White Rabbit
A TSN-compatible implementation ?

Maciej Lipiński

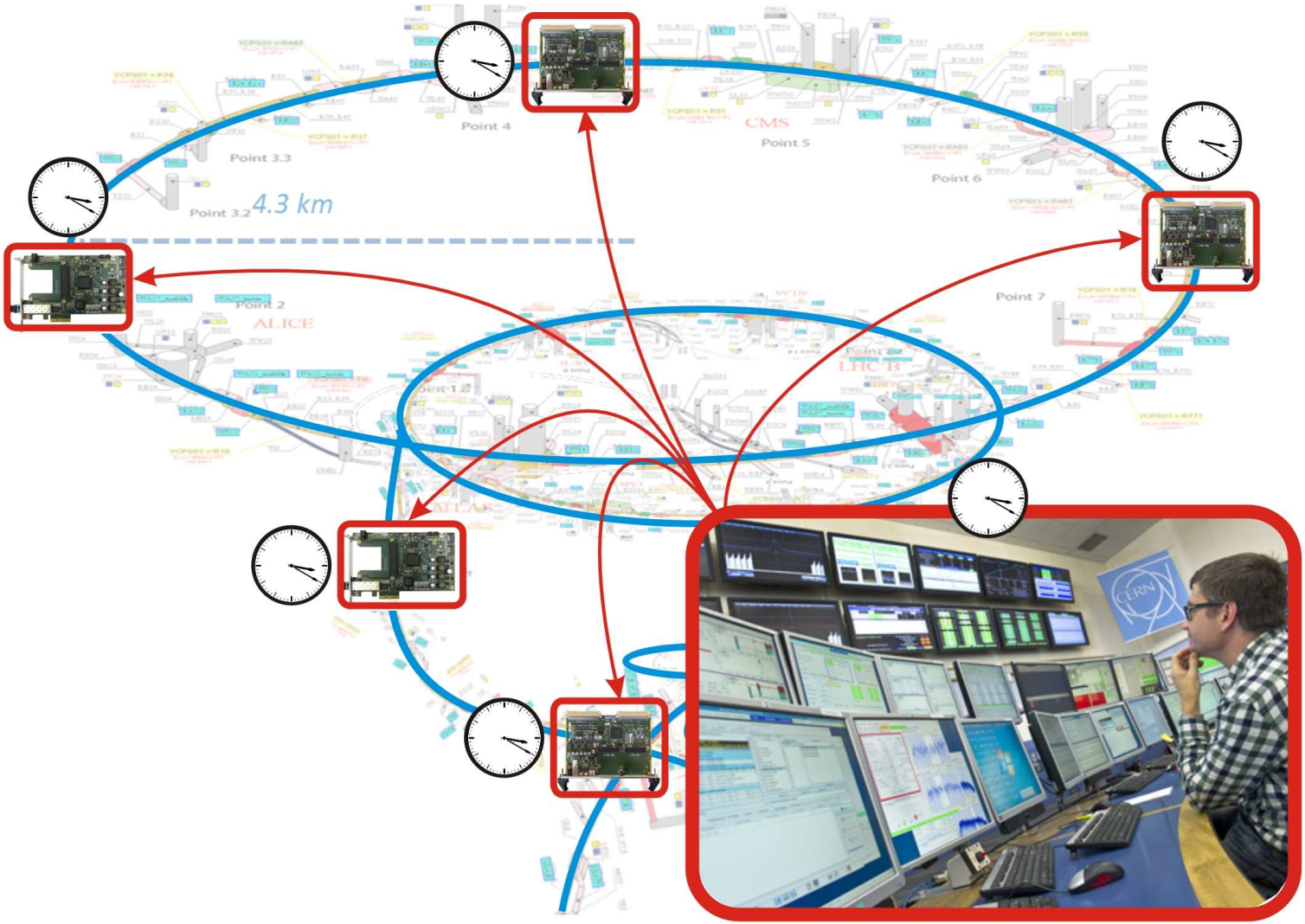
Hardware and Timing Section @ CERN
Institute of Electronic Systems @ Warsaw University of Technology

17 July 2013
IEEE Plenary Meeting Genève

Outline

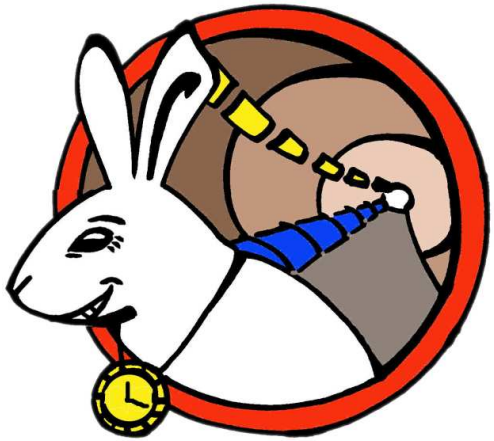
Outline

What is White Rabbit?



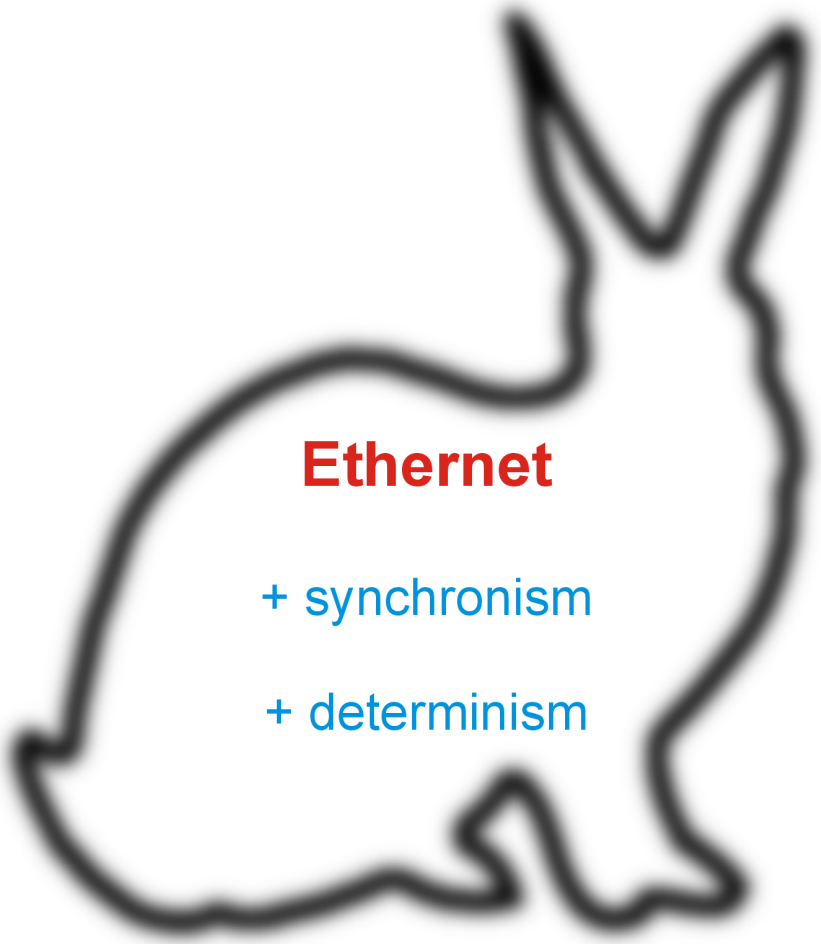
What is White Rabbit?

- Renovation of accelerator's control and timing
- Based on well-known technologies
- Open Hardware and Open Software
- International collaboration



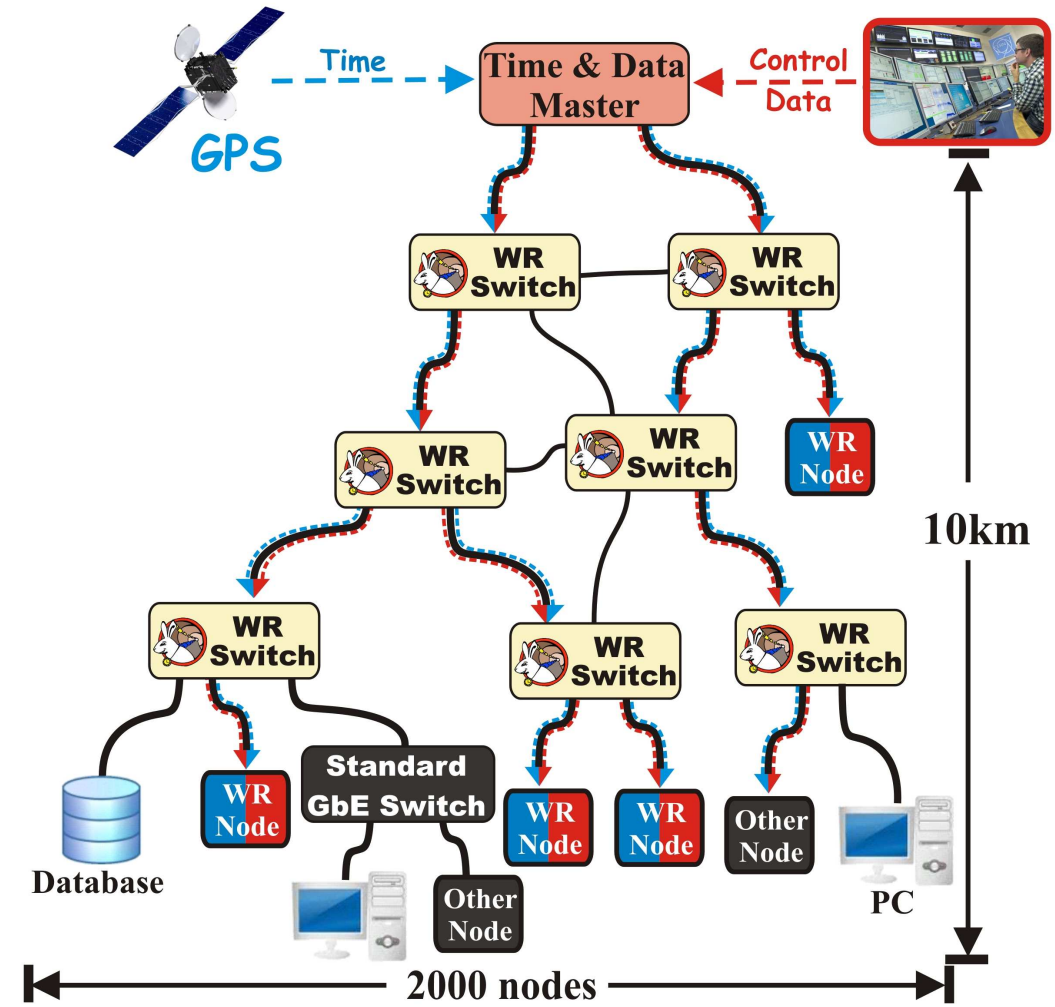
White Rabbit features

- standard-compatible
- sub-ns accuracy
- tens-ps precision
- upper-bound low-latency
- white-box simulation & analysis
- high reliability
- tens-km span
- thousands-nodes systems



