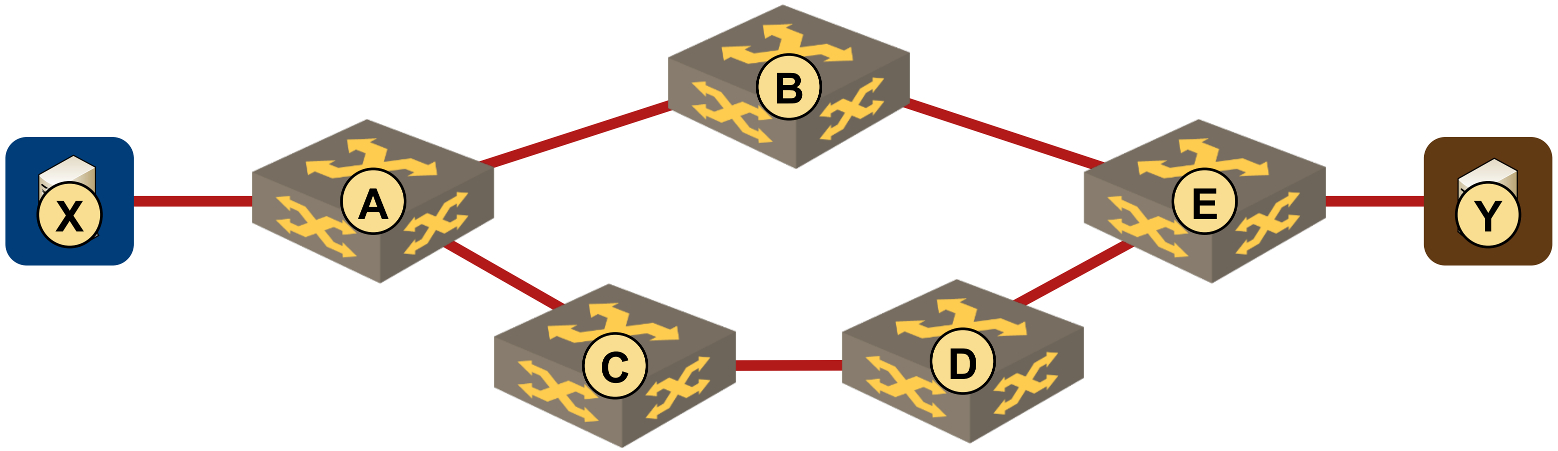
Network Architect

Deutsche Telekom



Ian:	We don't need MPLS or traffic engineering
Me:	How do you reach 50 msec convergence?
Ian:	We don't. There's no contractual need for it… and we can easily reach the convergence time we promised by tweaking IS-IS timers

Fast Failover: Challenge



After encountering a change in network topology, how quickly can we find an alternate topology?

- How is the change detected?
- Who reports the change?
- What happens next?
- How fast can we make it?

