

# Quantitative Estimations on Performance in IP/MPLS Networks



Dimitri Papadimitriou (dimitri.papadimitriou@alcatel-lucent.be)

DRCN 2007, La Rochelle, France

October 2007

# Introduction

---

- Lots of development currently focused on extending IP/MPLS capabilities to provide for OAM(P) functionality
- Rule of thumbs  
Optimization (over-engineering) introduces complexity and tighter coupling between network components  
Law of Diminishing Returns: “if one factor of production is increased while the others remain constant, the overall returns will relatively decrease after a certain point”  
⇒ Optimization after a certain point only adds complexity and leads to less reliable systems  
⇒ Trade-off must be found between functionality/efficiency - complexity - robustness/scalability
- **Question**: until which point a IP/MPLS OAM functionality needs to be optimized for purpose such as resiliency, performance monitoring, etc. before only adding complexity and decreasing reliability of the system

# Guiding Principles (from RFC 3439)

---

## **Simplicity Principle:**

“Complexity is the primary mechanism that impedes efficient scaling”

⇒ implications on the architecture, design and engineering issues in large scale networks

## **Amplification Principle:**

“Complexity can amplify small perturbations”

⇒ ensure that such perturbations are extremely rare

## **Coupling Principle:**

“Larger systems often exhibit increased interdependence between components”

⇒ unforeseen events due to interaction of simultaneous occurrence of known events

# IP/MPLS Evolution

---

Phase 1: L3 packet forwarding paradigm <-> <FEC = dest.prefix,label> mapping

IP/MPLS in-band OAM functionality ignored

IP/MPLS out-of-band OAM functionality (MIB)

Phase 2: traffic engineering <-> <FEC = Tunnel(next-hop),label> mapping

IP/MPLS in-band OAM functionality attracts more attention

IP/MPLS out-of-band OAM functionality (MIB, control plane)

Phase 3: service delivery (PW) <-> <FEC = Circuit ID,label> mapping

IP/MPLS implicitly associated to the service support

Single-segment PW (adaptation and multiplexing)

Multi-segment PW (networking properties)

IP/MPLS in-band OAM functionality becomes (more) serious concern

# OAM ⇔ FCAPS

---

## **Fault** management: LSP/link fault

- detection/isolation
- notification/(alarm) suppression
- correction (re-routing, protection switching)
- location, correlation
- diagnostic

## **Configuration** management

- LSP/link provisioning operation monitoring and verification (LSP/link status pro-active monitoring)
- Network element/system configuration monitoring and verification

## **Accounting** management (RFC 2975)

- Accounting: collect resource consumption/utilisation data
- Objectives:
  - Trend analysis and capacity planning
  - Billing/charging
  - Auditing

# OAM ↔ FCAPS

---

## **Performance** management

- Performance metrics and alert conditions (thresholds)
- Performance data collection
  - measurement: loss, delay, jitter, etc.
  - statistics/counters: MIBs
- Performance data processing wrt metrics
  - Resource-oriented objectives
  - Traffic-oriented objectives
  - ... user-oriented objectives

## **Security** management

- Data plane mis-connection/leaks from/to sensitive resource areas
- Control and monitor access to network elements/resources
- Prevent accessibility to sensible information without AAA

# Operational Concerns

---

1. Effectiveness and performance of conducted provisioning/re-routing operations
  - Resource selection and allocation
  - Bad LIB/LFIB entry (eq. RIB<->FIB)
2. Troubleshooting (verification/diagnosis) in case of data plane performance degradation
3. Coordination and synchronization between (data plane) resource status/usage and control plane “view”
  - Key for effective traffic-engineering
4. OAM operations in multi-domain / multi-service environment (inter-networking)
5. Unify (and simplify) maintenance and control operations in multi-level MPLS networks

# Fault management - OAM Classes

	Intrusive (frame header**)	Injection (OAM PDU)
Local-label(*)		<b>IP/MPLS:</b> LSP Ping, MPLS BFD, Signaling Specific tools
Domain-wide label	PBT OAM (SA-DA check)	<b>PBB-TE:</b> extension for 802.1ag PB/PBB (Bridged Ethernet): 802.1ag
Globally Unique (label or address)		IP: BFD, ICMP, Traceroute, etc.

(\*) Local-label = per-link/per-platform depending on the technology

(\*\*) frame header = client data traffic frame

# Fault management tools (MPLS)

---

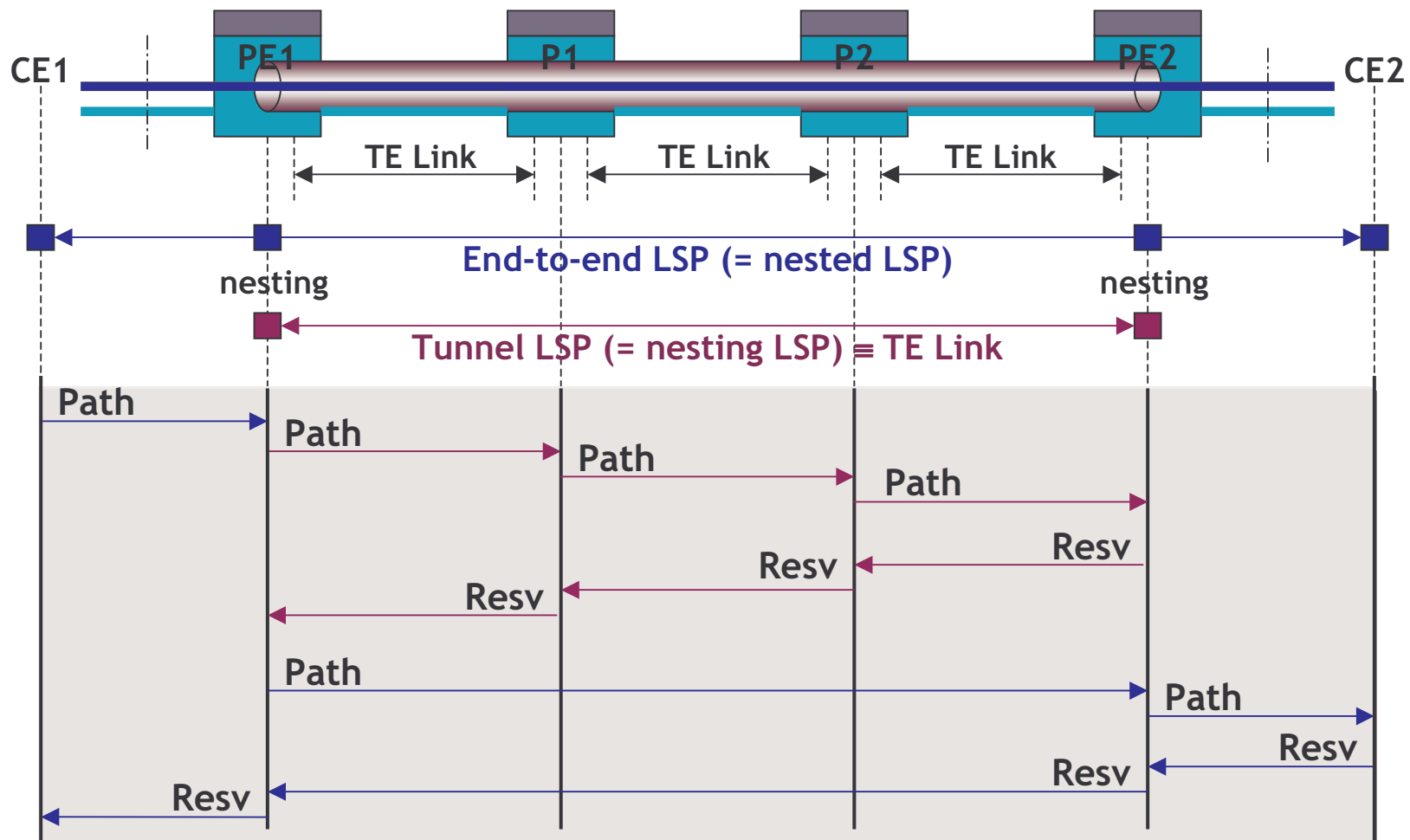
## Result: IP/MPLS fault management tools

- Connectivity check/verification:
  - BFD-MPLS
  - LSP Ping
- Alarms processing
  - Generation
  - Communication/propagation
  - Handling e.g. suppression
- Loopback tests:
  - BFD-MPLS (echo)
  - LSP Ping
- Fault location/diagnosis: LSP Traceroute (= incremental connectivity check)
- Performance monitoring

## Compared to: IP fault management tools

- Connectivity check/verification:
  - BFD
  - ICMP (Echo request/echo reply)
- Alarms processing
  - Generation
  - Communication/propagation
  - Handling e.g. suppression
- Loopback tests:
  - ICMP (echo)
- Fault location/diagnosis: Traceroute (= incremental connectivity check)
- Active and passive performance monitoring (PSAMP, IPPM)

# Positioning of MPLS OAM



# MPLS OAM Levels and Functionality

VPLS and VPWS delivered by IP/MPLS network

Provide and coordinate OAM at the relevant levels in the IP/MPLS network

Proactive and reactive OAM mechanisms which are independent at all levels

Ethernet OAM (ongoing IEEE 802.1ag effort)

VPLS and VPWS rely on

- MPLS tunnel OAM
- Pseudo-Wire OAM
- Service Level OAM

**Service Level**  
e.g VRF-Ping, MAC-Ping

**PW Level**  
e.g VCCV, PW status

**PSN Tunnel Level**  
e.g MPLS OAM

Leverage OAM capabilities of IP/MPLS network to provide enhanced resiliency and fault management compared with native Ethernet