White Rabbit Network – Ethernet-based

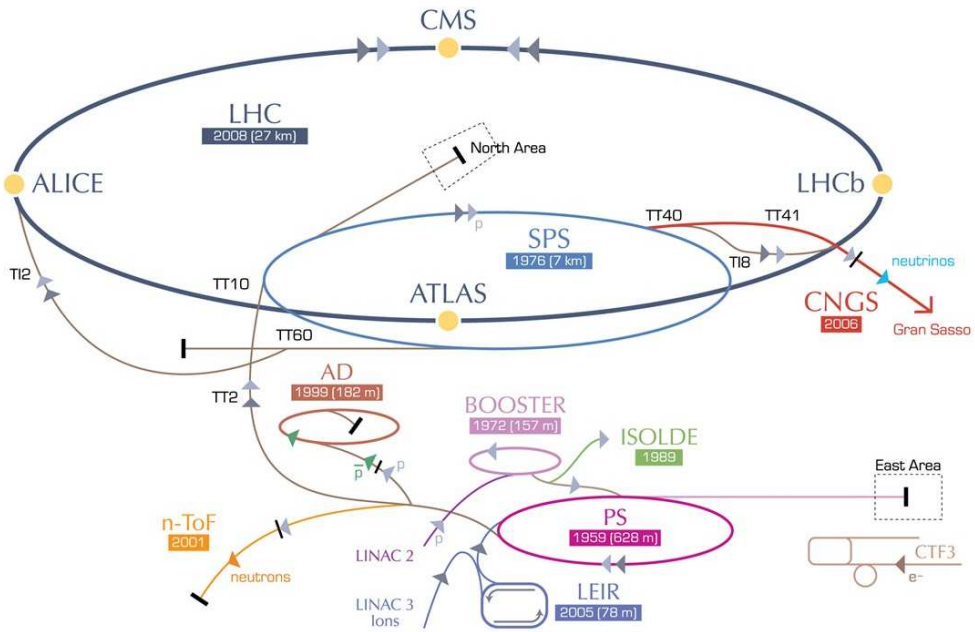
- High accuracy/precision synchronization
- Deterministic, reliable and low-latency Control Data delivery



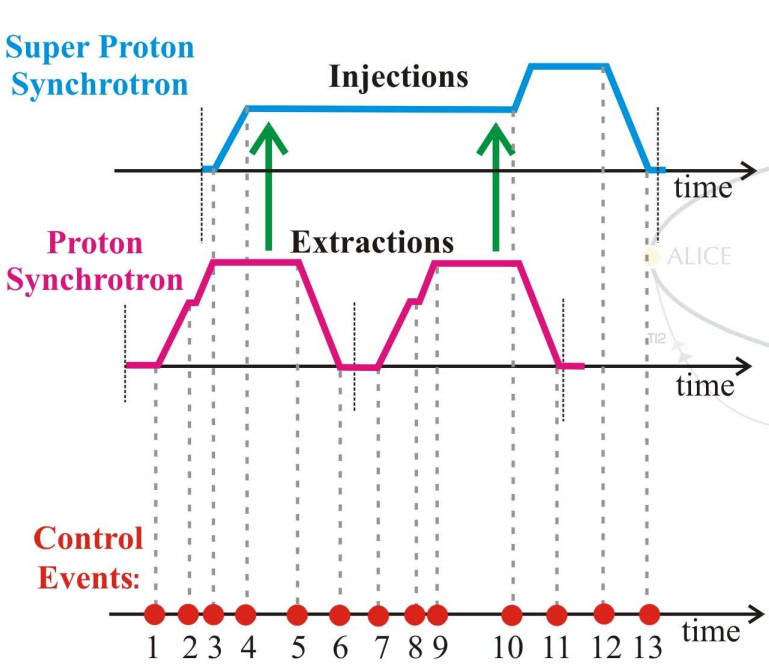
Outline

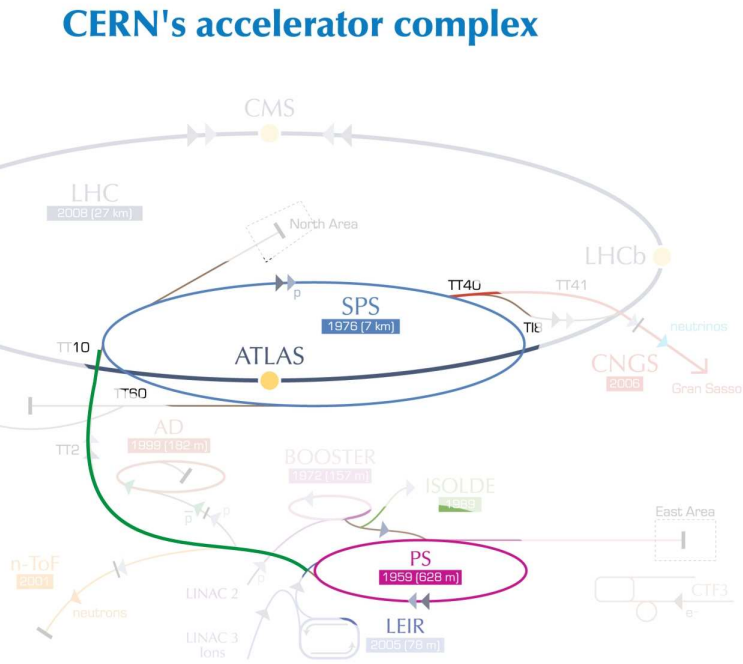
CERN Control and Timing System

- 6 accelerators including LHC: 27km
- A huge real-time distributed system
- Thousands of devices



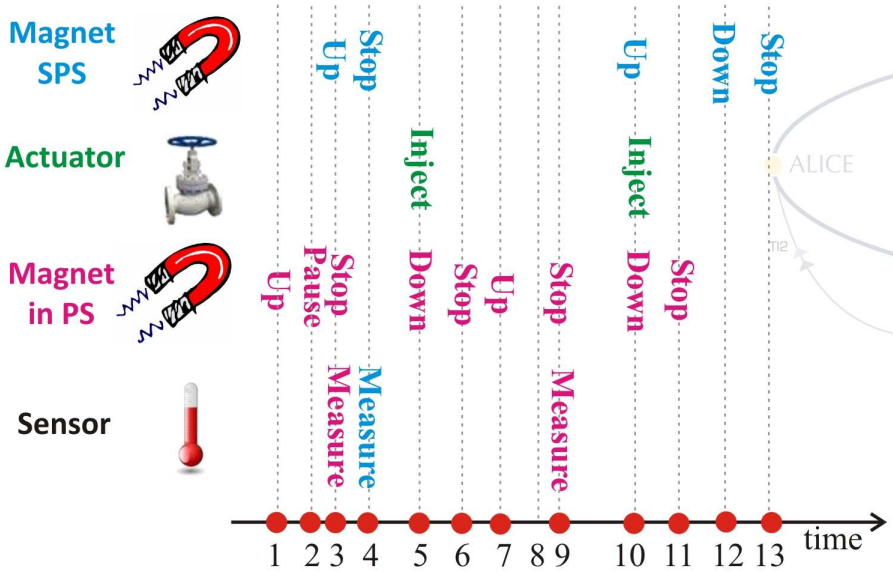
CERN Control System – event distribution (1)

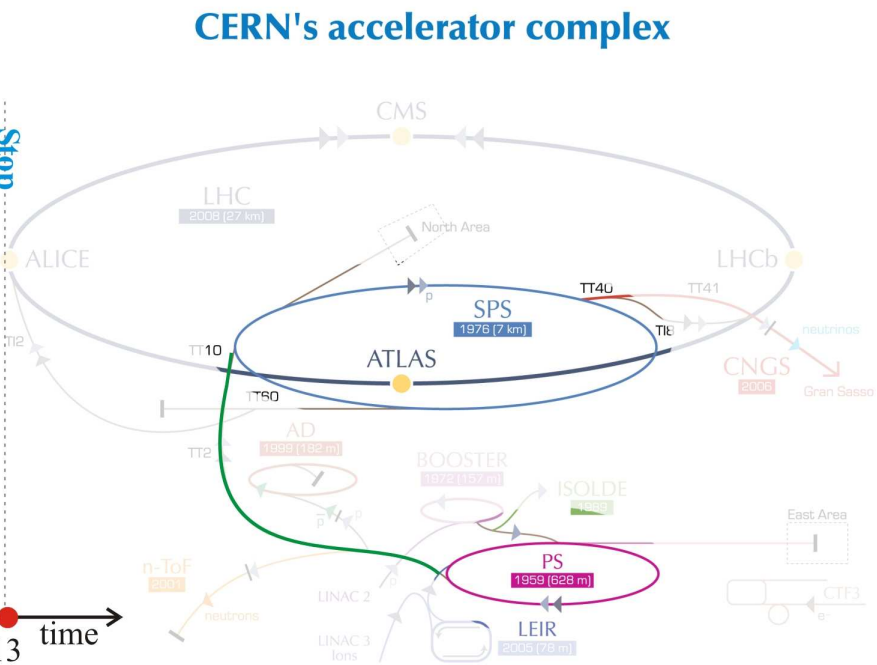




- **Events** – messages which trigger actions
- Each event is identified by an **ID**