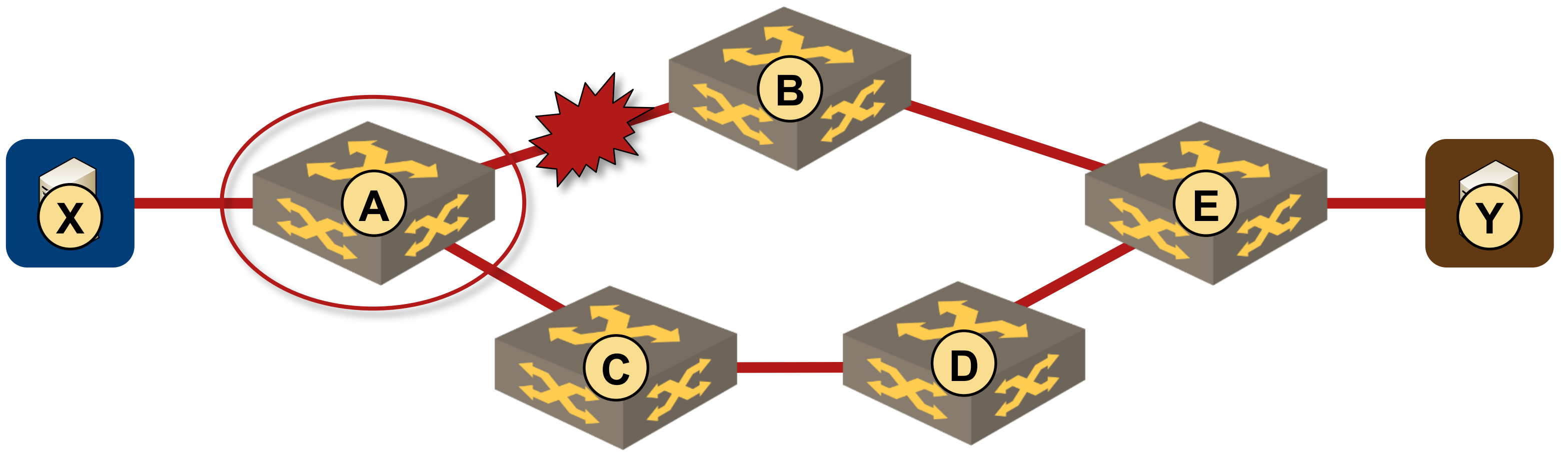
Don’t Rush into Technology Details

- How often do you experience failures?
- What is their impact?
- What is acceptable convergence time?
- How fast is good enough?

Consider the bigger picture

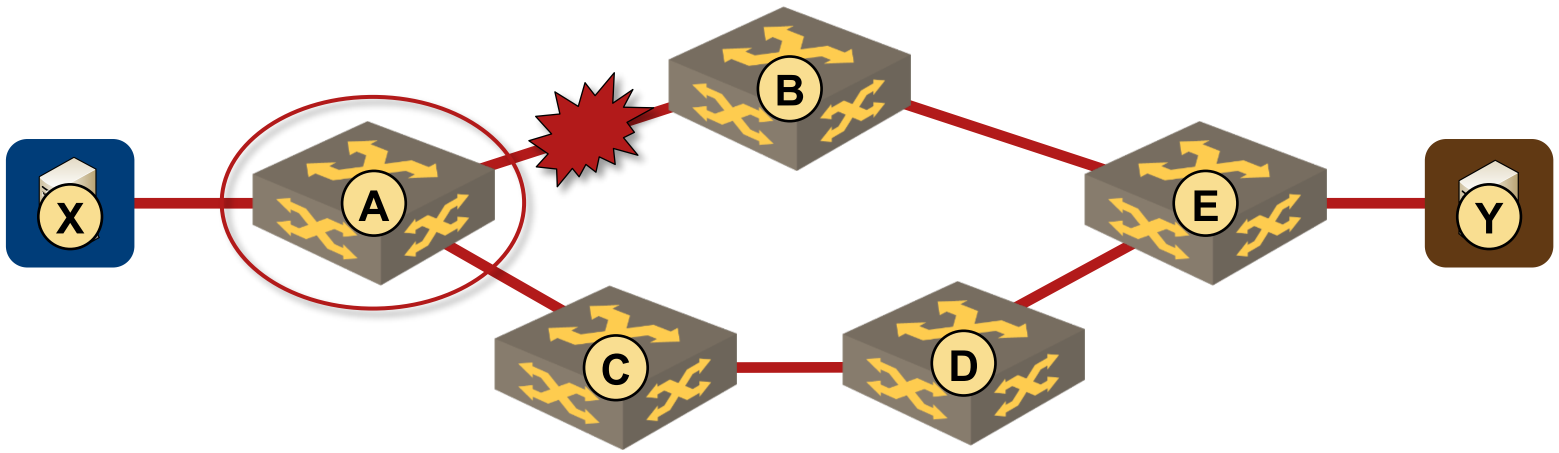
- Can you detect the failure fast enough?
- How many false positives will you get?
- Can you reach the desired convergence time?
- Will fast failure detection make the network unstable?