# MPLS OAM Levels and Functionality

---

## CE-to-CE OAM

- IP traffic
  - **IP/MPLS OAM (IP traffic)**
- Sub-IP traffic
  - Ethernet, ATM, FR, etc.
  - OAM specific mechanism associated to Service level in case CE-PE defines client-server boundary

## PE-to-PE OAM

- IP traffic
  - **MPLS Tunnel (BFD, LSP Ping)**
- Sub-IP traffic (PW over MPLS PSN)
  - PW level (LDP PW status, VCCV)
  - **MPLS tunnel (BFD, LSP Ping)**

# MPLS OAM Levels and Functionality

---

## Fault Indication

- PW Status signaling (LDP)

## Fault Detection

- **BFD (Bi-directional Forwarding Detection)**

## Continuity Check - On-demand connectivity check

- LSP Ping
- **BFD**

## Path/LSP Trace

- LSP Traceroute (relies on LSP Ping)

## Service Check

- VRF Ping/Traceroute
- MAC Ping/Traceroute

## Performance

- MIBs

# Inter-domain / Multi-domain Fault management

---

OAM functions: part of requirement set for establishing inter-connects

- Fault management
  - Connectivity verification
  - Alarms communication and handling
  - Loopback tests
  - Fault diagnosis (e.g. traceroute)
- Performance management
  - Performance measurement

Note: distinction between intra- vs Inter-domain boundary specifics

- Confidentiality: control what is returned in traceroute, route record, etc.
- Policy: filter/modify control plane error/alarm propagation
- Scaling: block performance OAM packets

# Inter-domain / Multi-domain Fault management

---

Some elements may be intentionally confined within a domain

## Fault management tools (MPLS data plane)

- ensure capability support by intermediate end-points
- connectivity check/verification: BFD-MPLS, LSP Ping
- alarms handling
- loopback tests: BFD-MPLS, LSP Ping
- fault location/diagnosis: LSP Traceroute (= incremental connectivity check)

## Fault and configuration management tools (control plane)

- administration: admin\_status (notify/path/resv message)
- alarm handling: alarm\_spec
- fault notification: error\_spec in notify/error message
- fault location/diagnosis: error\_spec, *route recording* (label)
- path verification

# BFD Overview

# Bi-directional Forwarding Detection (BFD)

---

## Goals

- Detection of forwarding plane-to-forwarding plane connectivity (including links, interfaces, tunnels etc.)
  - Distinction between forwarding plane connectivity vs. control plane connectivity
  - => FEC/label processing independent
- Single mechanism independent of media, routing protocol, and data protocol
- No changes to existing IP routing protocols
  - Result in faster convergence of routing protocols, particularly on shared media (Ethernet)
- Detection of one-way link failures

# BFD-MPLS usage

---

BFD can be used to detect a data plane failure in the forwarding path of LSPs (FEC: RSVP Session, IP Prefix, Circuit ID, VPN)

## Advantages

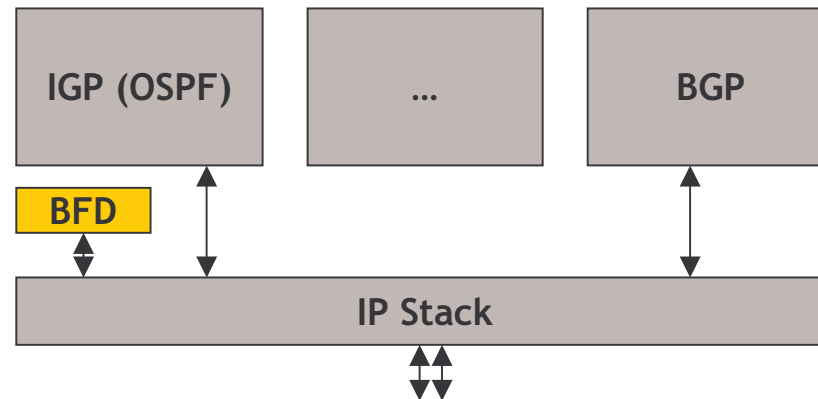
- Support for fault detection for greater number of LSPs (BFD lightweight mechanism)
- Fast detection  $\Rightarrow$  detection with sub-second granularity (BFD designed for sub-second fault detection intervals)
- Negotiable Transmit/Receive and Detection interval
- Support of Periodic (Asynchronous) and Demand-mode both modes allows for loopback (Echo)

# BFD-MPLS Overview

BFD allows LSRs to periodically exchange messages to verify operational status of LSP(s)

BFD control messages

- Sent along the same data path as the LSP being verified
- Processed by BFD processing module (software or hardware)
- LSRs negotiate the transmission interval of the control messages



Note: LSPs are only declared to be operational when two-way communication has been established between LSRs (this does not preclude unidirectional LSPs)

# BFD Sessions

---

BFD session boot-strapped to initiate fault detection for a particular <LSP, FEC>

BFD session bootstrapping requires exchange

- Request (from ingress LSR to egress LSR): includes local discriminator assigned by the ingress LSR for the BFD session (used as the My Discriminator field in the BFD session packets sent by the ingress LSR)
- Response (from egress LSR to ingress LSR): includes local discriminator assigned by the egress LSR for the BFD session (used as the My Discriminator field in the BFD session packets sent by the egress LSR)

Then

- Head-end LSR (for this BFD <LSP,FEC> session) sends a BFD control packet to the tail-end LSR with Your Discriminator field set to the local discriminator of the egress LSR
- Tail-end LSR demultiplexes the BFD session based on the received Your Discriminator field and sends control packets to head-end LSR with the Your Discriminator field set to the local discriminator of the head-end LSR
- The head-end LSR can use this field to demultiplex the BFD session

**Note:** bootstrap of BFD session can be performed via the control plane (LSP ping) or management

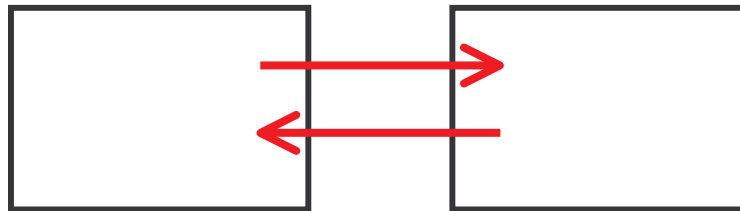
# BFD Modes (1)

---

## Two selectable operating modes

- **Asynchronous mode:** BFD PDU in each direction

Systems periodically send BFD control packets to one another, and if a number of those packets in a row are not received by the other system, the session is declared to be down



- **Demand mode:**

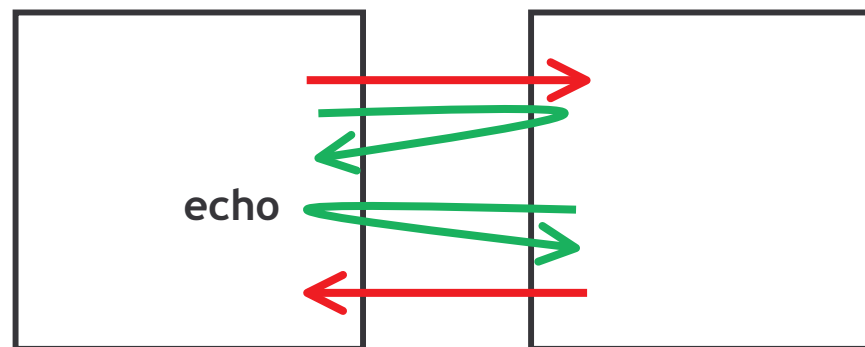
Each system has an independent way of verifying that it has connectivity to the other system (once a BFD session is established, the systems stop sending BFD control packets, except when either system feels the need to verify connectivity explicitly)

## BFD Modes (2)

---

### Echo function

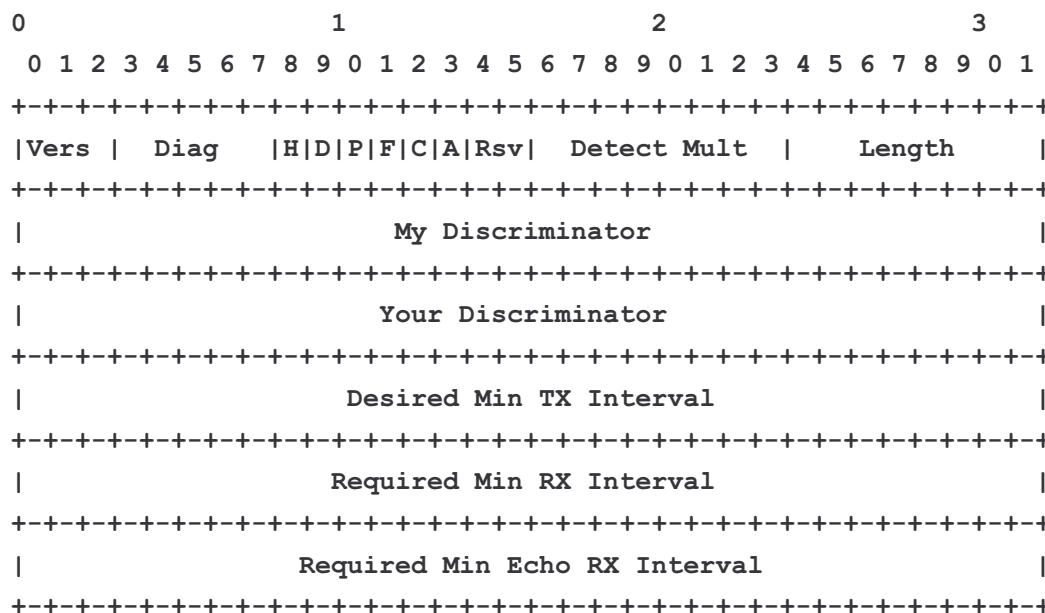
- Definition: stream of BFD Echo packets transmitted such that the other system **loopback** them through its forwarding path. If a number of packets in a row of the echoed data stream are not received, the session is declared to be down
- Echo function may be used with either Async or Demand modes (and individually in each direction)
- Note: since the Echo function is handling the task of detection, the rate of periodic transmission of Control packets may be reduced (in case of Async mode) or completely eliminated (in case of Demand mode)