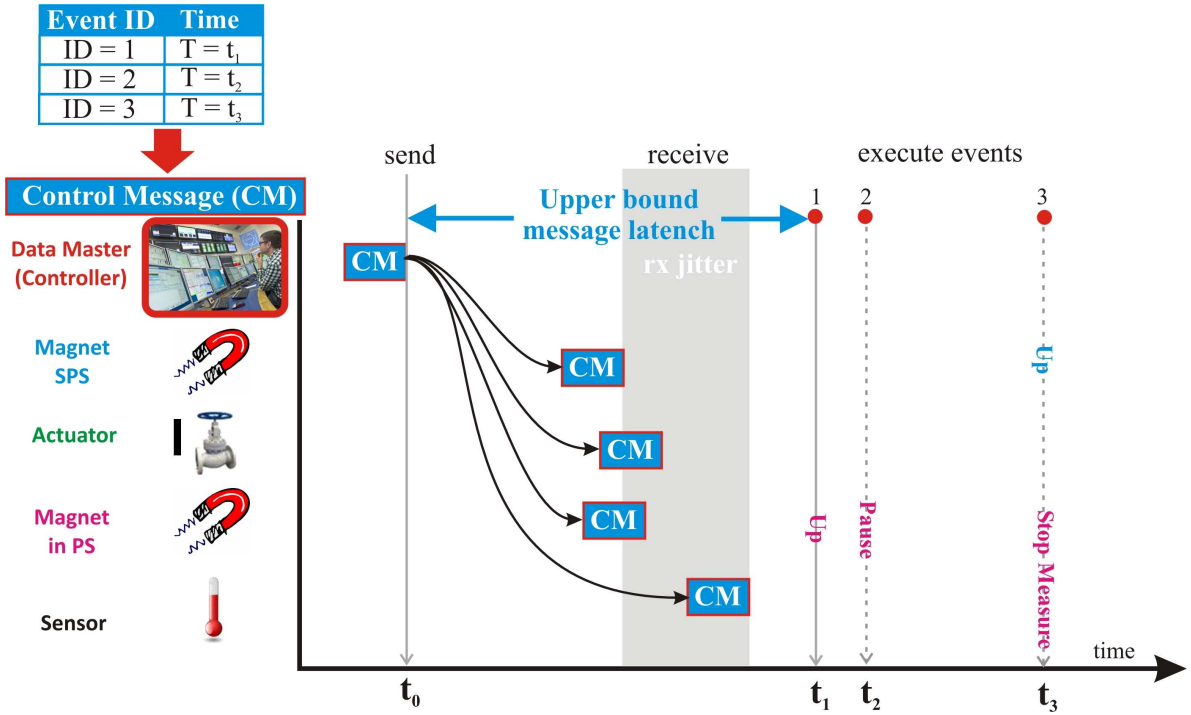
CERN Control System – event distribution (2)





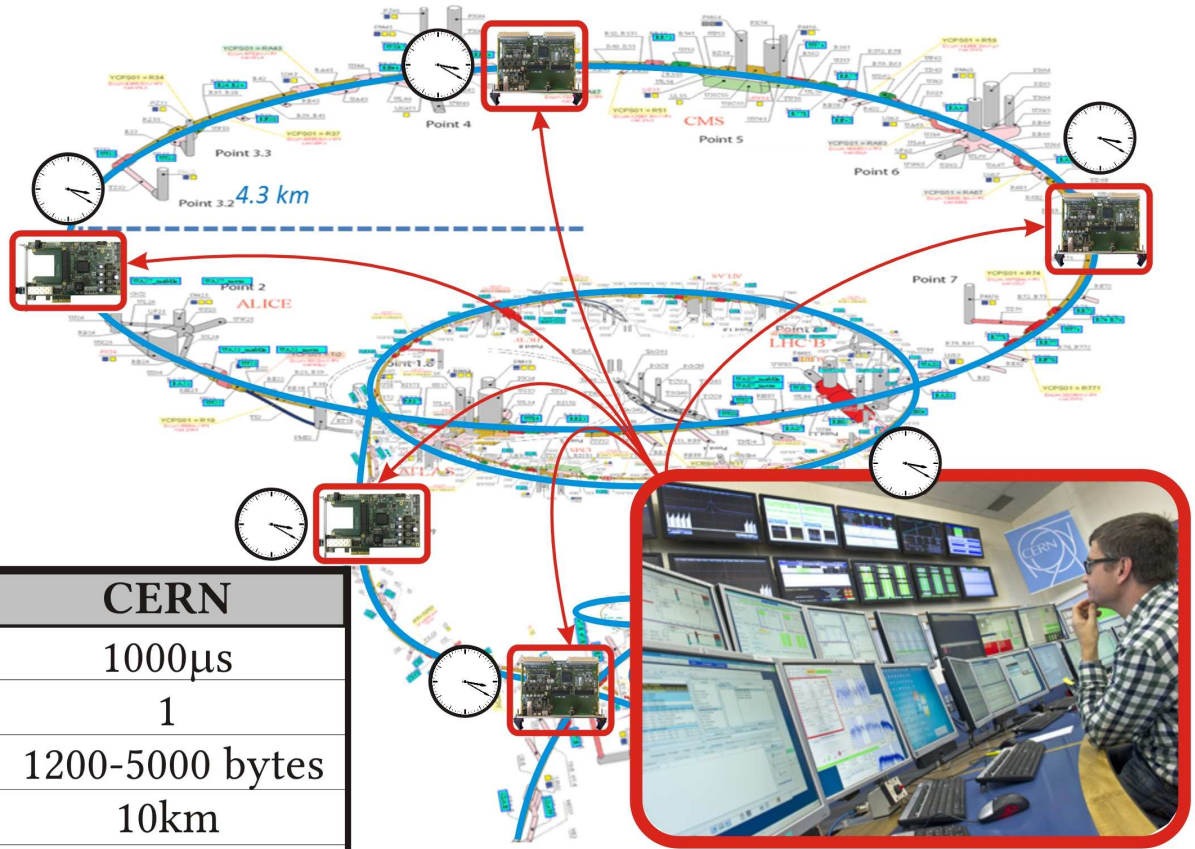
- Devices are subscribed to events
- Each device ”knows” what to do on a particular event

CERN Control System – event distribution (3)



- Each event (ID) has a trigger time associated
- A set of events is sent as a single **Control Message (CM)**
- CM is broadcast to all the end devices (nodes)
- CM is sent in advance (**upper-bound message latency**)

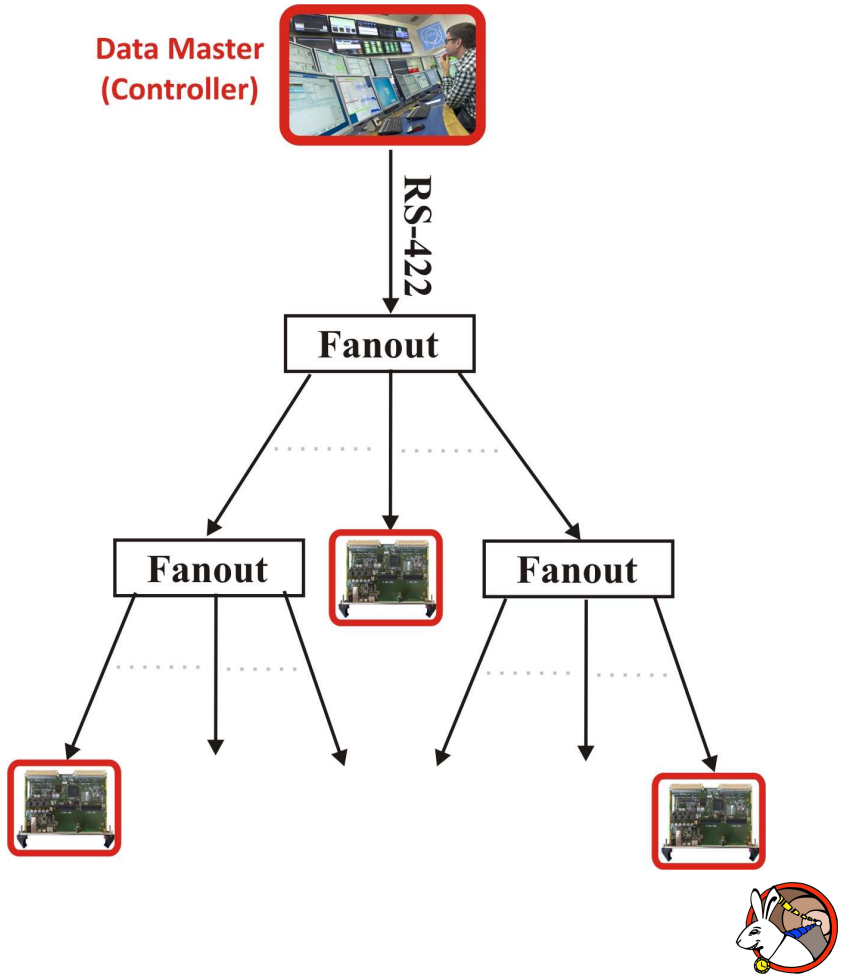
CERN Control & Timing Network – requirements



Requirement	CERN
Max latency	1000µs
CMs lost per year	1
CM size	1200-5000 bytes
Network span	10km
Accuracy	up to 1ns