Components of Convergence Time



- Detect failure
- Find alternate paths
- Update routing and forwarding table

Optimizing Convergence Time



Detect failure

- Reliable hardware failure detection
- Simple and fast liveliness protocols (**BFD**)

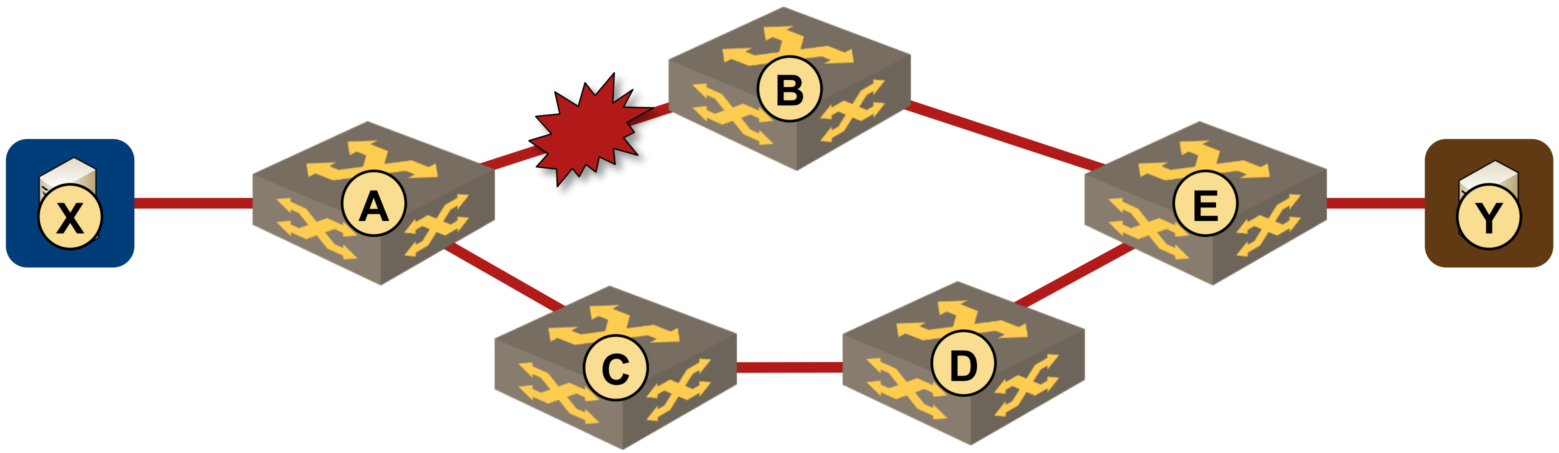
Find alternate paths: Fast Reroute

- Pre-establish alternate path (**MPLS-TE FRR**)
- Precompute alternate paths (**IP FRR – LFA, rLFA, TI-LFA**)

Update routing and forwarding table

- Minimize updates (**PIC**)
- Pre-install alternate paths (**PIC + FRR**)

How Is a Link or Node Failure Detected?