# BFD Control Packet

BFD Control packets sent using encapsulation appropriate to the environment

Payload of a BFD Control packet:



I Hear You (H) bit: set to 0 if transmitting system either not receiving BFD packets from the remote system, or in the process of tearing down the BFD session for some reason (see the Elements of Procedure slides)

Demand (D) bit: set if transmitting system wishes to operate in Demand Mode

Poll (P) bit: set if the transmitting system is requesting verification of connectivity, or of a parameter change

Final (F) bit: set if the transmitting system is responding to a received BFD Control packet that had the Poll (P) bit set

# BFD Procedure

When:

- Either downstream PE2 doesn't receive control messages from upstream PE1 during a certain number of transmission intervals (number provisioned by the operator)
- or if PE2 determines in another way that communication is lost

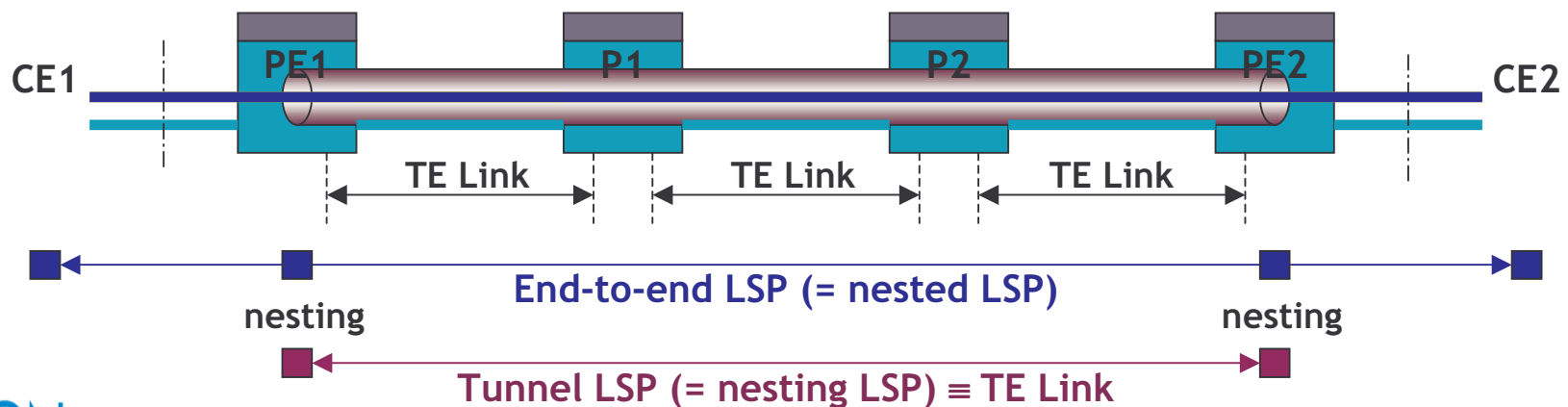
Then, PE2 declares that LSP (direction from PE1 to PE2) as down

- PE2 stores the cause (e.g. "control detection time expired") and sends a message to PE1 with H=0 (i.e., "I don't hear you")

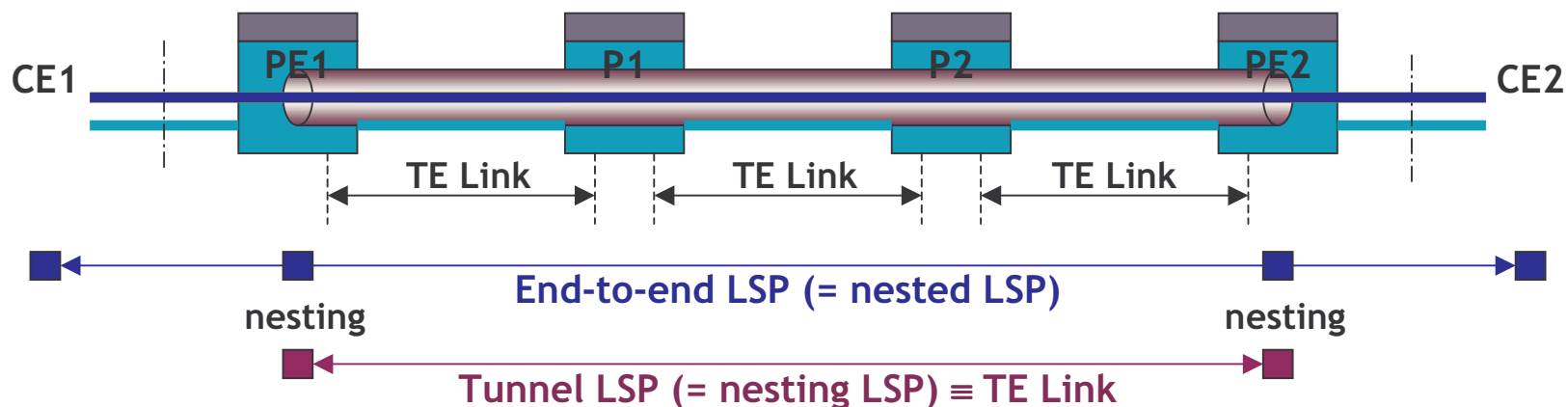
Then, PE1 declares the LSP in the direction from PE1 to PE2 down

- PE1 stores as cause: "neighbor signaled session down"

Note: PE2 may send a FDI indication to CE2 and PE1 may send an RDI indication to CE1



# BFD Diagnostics



BFD diagnostics that define the local system's reason for the last transition of the session from UP to some other state:

- |                                    |                               |
|------------------------------------|-------------------------------|
| 0 : No Diagnostic                  | LSP is UP                     |
| 1 : Control Detection Time Expired |                               |
| 2 : ECHO Function Failed           | not applicable to ASYNC mode  |
| 3 : Neighbor Signaled Session Down |                               |
| 4 : Forwarding Plane Reset         | Local equipment failure       |
| 5 : Path Down                      | Alarm Suppression             |
| 6 : Concatenated Path Down         | Propagating access link alarm |
| 7 : Administratively Down          |                               |

# BFD Packet encapsulation

---

BFD control packets: sent by head-end LSR are encapsulated in data plane entity (e.g. label stack) that corresponds to the FEC for which fault detection is being performed

- UDP packet with a well known destination port (TBA) and a source port assigned by the sender.
- Source IP address = routable address of the sender
- Destination IP address = (randomly chosen) address from 127/8

BFD control packets sent by tail-end LSR

- UDP packets (source port is well-known UDP port (TBA))
- Source IP address = routable address of the replier
- Destination IP address and UDP port are copied from the source IP address and UDP port from the received control packets)

Note: BFD control packets sent by the egress LSR to the ingress LSR are routed based on the destination IP address

# BFD Applicability

# Fault management - OAM Classes

	Intrusive (frame header)	Injection (OAM PDU)
Local-label		IP/MPLS: LSP Ping, <b>MPLS BFD</b>
Domain-wide label	PBT OAM (SA-DA check)	802.1ag ext. for PBB-TE 802.1ag for PB/PBB (Bridged Ethernet)
Globally Unique (label or address)		IP: BFD, ICMP, Traceroute, etc.

# BFD and On-demand/continuous CC/CV

---

## Continuous CV can be provided by BFD *Async mode*

- Both end systems periodically send BFD packets
- If a number of those packets in a row are not received by the other system, the session (i.e. the <LSP, FEC> association) is declared to be down

## On-demand CV can be provided by BFD *Demand mode*

- Each system independently verifying its connectivity to the other system
  - Note: once a BFD session is established, the systems stop sending BFD control packets, except when either system feels the need to verify connectivity explicitly with the remote system)
- Demand mode is negotiated by virtue of both end systems setting the Demand (D) bit in its BFD Control packets
  - => both systems must request Demand mode for it to become active
- Demand mode may require that some other mechanism is used to imply continuing connectivity between the two systems

# BFD and On-demand/continuous CC/CV

---

Each mode is driven by a specific Detection Time computation (at local system)

- Async mode: Detection Time = [value of DetectMult received from the remote system, x agreed Tx interval (the greater of RequiredMinRxInterval and the last received DesiredMinTxInterval)]
- Demand mode: Detection Time = [DetectMult x agreed Tx interval (the greater of RequiredMinRxInterval and the last received Desired MinTxInterval)]

**Continuous CV** expected with lowest Tx interval for LSP trunks

- Realistic Tx interval for trunks: O(10ms) - for nested LSP: O(100ms) depending on the nested LSP criticality and LSP hierarchy depth
- Once negotiated, a default period of [Detection time multiplier (DectMult) x Tx interval] should elapse before declaring BFD session down (= Detection Time)

**On-demand CV** expected when the number of LSPs to be monitored is relatively high compared to the number of trunks

Other dimensions driving BFD mode: LSP granularity, LSP priority, etc.