Current system: General Machine Timing

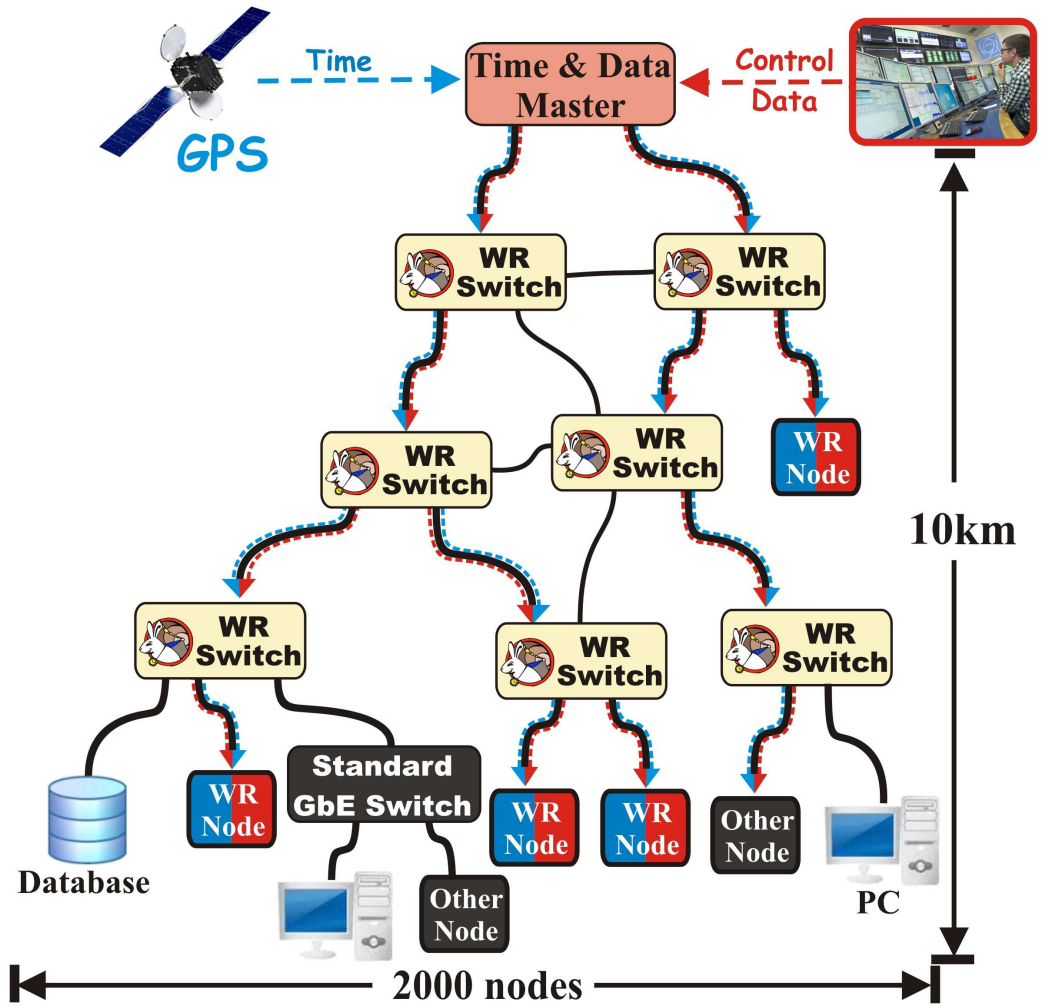
- Based on RS-422, low speed (500kbps)
- Unidirectional communication (controller$->$end stations)
- Separate network required for end station$->$controller communication
- Custom design, complicated maintenance



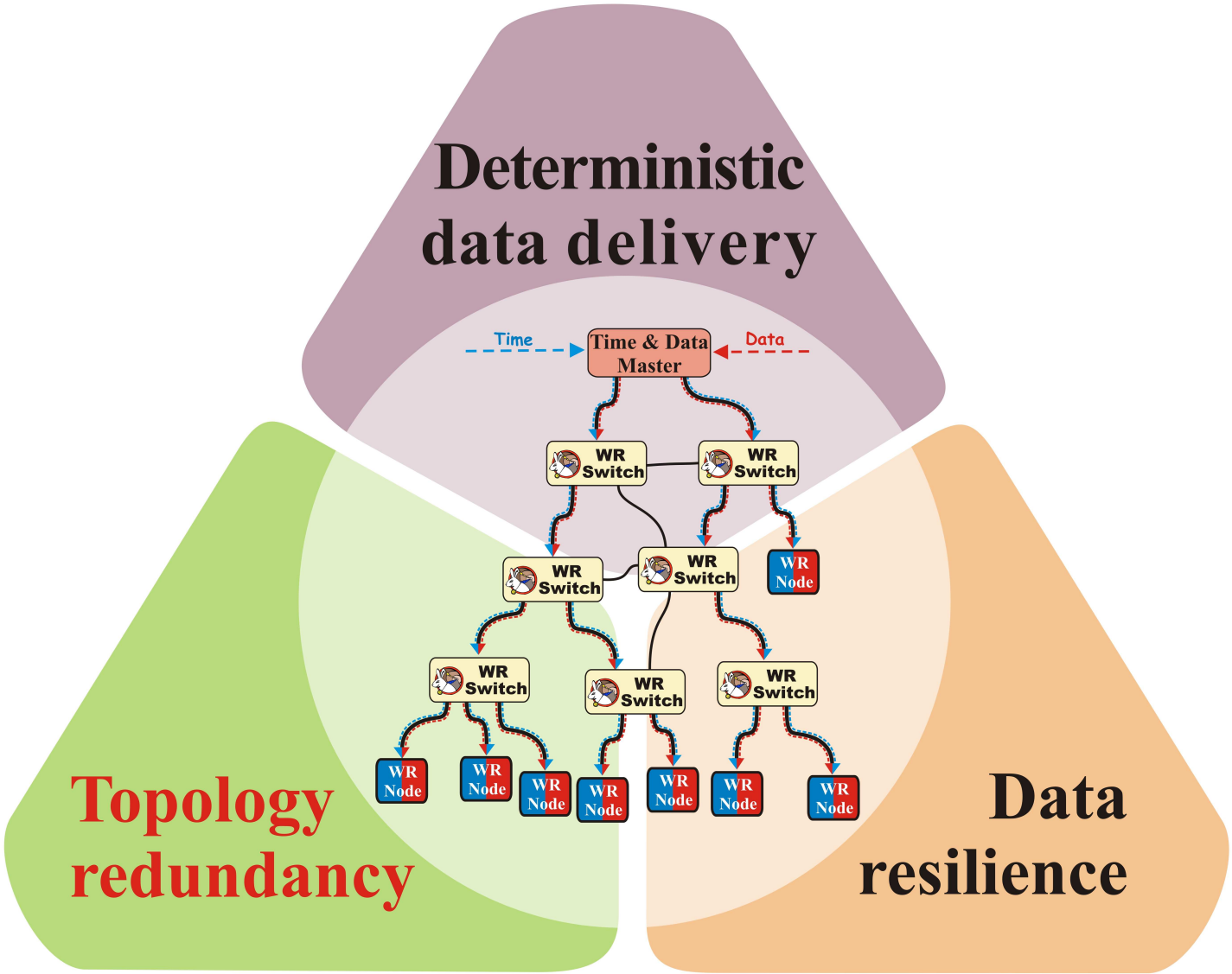
Outline

White Rabbit Network – Ethernet-based

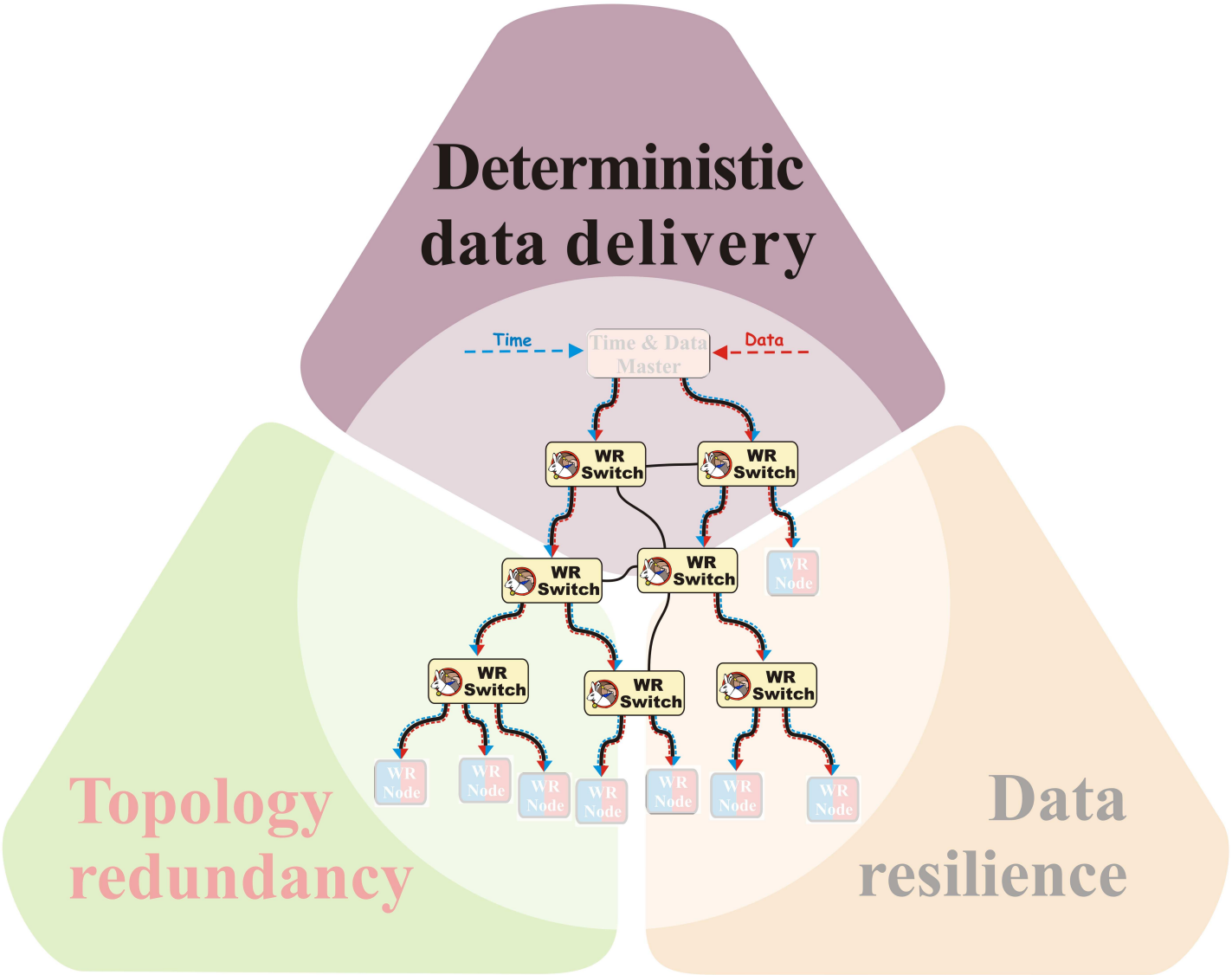
- High accuracy/precision synchronization
- Deterministic, reliable and low-latency Control Data delivery