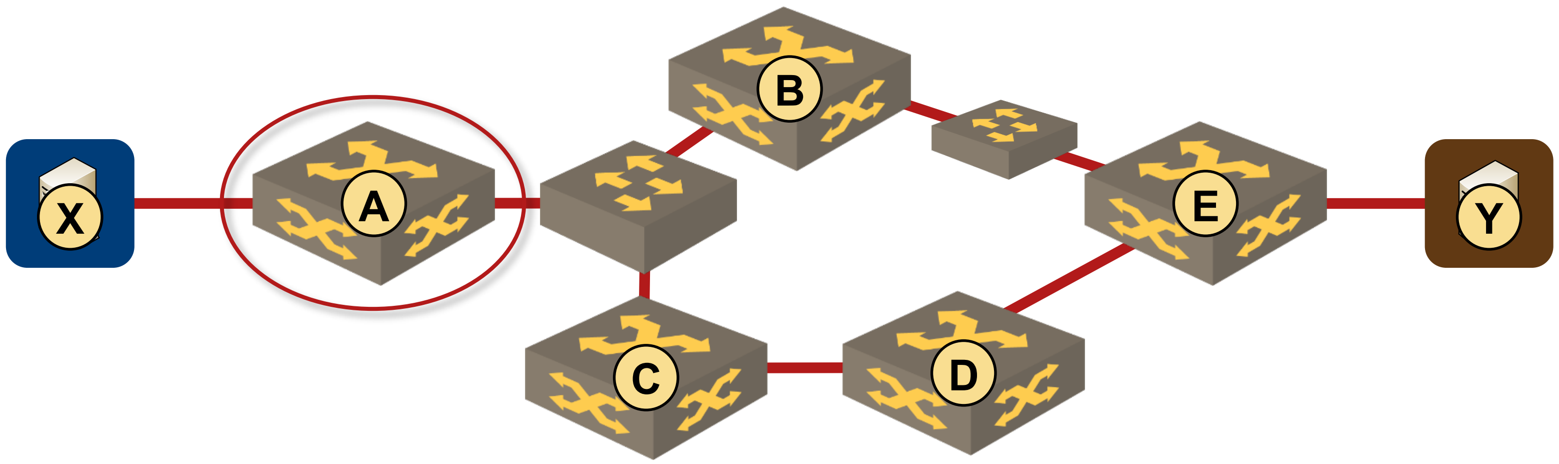
External triggers

- Interface is shut down
- Loss of carrier (light)
- Lower-layer protocol reports a failure (Link Fault Signalling, LACP, UDLD, Ethernet CFM, BFD)

Routing protocol detects the failure

- Adjacency or keepalive protocol timeout (OSPF, EIGRP, IS-IS)
- Transport layer timeout (BGP or LDP)

Complicating Failure Detection



Detecting adjacent node failure

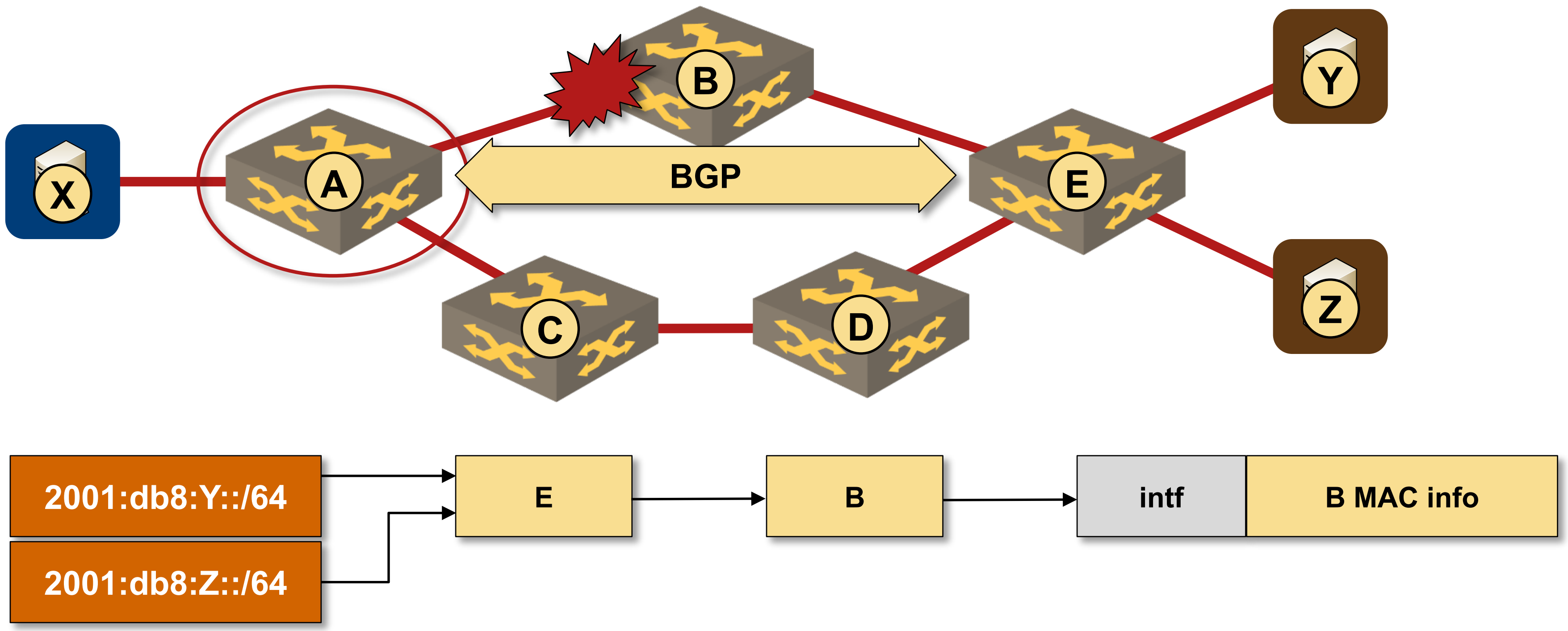
- Might result in link failure on point-to-point links
- Needs active probes on multi-access networks