# BFD and Loopback

---

## Loopback can be provided using the *BFD Echo mode*

- Stream of BFD Echo packets transmitted such that the other system loop them back through its forwarding path
- If a number of packets in a row of the echoed data stream are not received, then BFD session is declared to be down
- Echo function applicability
  - Used by default with Demand mode (default)
  - Can run independently in each direction between a pair of systems
    - => as BFD allows independent transmission rates in each direction, if the Echo function is only being run in one direction, the system not running the Echo function usually sends fairly rapid Control packets in order to achieve its desired detection time

# Impact on availability

# Analogy

---

Starting point: routing protocol convergence

The only constant is change

- Non pre-planned
  - Equipment failures (soft vs hard)
    - May affect unicast FIB/RIB
    - May affect multicast FIB/RIB
  - Link failure
- Pre-planned
  - Routing-protocol configuration changes
  - Planned maintenance

# Convergence after a routing protocol failure occurrence

## Failure detection

- E.g. Router detects an incident link has failed

=> Faster detection

- **Smaller hello timers**
- **DLL technologies that can detect failures**
- **BFD (DLL independence, minimize overhead)**

## Failure notification

- Router informs other routers about the change

=> Faster flooding/notification

- Flooding immediately
- Sending link-state update packets with high-priority

## Path (re-)computation and selection

- Routers compute new paths avoiding the link

=> Faster computation

- **Faster processors on the routers**
- Incremental Dijkstra algorithm on LS topology

## Forwarding-table update

- Routers update their forwarding tables
- Data traffic starts to flow over the new path

=> Faster forwarding-table update

- Data structures supporting incremental updates

# BFD Example

## Control Plane Independent (C)

- If C=1, transmitting system's BFD implementation does not share fate w/ control plane
  - BFD when implemented in the forwarding plane and can continue to function after control plane failure
- If C=0, transmitting system's BFD implementation shares fate with its control plane.

Failure case	Case 1: Routing engine failure	Case 2: Routing and forwarding engine failure
Detection	C=1: Forwarding may still be running and if BFD in hardware => BFD OK C=0: (shared state) => BFD down	BFD down (C=0, C=1)
Notification	Link-state updates LSP error indication	Link-state updates LSP error indication
Re-routing	e.g. IP-FRR/LFA or MPLS FRR	e.g. IP-FRR/LFA or MPLS FRR

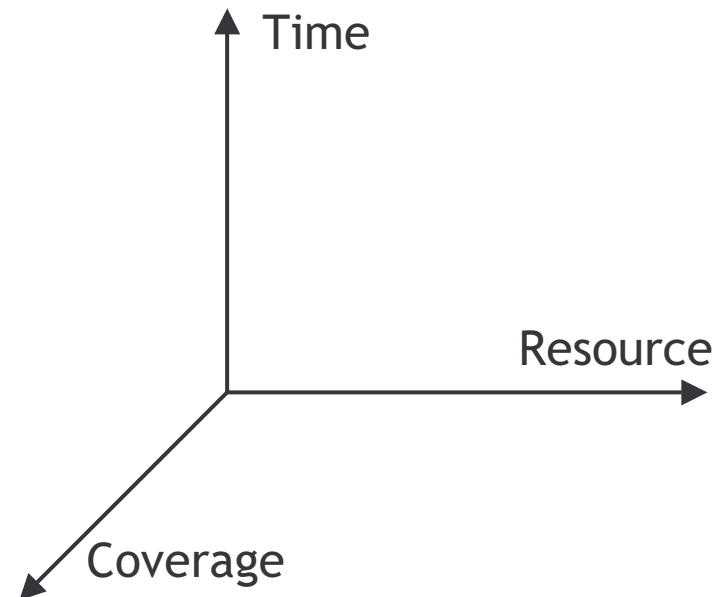
# Performance criteria: 3-dimension space

**1. Time:** Period of time needed to reach a certain degree of adaptation / recovery to a change / failure occurrence (recovery => network routing/ convergence time)

**2. Coverage:** Percentage of (connectivity x traffic) that has recovered from the recoverable set of (destinations x traffic)

**3. Resources:** How many “resources” are required to implement / execute the mechanism (4 steps)

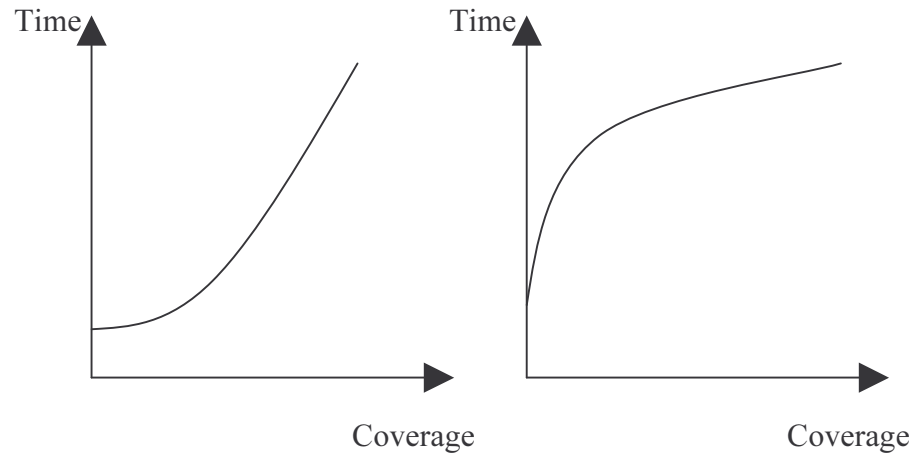
- multi-parameter variable, with some fixed, and some variable parts
- can be subdivided into infrastructure and operational resources (node/link and system resources: CPU/memory)



# Performance criteria

---

1. Convergence time wrt resource usage => Resource versus time is fundamental analysis plane (2-dimension)  
... and especially the effects on the data traffic
2. Time versus coverage is another important analysis plane to be verified by a IP/MPLS resiliency solution (assuming fixed resources)



# Events during convergence

---

## Transient inconsistencies

- RIB: Routers have different views of the network
- FIB: Forwarding decisions may be inconsistent

## Effects on data traffic

- Black-hole: packet loss
- Micro-loops: packets going in circles
- Delay: packets going on very long paths
- Out-of-order: new packets arrive before old ones

# Analysis Steps

---

## Step 1: Analyze causes

- Technology dependent
  - connectionless forwarding (IP or Ethernet 802.1) vs frame/packet switching (MPLS)
  - correlation of the forwarding/switching decision
- System engineering dependency
- Network engineering dependency

## Step 2: Assess consequences/effects and relative occurrence / importance

- Traffic engineering dependency

## Step 3: Define tools and applicability

Analysis: tools reliability and coherency

# Fault management - OAM Classes

	Intrusive (frame header)	Injection (OAM PDU)
Local-label		IP/MPLS: LSP Ping, MPLS BFD
Domain-wide label	PBT OAM (SA-DA check)	802.1ag ext. for PBT 802.1ag for PB/PBB (Bridged Ethernet)
Globally Unique (label or address)		IP: BFD, ICMP, Traceroute, etc.

## Domain vs Local-label

---

PBB-TE resulted from the observation that label-swapping technologies such as MPLS may introduce two classes of erroneous label operations:

- **Class 1: bad forwarding entry (incorrect label\_out) OR incorrect label (stack) operation** (e.g. incorrect read, head-end label insertion, etc.):  
but errors **affecting any swapping technology** and at a lesser extend any switching technology (read and head-end insertion errors only but not write errors)
- **Class 2: bad forwarding entry (incorrect next\_hop) OR incorrect outgoing interface selection:**  
but errors **affecting any switching technology** (so also PBB-TE)

Both classes result in **mis-merge** (=> traffic arriving potentially at wrong destination or resulting in network resource contention inside the network)

**Condition:** if the incoming label matches an already installed entry (with the same value) in the next\_hop forwarding

## Domain vs Local-label

---

PBB-TE designed with explicit requirement to provide for **in-band detection of mis-merge traffic** as close as possible to failure occurrence i.e. at the next-hop (instead of the destination like with classical OAM techniques)

Problems occur with correct forwarding entries but either

- incorrect label operation
- or incorrect outgoing interface selection (mismatch)

Both result in permanent errors that can be detected but cannot be corrected  
=> Error prevention: drop mechanism (to prevent error in traffic propagation)

# Domain vs Local-label

---

## PBB-TE

- less prone to implementation errors resulting from mismatch between forwarding entry and incorrect label stack operation because it **minimizes the number of label operations (only read, no write)**
- enables in-band detection of mismatch between forwarding entry and incorrect outgoing interface selection i.e. mis-merge detection

**Price to pay** to detect (not correct) such errors that may occur with a low probability

- PBB-TE relies on <B-DA, B-VID> domain-wide invariant identifiers: encoded in the frame header) for frame forwarding it must be complemented by **B-SA processing for mis-merge detection**
  - => RPF check for incoming unicast traffic
- In-band mis-merge detection (in case of incorrect outgoing interface selection) is mandatory when data path multiplexing inside PBB network
  - => path multiplexing increases complexity of PBB-TE network

# Step 1: Analyze causes

P1 = Probability of incorrect label\_out (event A)

At next-hop

- if label\_in exists in LFIB entry => error propagation/mis-merge (probability P3)
- if label\_in does not exist in LFIB entry => drop

P2 = Probability of incorrect outgoing I/f (event B)