Data Distribution in a White Rabbit Network

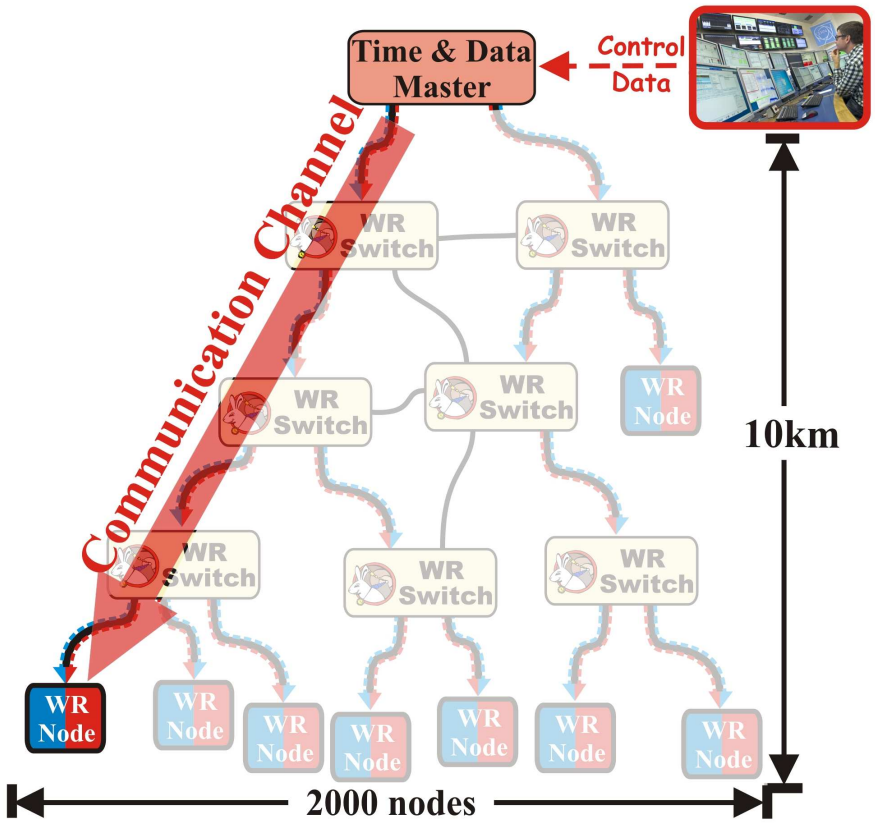


Determinism and Latency (Switch)



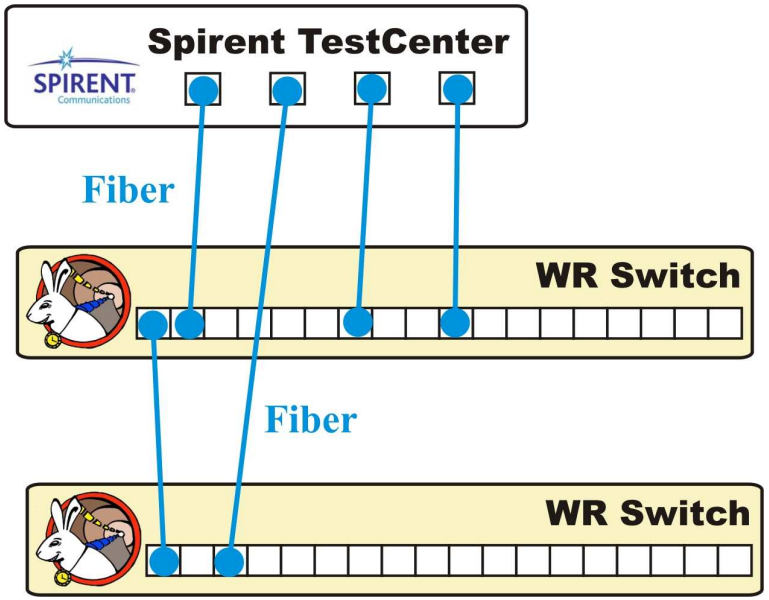
High Priority

- Types of data distinguished by 802.1Q tag:
  - **High Priority** (strict priority)
  - Standard Data (Best Effort)
- **High Priority** characteristics:
  - Broadcast/Multicast
  - Low-latency
  - Deterministic
  - Uni-directional
  - Re-transmission excluded
- Failure of **High Priority**:
  - Medium imperfection
  - Network element failure
  - Exceeded latency

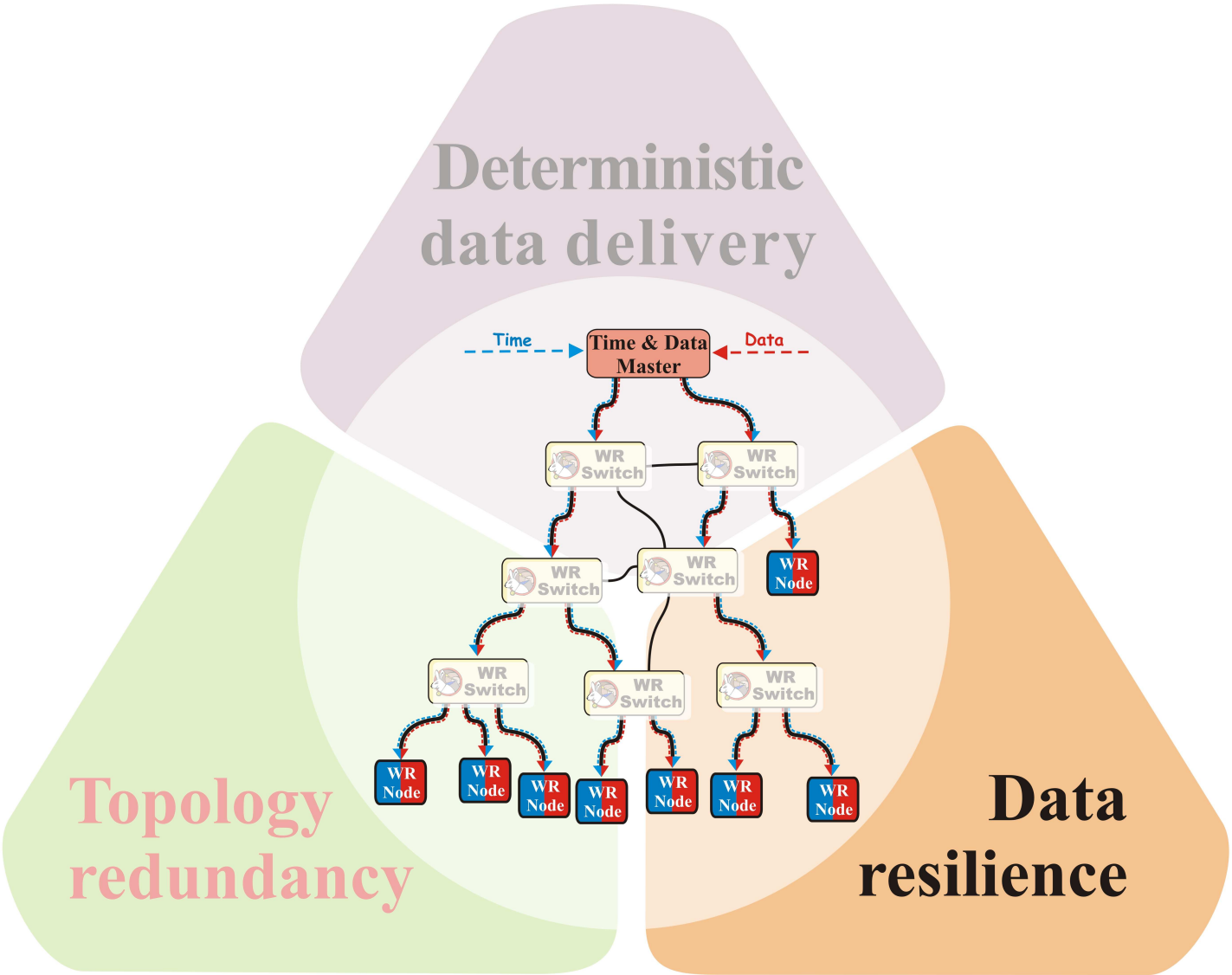


Determinism and Latency

- Deterministic Latency of High Priority
  - By design: $<$ 10us
    (single source of High Priority)
  - All size of frames
  - All rates
  - Regardless of Best Effort traffic
- Preliminary tests: $\approx$ 3us

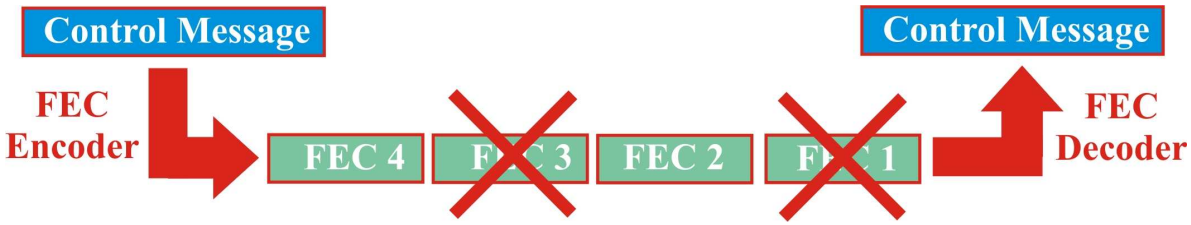


Data Resilience (Node)



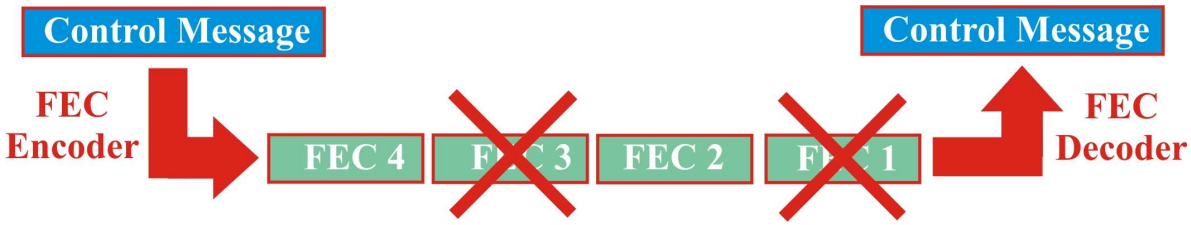
Data Redundancy

- **Forward Error Correction (FEC)** – transparent layer:
  - One message encoded into N Ethernet frames
  - Recovery of message from any M (M$<$N) frames