Byzantine faults (failures)

- Gray failures
- Malfunctioning third-party equipment on point-to-point path

See https://en.wikipedia.org/wiki/Byzantine_fault for more details

Updating Forwarding Tables: Prefix-Independent Convergence



Example: Adjustment to the forwarding table on IGP change (BGP next hop unchanged)

- BGP next hop points to a different IGP next hop
- No prefixes information or BGP path list is changed

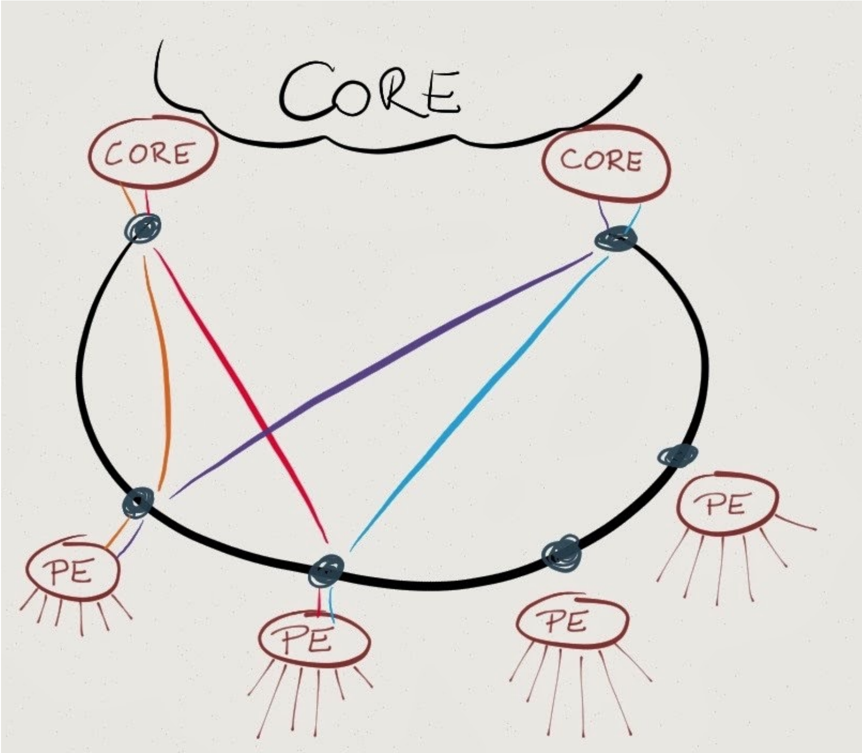
Summary: Adjust Expectations, Design Your Network

Adjust Expectations

- Set realistic targets
- Figure out most common failures
- Focus on common failures, not exotic ones

Design Your Network

- Avoid snake oil (NSF, SSO, GR, NSR)
- Identify components of convergence time
- Focus on the largest components
- Minimize RIB/FIB updates



Triangles Are Better than Squares (and Rings Suck)

- Local failover is all you need when you have two uplinks (triangle)
- You need at least LFA for fast failover when you have two routers per site (square)

Questions?

Web:	ipSpace.net
Blog:	blog.ipSpace.net
Email:	ip@ipSpace.net
Twitter:	@ioshints

Data center:	ipSpace.net/NextGenDC
Automation:	ipSpace.net/NetAutSol
Webinars:	ipSpace.net/Webinars
Consulting:	ipSpace.net/Consulting