At next-hop

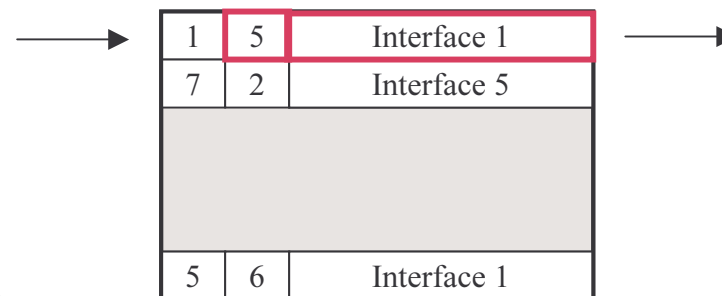
- if label\_in exists in LFIB entry => error propagation/mis-merge (probability P3)
- if label\_in does not exist in LFIB entry => drop

Event A:

- bad LFIB entry (label\_out only)
- or label stack operation (switching or swapping)

Event B:

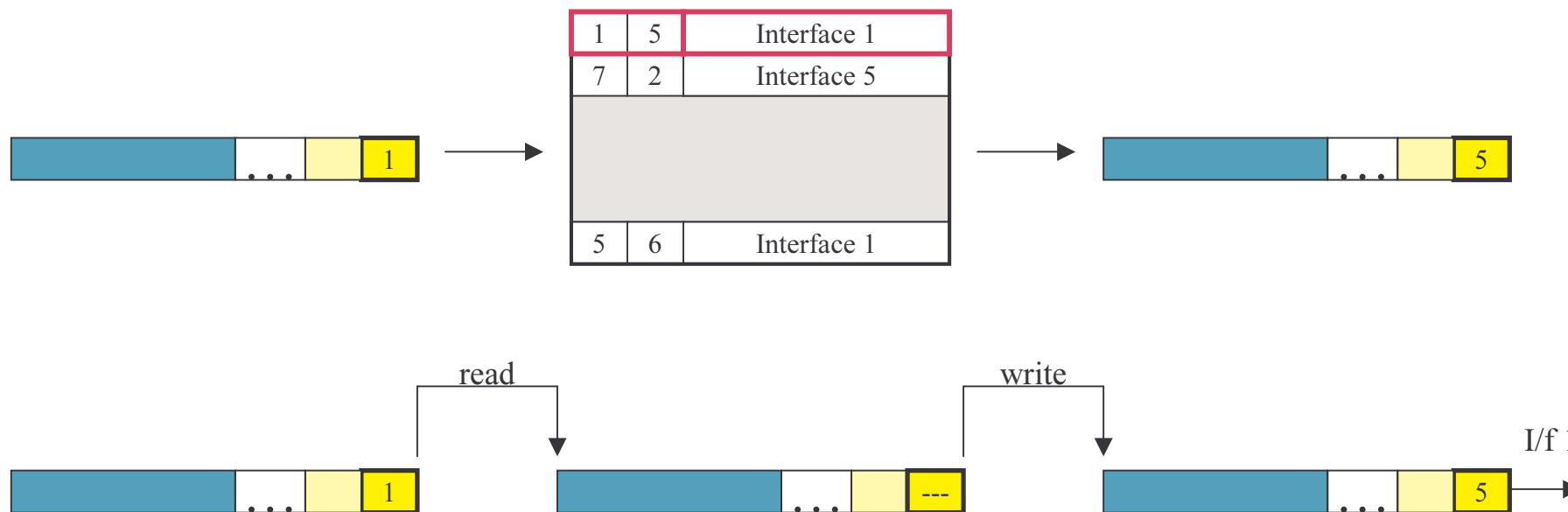
- bad LFIB entry (next hop only)
- or outgoing interface selection



# Step 1: Analyze causes

## Bad LFIB entry

next hop: label processing independent  
 label\_in -> label\_out values: label processing independent } => affects any switching technology

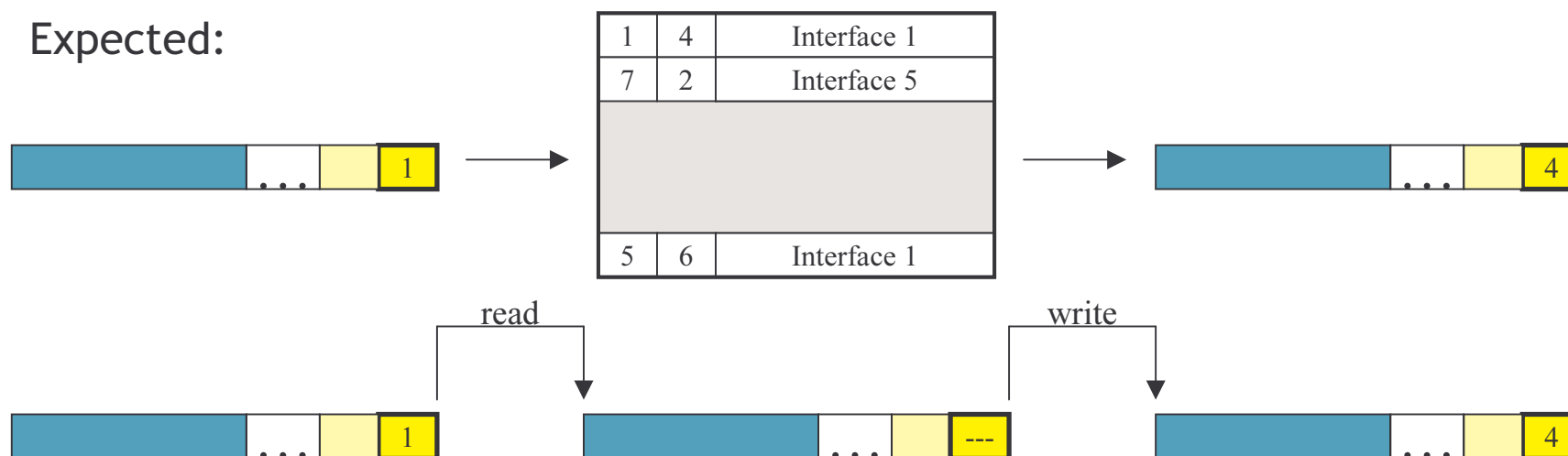


# Step 1: Analyze causes

Problems comes from false positive

- Correct LFIB entries
- Incorrect label processing operations

Expected:



Occurrence:

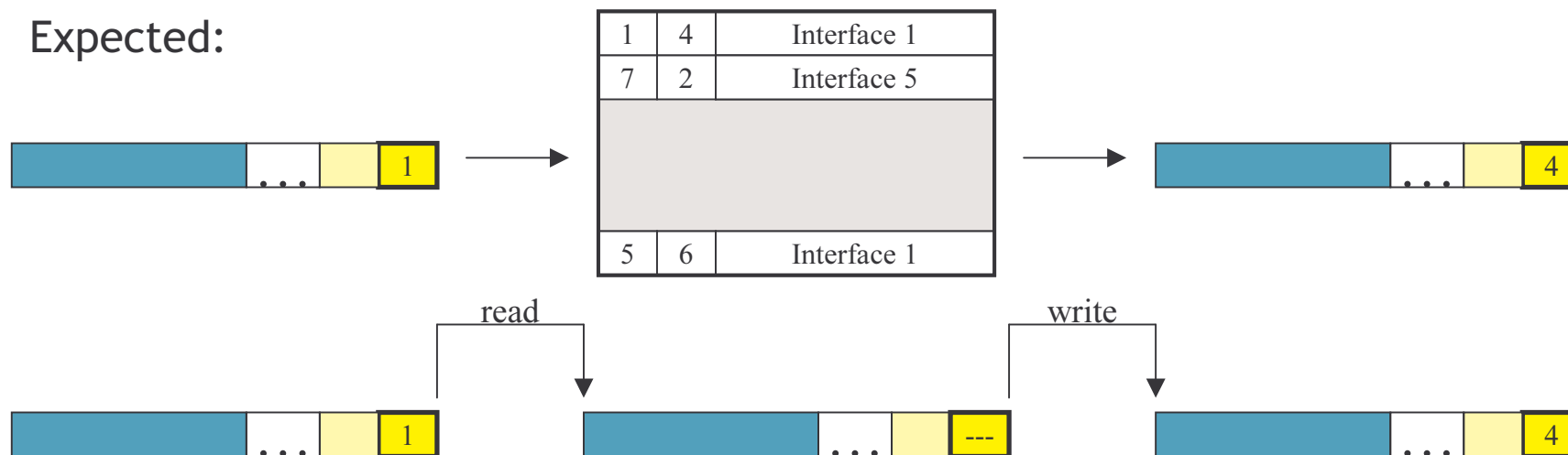


# Step 1: Analyze causes

Problems comes from false positive

- Correct LFIB entries
- Incorrect label processing operations

Expected:



Occurrence:

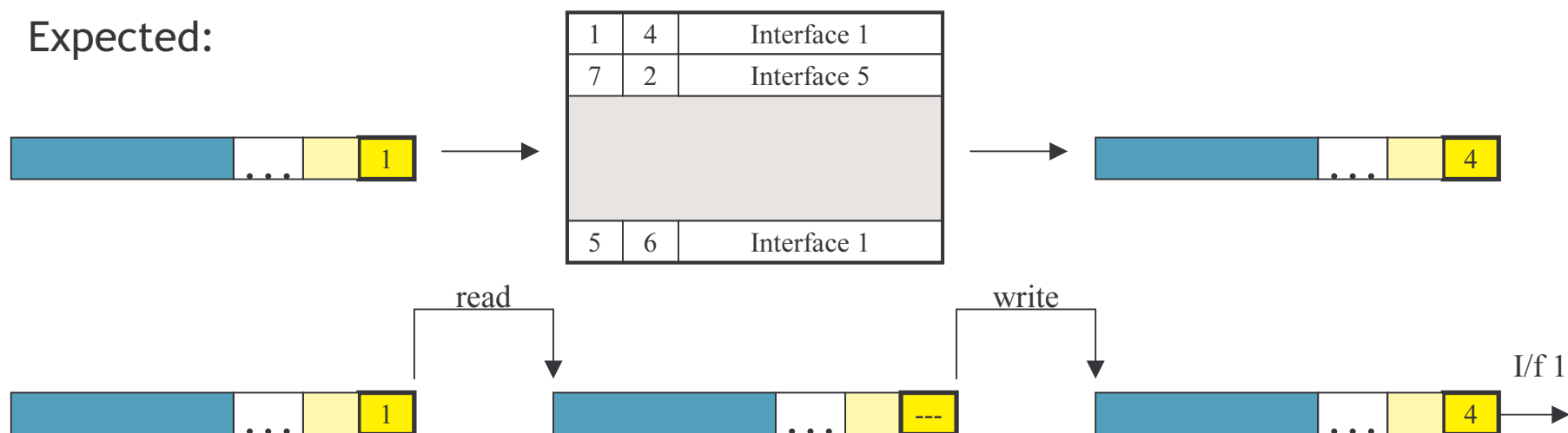


# Step 1: Analyze causes

Problems comes from false positive

- Correct LFIB entries
- Incorrect interface selection

Expected:



Occurrence:



## Step 2: Consequences and Relative importance

---

### Event A:

bad LFIB entry (next hop, label\_out)  
or label stack operation (switching or swapping=>switching)

### Event B:

bad LFIB entry (next hop only)  
or outgoing interface selection

### Detection mechanism required

- LFIB Entry: can be detected in-band e.g. tracing / out-of-band e.g. MIB
  - Note: can be corrected (not really an issue)
- Label stack operation and outgoing interface selection: permanent errors can be detected in-band (not of out-of-band or without correlation)
  - Note: can not be corrected => requires avoidance/drop mechanism to prevent error propagation
- Outgoing interface selection problem affects any switching technology

## Step 2: Consequences and Relative importance

---

P1 = Probability of incorrect label\_out (bad label stack operation)

If erroneous label\_out at hop i

At next-hop (i+1)

- if label\_in exists in LFIB entry => error propagation/mis-merge (probability P3)
- if label\_in does not exist in LFIB entry => drop

P2 = Probability of incorrect outgoing I/f

If erroneous outgoing interface at hop i

At next-hop (i+1)

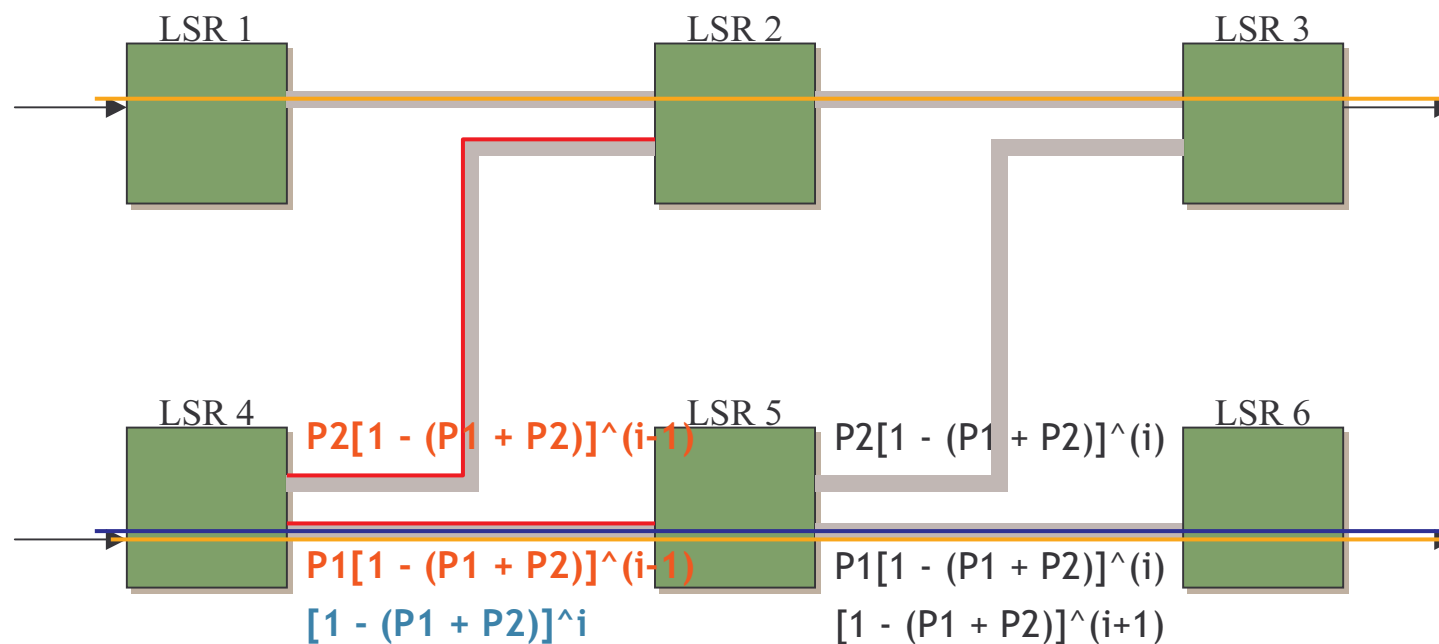
- if label\_in exists in LFIB entry => error propagation/mis-merge (probability P3)
- if label\_in does not exist in LFIB entry => drop

## Step 2: Error probability

Example:  $i = 5$ ,  $P1 = 0.001$ ,  $P2 = 0.0001$

At LSR 4:  $P2[1 - (P1 + P2)]^{(i-1)} \sim 10^{-4}$ ,  $P1[1 - (P1 + P2)]^{(i-1)} \sim 10^{-3}$

At LSR 5:  $P2[1 - (P1 + P2)]^{(i)} \sim 10^{-4}$ ,  $P1[1 - (P1 + P2)]^{(i)} \sim 10^{-3}$



— Default free path

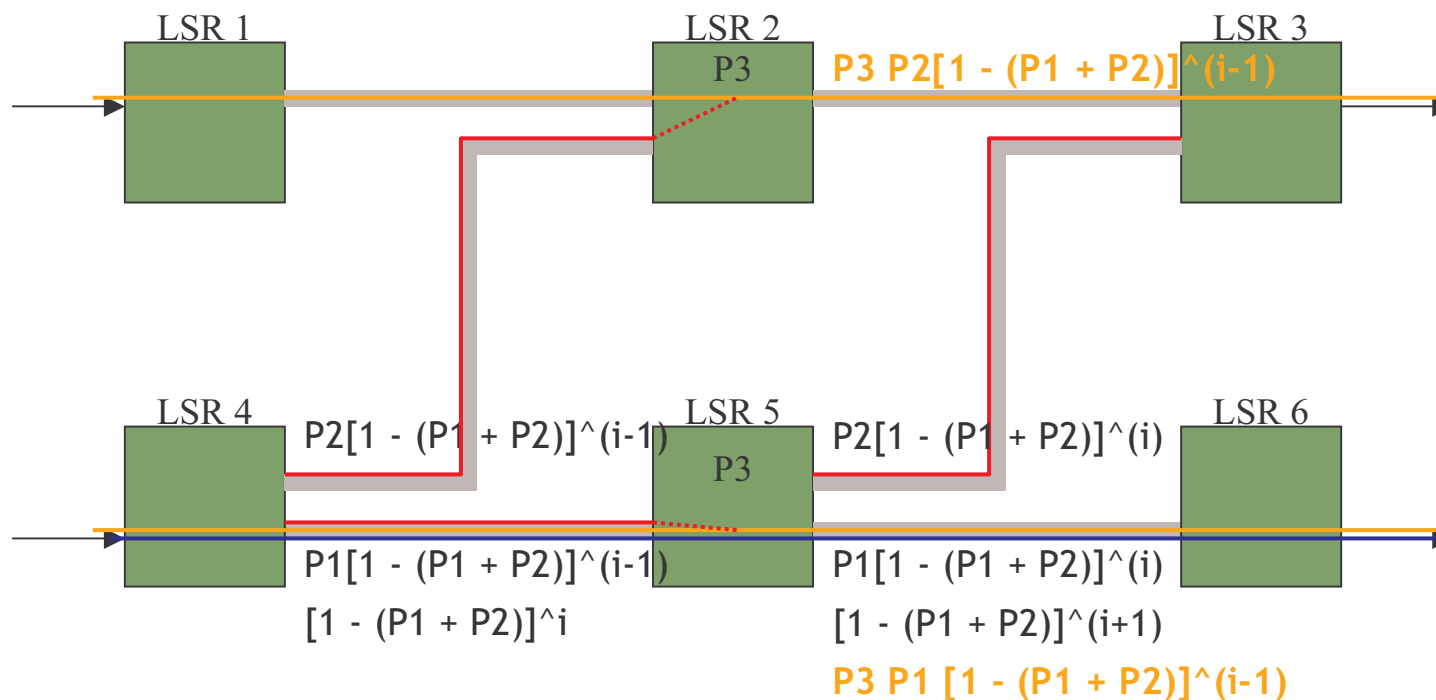
— Potentially mis-merge path

## Step 2: Error propagation/mis-merge probability

Example:  $i = 5$ ,  $P1 = 0.001$ ,  $P2 = 0.0001$ ,  $P3 = 0.01$  (~ 10k LSP)