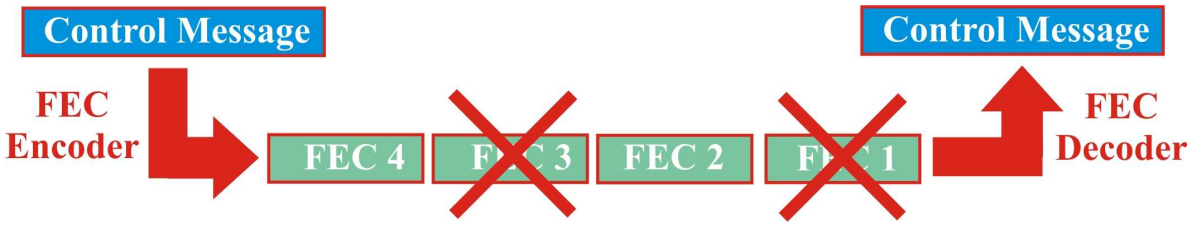
Data Redundancy

- **Forward Error Correction (FEC)** – transparent layer:
  - One message encoded into N Ethernet frames
  - Recovery of message from any M (M$<$N) frames
- FEC can prevent data loss due to:



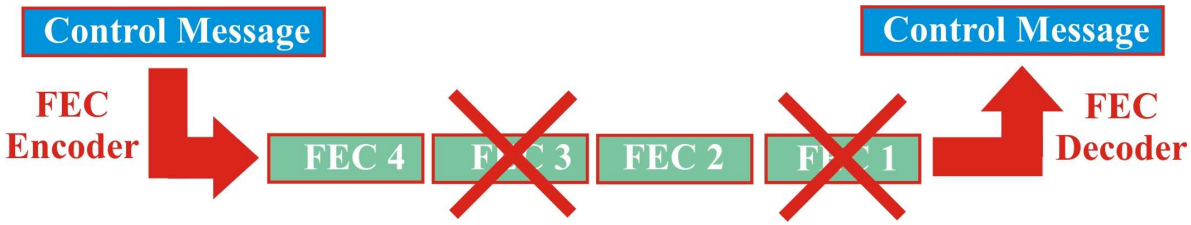
Data Redundancy

- **Forward Error Correction (FEC)** – transparent layer:
  - One message encoded into N Ethernet frames
  - Recovery of message from any M (M$<$N) frames
- FEC can prevent data loss due to:
  - **bit error**

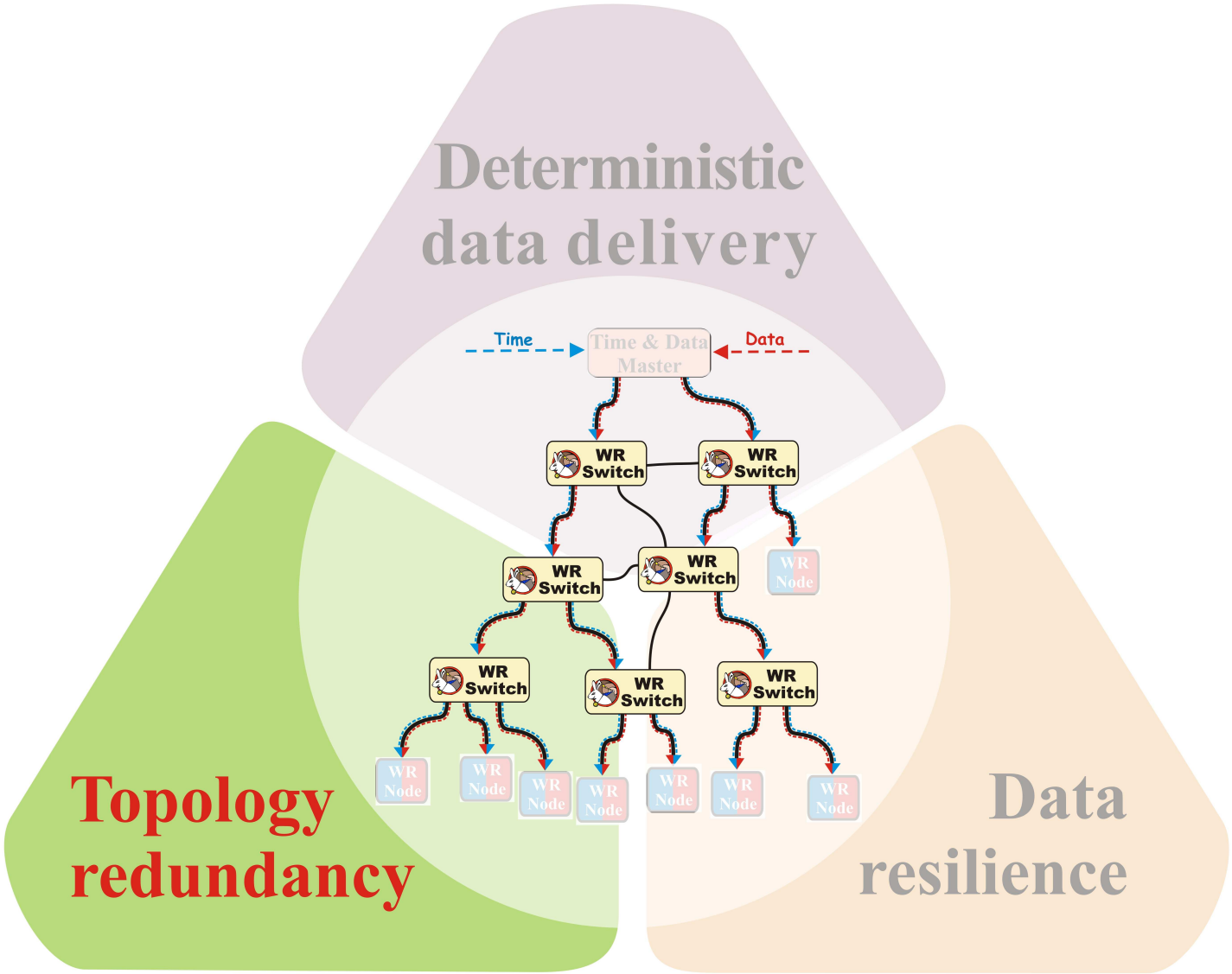


Data Redundancy

- **Forward Error Correction (FEC)** – transparent layer:
  - One message encoded into N Ethernet frames
  - Recovery of message from any M (M$<$N) frames
- FEC can prevent data loss due to:
  - **bit error**
  - **network reconfiguration**



Topology Redundancy (Switch)



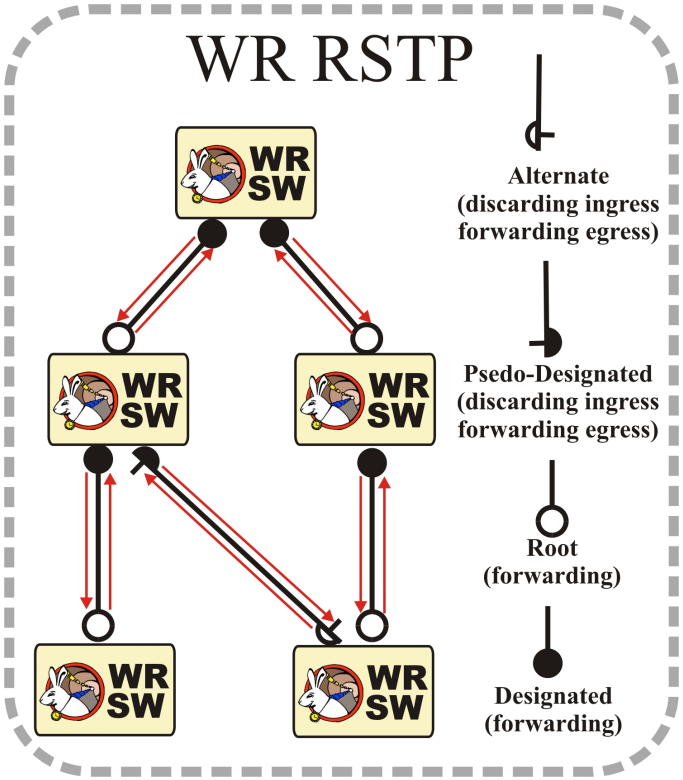
Topology Redundancy (Switch)

- Ideas:
  - Enhanced Link Aggregation Control Protocol (eLACP)
  - WR Rapid Spanning Tree Protocol (WR RSTP)
  - WR Shortest Path Bridging (WR SPB)
- Seamless redundancy = FEC + WR RSTP/SPB/eLACP
- Redundant data received in end stations
- Take advantage of broadcast/multicast characteristic of Control Data traffic (within VLAN)

Outline

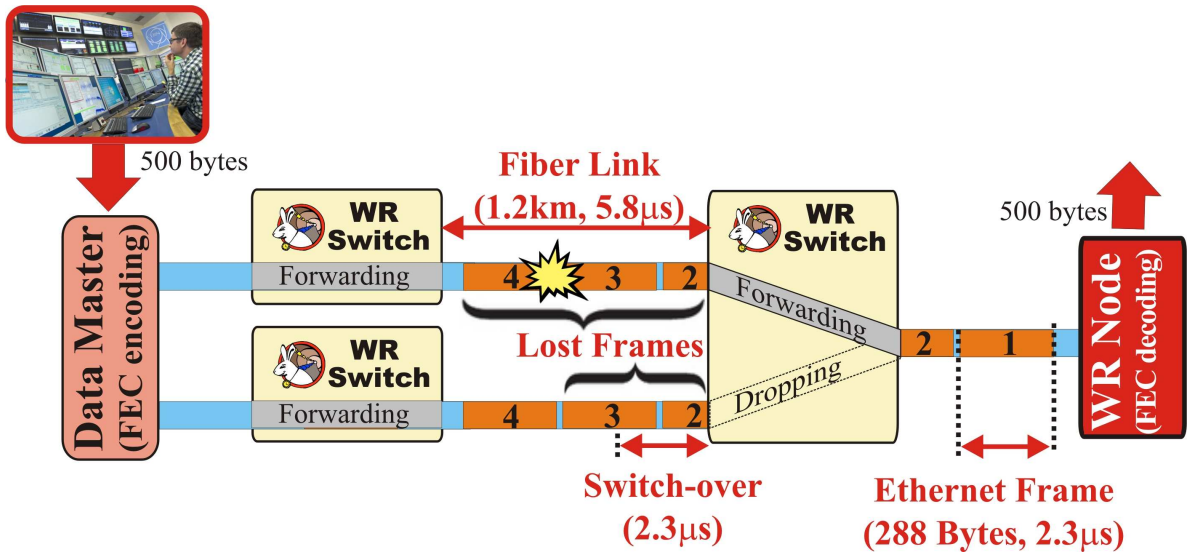
Topology Redundancy: WR RSTP

- Speed up RSTP – max 2 frames lost on re-configuration
- H/W switch-over to the backup link
- RSTP's a priori information (alternate/backup) used
- Limited number of allowed topologies
- Drop only on reception – within VLAN

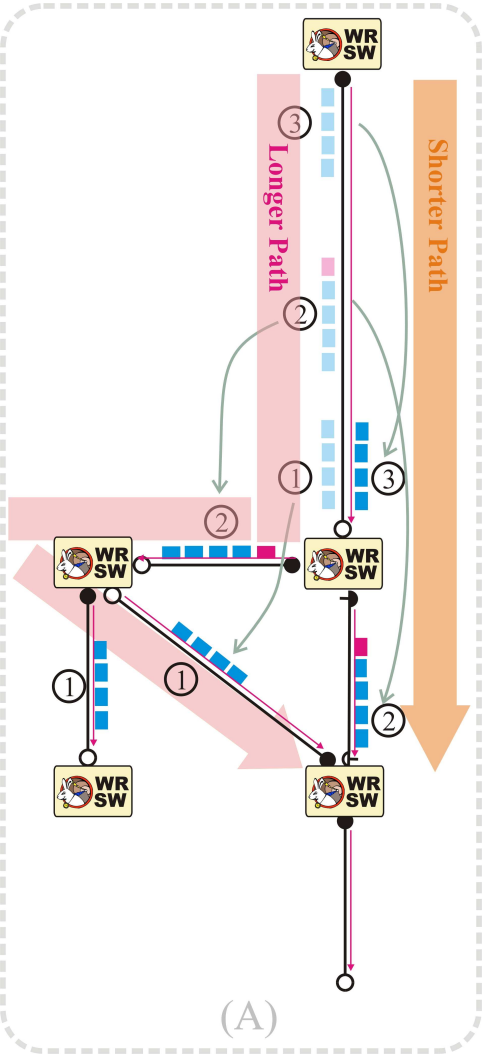


Seamless Redundancy: WR RSTP + FEC

- Seamless redundancy = WR RSTP+FEC <=> max 2 frames lost on reconfiguration
- 500 bytes message (288 byte FEC) – max re-conf ≈**2.3us**
- A priori backup configuration used for hardware switch-over – **broadcast/multicast** traffic (within VLAN)

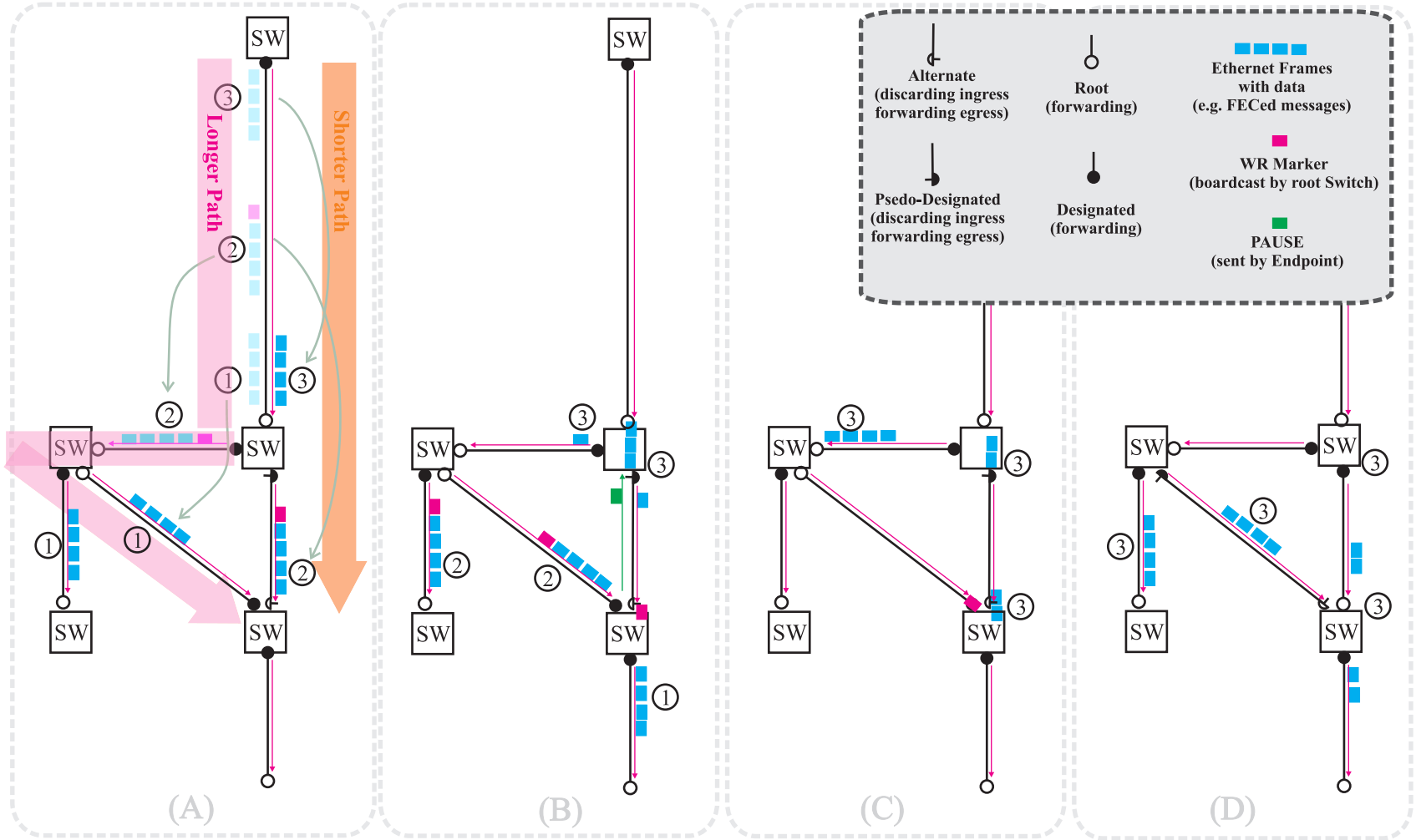


Semi-automatic re-configuration



- Adding new link/switch can cause dangerous re-configuration
- Any re-configuration not foreseen by stable-state BPDU exchange shall be semi-automatic
  - Run RSTP with some ”simulation” flag
  - Re-configuration is virtual
  - Re-configuration is reported to management for ack
  - New configuration (known in entire network) might be time-triggered
  - Done in no-Control-Data windows
- **How to do it in a standard-compatible way, if possible ?**

WR RSTP: adding new network element



WR RSTP: adding new network element

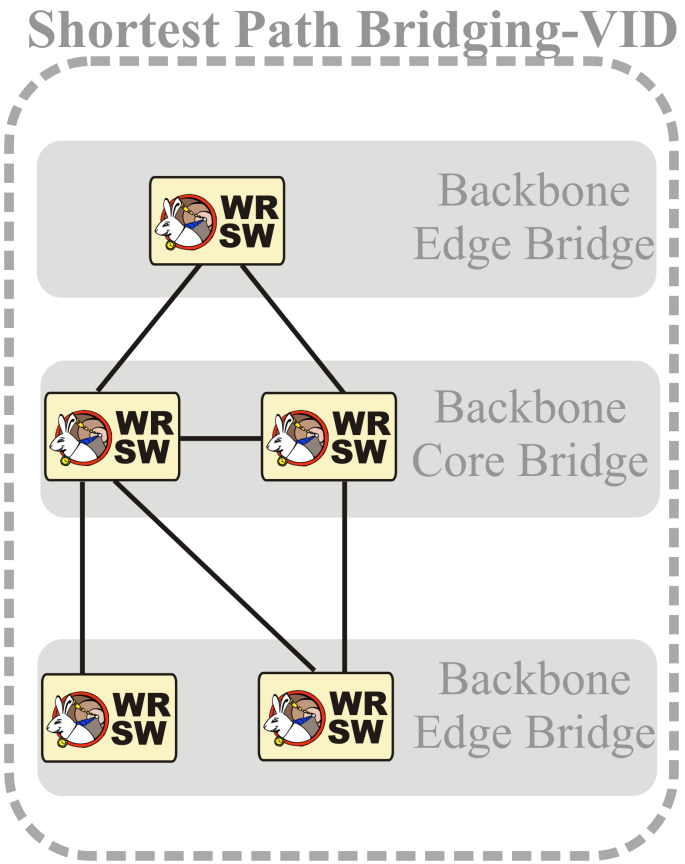
WR Marker

- Sent by Root Switch
- Forwarded by switches as other High Priority traffic
- Treated as BPDU for timestamping
- Used to:
  - measure real latency from Root Switch
  - trigger safe reconfiguration

Outline

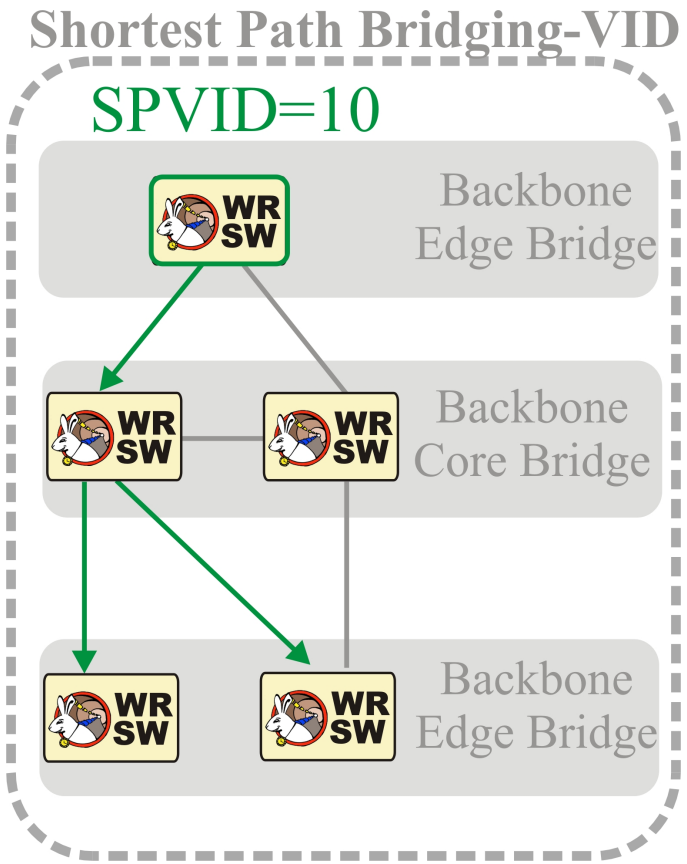
Shortest Path Bridging (SPB)