At LSR 2:  $P3 P2 [1 - (P1 + P2)]^{(i-1)} \sim 10^{-2} \times 10^{-4} \sim 10^{-6}$  otherwise drop

At LSR 5:  $P3 P1 [1 - (P1 + P2)]^{(i-1)} \sim 10^{-2} \times 10^{-3} \sim 10^{-5}$  otherwise drop

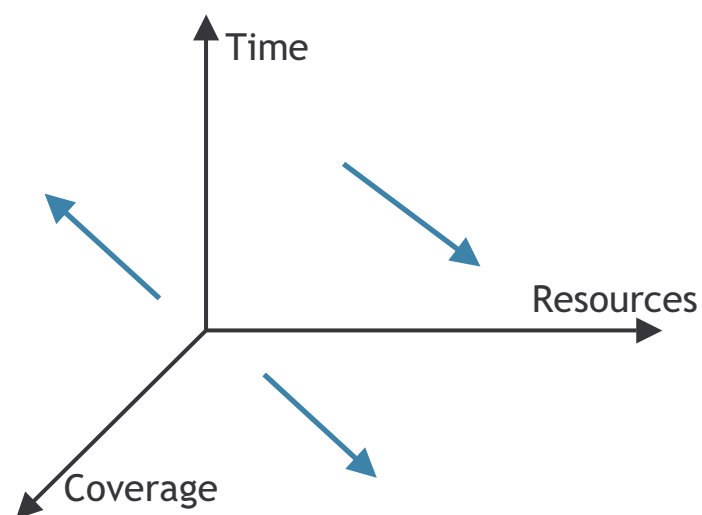


- Default free path
- Potentially mis-merge path

## Step 3: Tools

---

- 3 Critical phases
  - Detection
  - Notification
  - Correction
- Tools performance:  
detection  $\oplus$  notification  $\oplus$  correction
- 3 phases depending on 3 dimensions
  - Time
  - Resources (network, system and operational)
  - Coverage (connectivity/destination, traffic)



## Step 3: Tools - Detection

### Fault management - OAM Classes

	Intrusive (frame header)	Injection (OAM PDU)
Local-label		IP/MPLS: <b>LSP Ping</b> , <b>MPLS BFD</b> , Signaling Specific tools
Domain-wide label	PBT OAM (SA-DA check)	PBT: 802.1ag extensions PB/PBB (Bridged Ethernet): 802.1ag
Globally Unique (label or address)		IP: BFD, ICMP, Traceroute, etc.

## Step 3: Tools - Detection

---

### Intrusive

- Relies on actual data traffic (frame header)
- Con's: traffic dependent
- Pro's: fast detection (but comes at the expense of additional complexity)

### Injection of OAM-PDU

- Control-driven  $\neq$  off data-path
  - Example 1: BFD for detecting a data plane failure in the forwarding path of a MPLS LSP
  - Example 2: LSP-Ping for verifying the MPLS LSP data plane against the control plane
    - => **False positive**: in the event of a MPLS LSP failing to deliver data traffic, not always possible to detect the failure using the MPLS control plane
    - For instance the control plane of the MPLS LSP may be functional while the data plane may be mis-forwarding or dropping
- Con's: overhead
- Pro's: failure detection independent of activity on the data path, some level of genericity (per forwarding class)

## Step 3: Tools - Detection

---

### For detection of permanent errors

- OAM PDUs follow same data path => permanent errors can be detected but not corrected (potentially avoided)
- Detection
  - Tail-end either by reception of false OAM PDUs or non-reception of true OAM PDU
  - Head-end receives or not echoed OAM PDUs from tail-end

### To decrease detection time, increase OAM PDU frequency

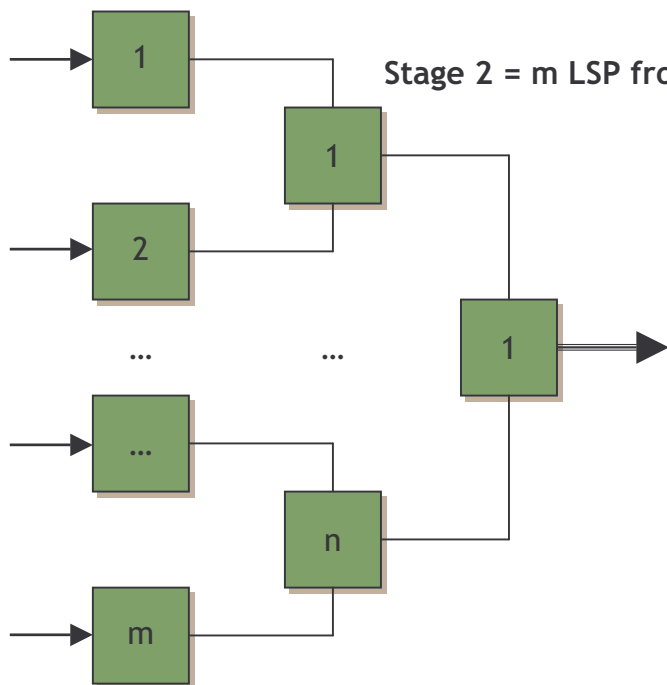
- Does not correct error (drop traffic  $\equiv$  avoidance mechanism)
- Consequences
  - Increase fractional resources used by OAM PDUs (result usually in limiting the max number of OAM PDUs per s.)
  - Increase number of false OAM PDUs received from mis-merged LSPs

# Step 3: Tools - Detection

## OAM PDU bandwidth consumption problem

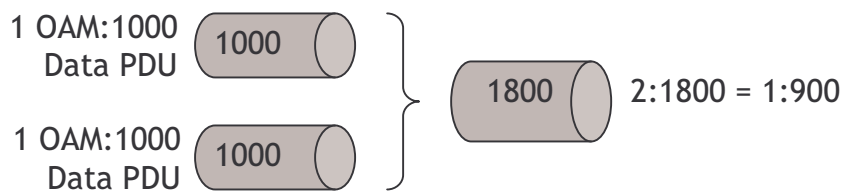
- Processing: LSP granularity (max number of LSP per interface wrt OAM PDU)

Stage 1 = m LSP from m nodes



Stage 2 = m LSP from n nodes,  $n < m$

After k stages = m LSP from p nodes,  $p \ll m$   
 Fractional increase of OAM bandwidth needs wrt to provisioned (link) capacity for m LSPs

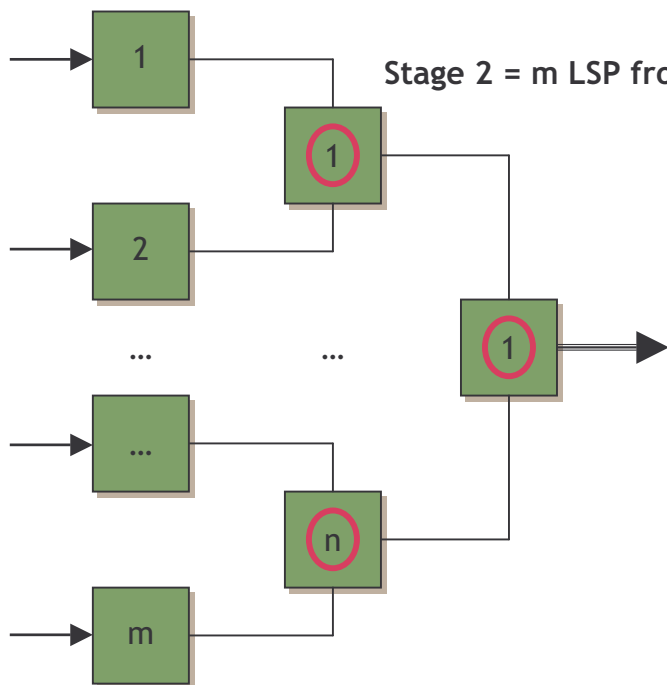


# Step 3: Tools - Detection

OAM filtering at merging/multiplexing points: results in “state maintenance” for intermediate OAM processing

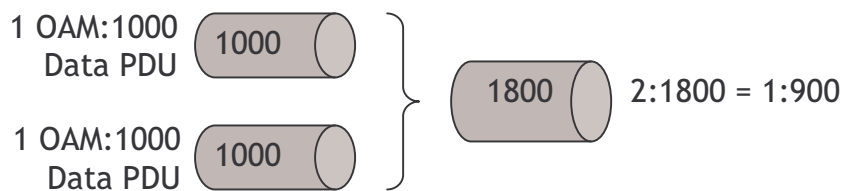
=> Bandwidth consumption vs state maintenance problem

Stage 1 = m LSP from m nodes



Stage 2 = m LSP from n nodes,  $n < m$

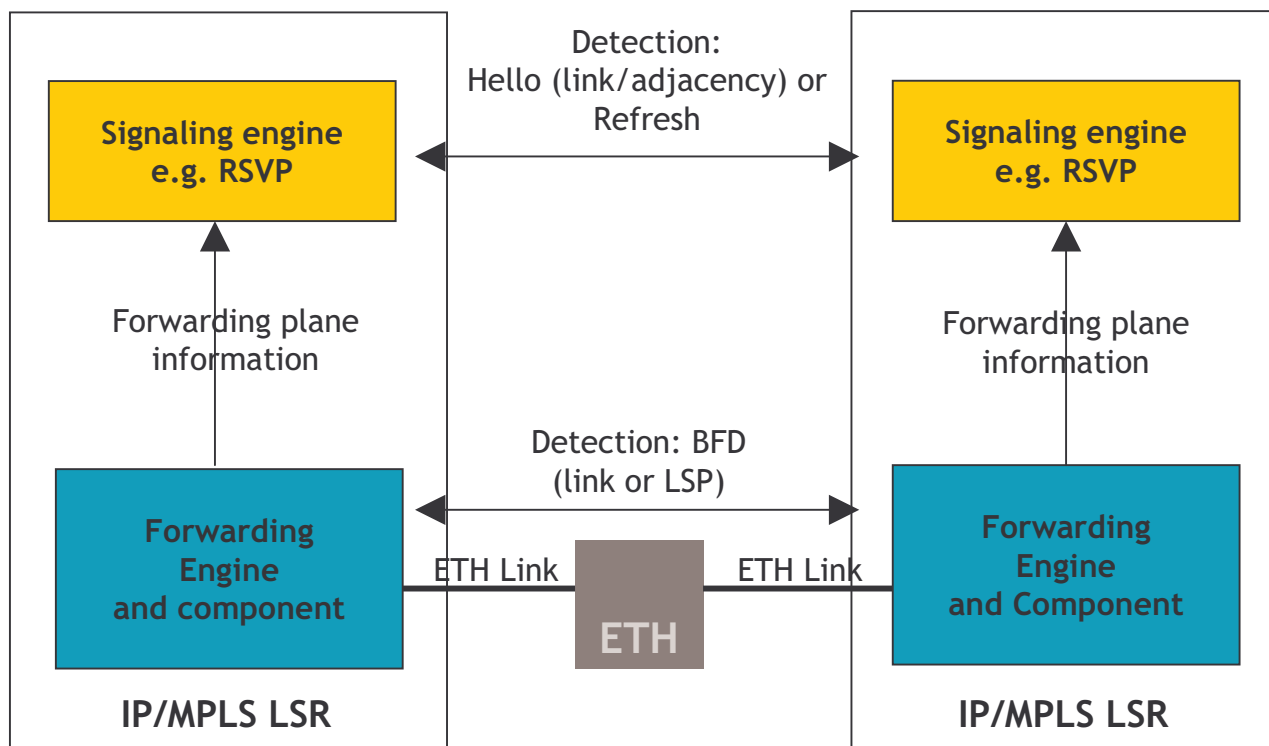
After k stages = m LSP from p nodes,  $p \lll m$   
 Fractional increase of OAM bandwidth needs wrt to provisioned (link) capacity for m LSPs



# Step 3: Tools - Time performance vs processing resources

Detection mechanisms: rules

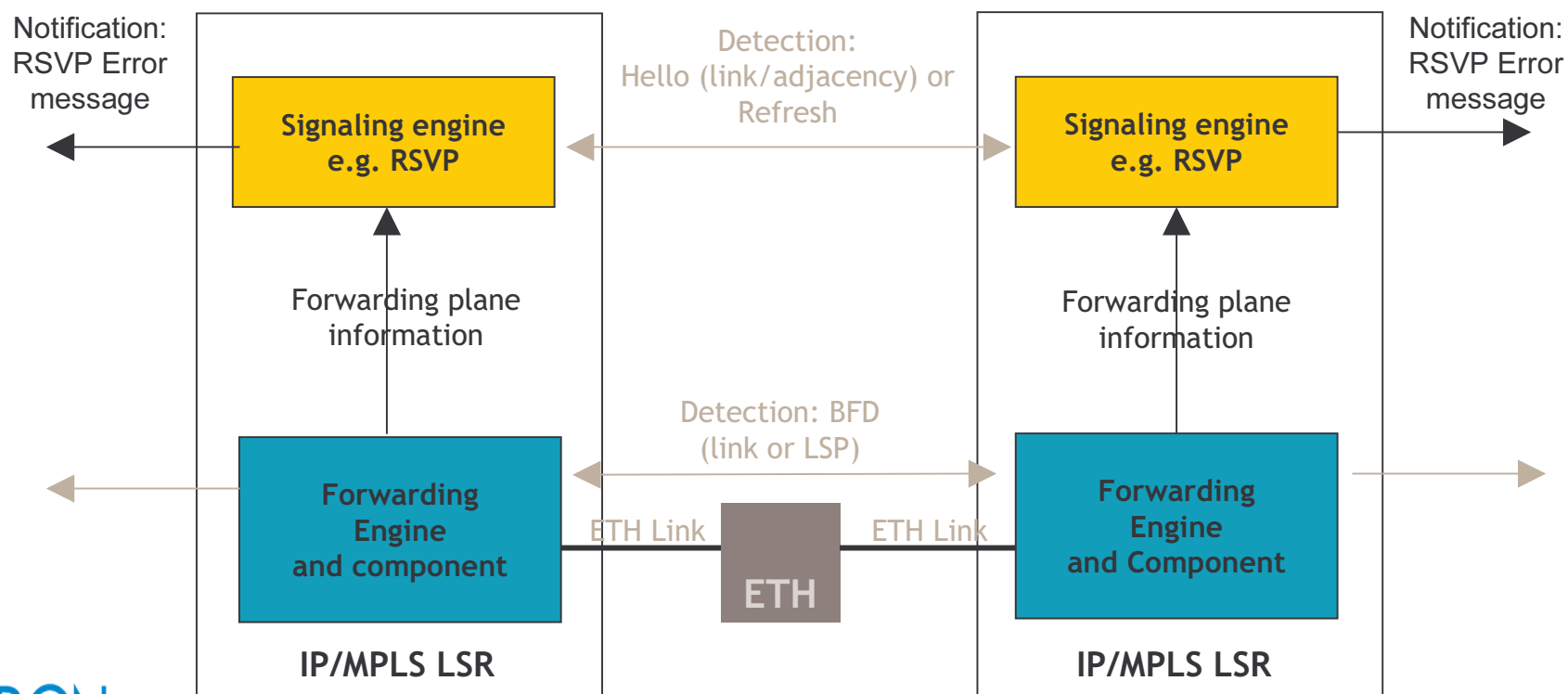
- Granularity (detected entity vs link capacity) ↓ => messaging/processing ↑
- Timing:
  - Intrusive: independent of frequency OAM PDU, closeness to the failure
  - Injection: detection time ↓ => messaging/processing ↑ (OAM PDU frequency ↑)



# Step 3: Tools - Time performance vs processing resources

Notification mechanisms: major dependency of processing

- Meshedness (connectionless forwarding)
- Granularity of failed entity vs Granularity of notification



# Conclusion from Analysis

---

## Detection of false positive

- Possibility to bring back specific system information such as to prevent/avoid undetectable errors
- Write/insertion consistency check
- Otherwise out-of-band correlation (control plane verification against the data plane)

## Several errors affect any label-switching technology

- outgoing interface selection
- label operation (e.g. incorrect read, head-end insertion, etc.)

## Mis-merge detection (error propagation) is the most concerning issue

- affects any technology supporting “merging” = local-

# Overview of Standardization activities in this domain

# IETF WG: OAM Tools

---

- PWE3 WG
  - PW OAM
  
- MPLS WG
  - MPLS OAM Framework: RFC 4378
  - MPLS OAM Requirements: RFC 4377
  - LSP Ping and Traceroute: RFC 4379
  
- BFD WG
  - BFD for MPLS LSP: draft-ietf-bfd-mpls-0x.txt
  - Developed in BFD WG at the IETF: draft-ietf-bfd-base-06.txt
  
- IPPM WG (Performance metrics)
  
- PSAMP WP (Packet sampling)

# BFD Working Group

---

Major enhancements from the initially proposed charter:

- Extend scope to Packet LSP
- Extend scope to P2MP packet LSP

BFD for single/multi-hop IP adjacencies

- on-demand/continuous CV: supported
- FDI/BDI: not supported
- loopback (LB): supported
- active monitoring: not in scope

BFD for MPLS LSPs

- on-demand/continuous CV: supported
- FDI/BDI: not supported
- loopback (LB): supported
- active monitoring: not in scope

# IETF IP/MPLS OAM Tools

OAM Feature	Defect Indication	Connectivity Verification	Continuity Checking	Path Trace	Loopback	Performance Monitoring	Notes
LSP Ping		x					
LSP Traceroute				x			
LSR Self Test		x		x			
BFD for MPLS LSPs	Detection by poll	x	x		x		BPD pushed as a faster and more scalable failure detection mechanism.
BFD/VCCV for PW		x	x		x		
PW Status Signaling	x						Well suited for PW defect indication.

# Current Status: Comparison

	ITU-T Transport MPLS (SG15)		IETF MPLS	
	Path (tunnel trail)	Section (interface trail)	MPLS PSN profile used by Eth PW	Link
Forwarding Plane	No PHP No merge	Section (G.805)	PHP/non-PHP Label merging (FRR)	IP-over-X interface: PoS, Ethernet
Control Plane	Static	Static	MIB MPLS RSVP-TE signaling (RFC 3209) GMPLS RSVP-TE signaling (RFC 3473)	Link configuration (MIB)
OAM	G.8114	G.8114	BFD (draft-ietf-bfd-base-06) LSP Ping (RFC 4379)	BFD, ICMP-Ping
Resiliency	G.8131 & G.8132 (linear & ring)	G.8131	End-to-end Re-routing (RFC 3209) - make before break Fast re-routing - FRR (RFC 4090)	Link protection via FRR

# Acknowledgements

---

This work was carried out within the framework of the IWT TIGER project sponsored by the Flemish government institute for Innovation through Science and Technology in Flanders (IWT)

The background is a deep blue color with a fine, light-colored grid pattern. Overlaid on this grid are several bright, glowing light streaks and curved lines that create a sense of motion and depth. The text is centered in the middle of the image.

Thanks !

[www.alcatel-lucent.com](http://www.alcatel-lucent.com)