- SPB studies for WR
- Shortest Path Bridging – VID
  - Better fitted for existing development
  - Less overhead
  - Client isolation not required



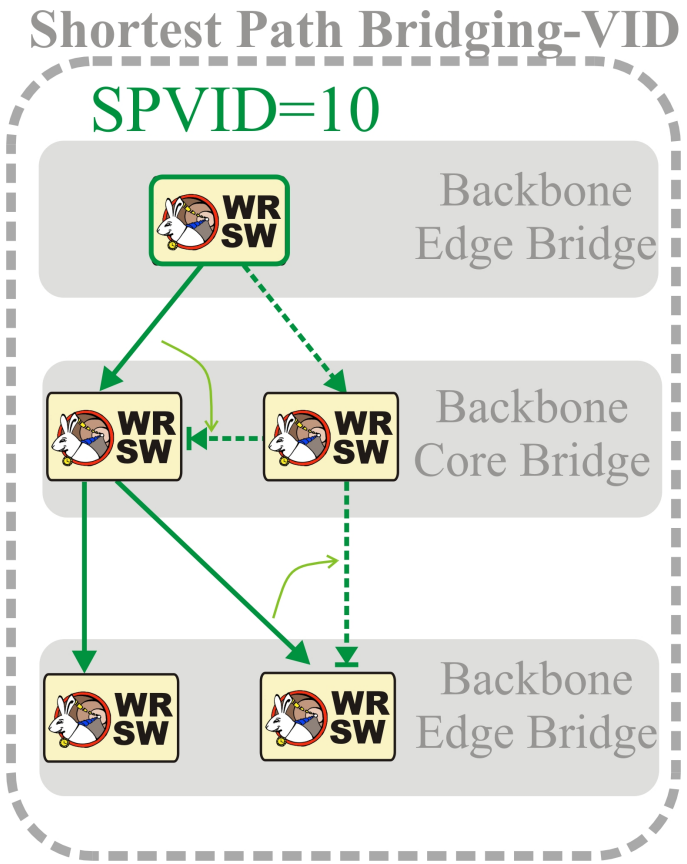
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree



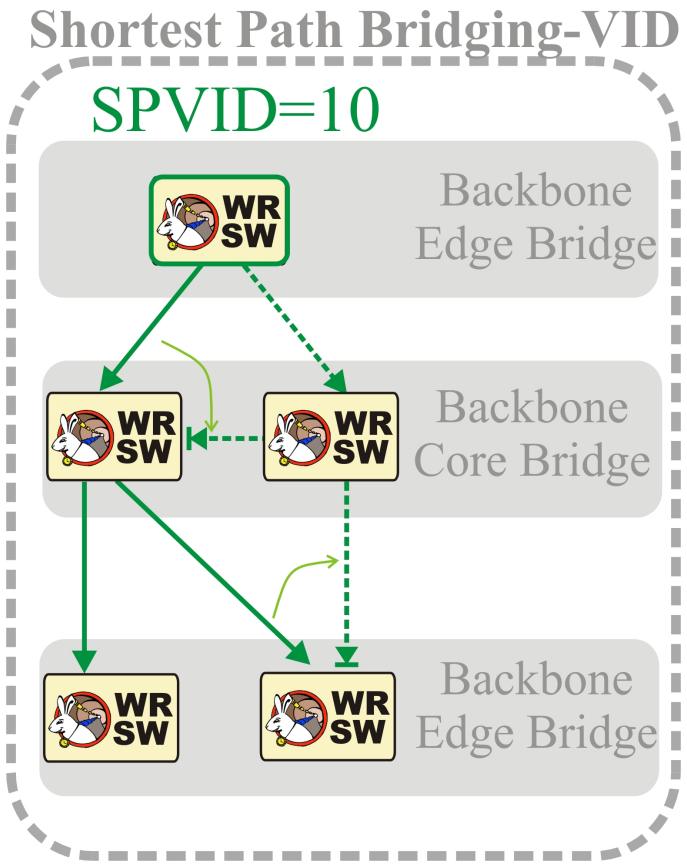
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree
- Backup tree – ports blocking on reception



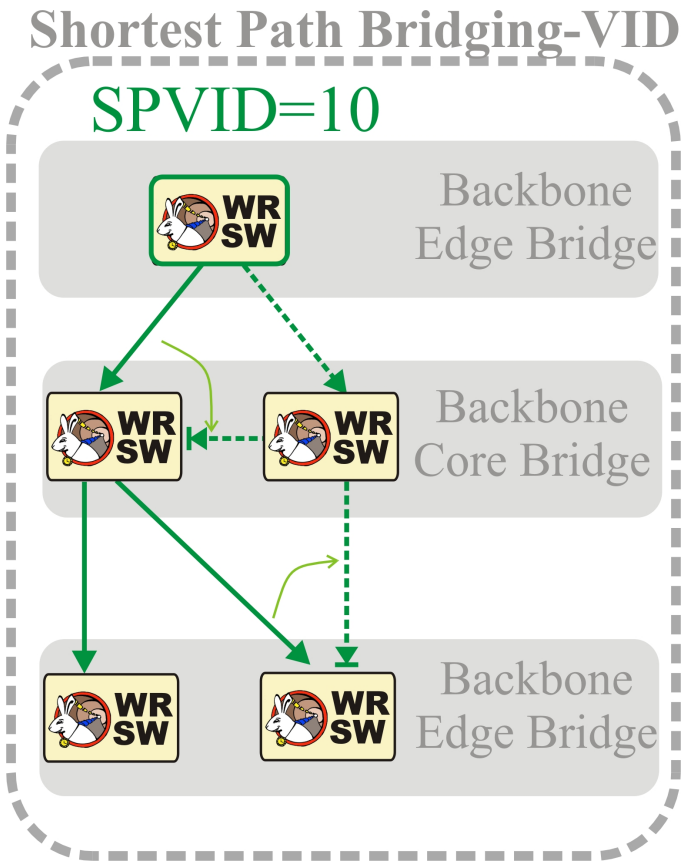
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree
- Backup tree – ports blocking on reception
- Single port forwarding from source



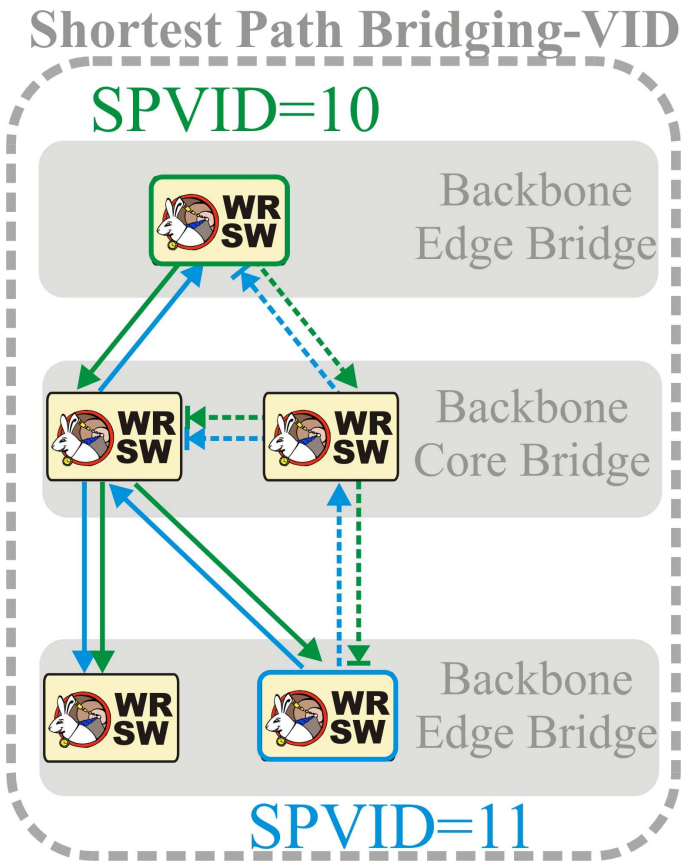
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree
- Backup tree – ports blocking on reception
- Single port forwarding from source
- H/W switch-over to path equally or more distant to the root



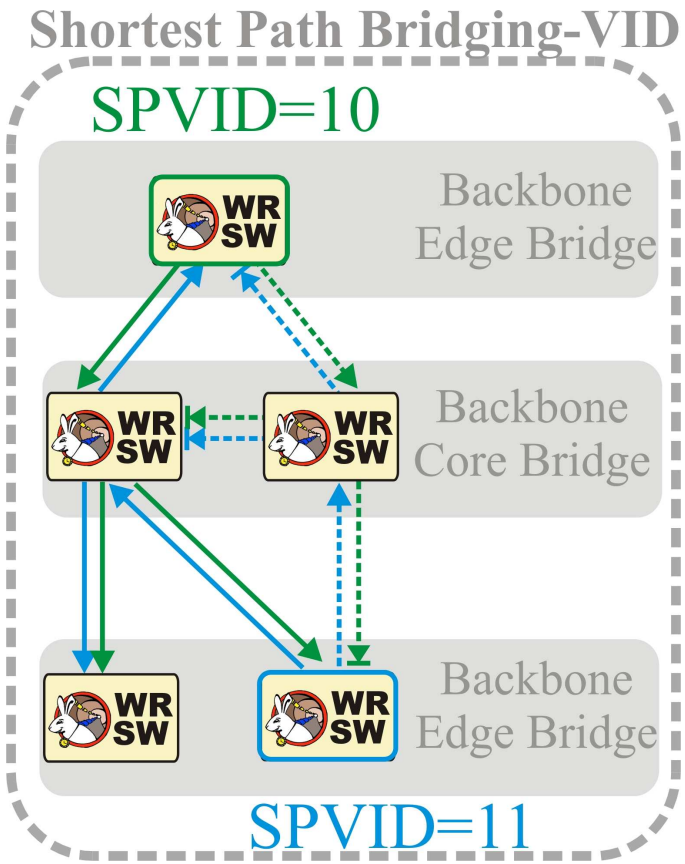
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree
- Backup tree – ports blocking on reception
- Single port forwarding from source
- H/W switch-over to path equally or more distant to the root
- More backup trees/ports possible (supported by H/W)



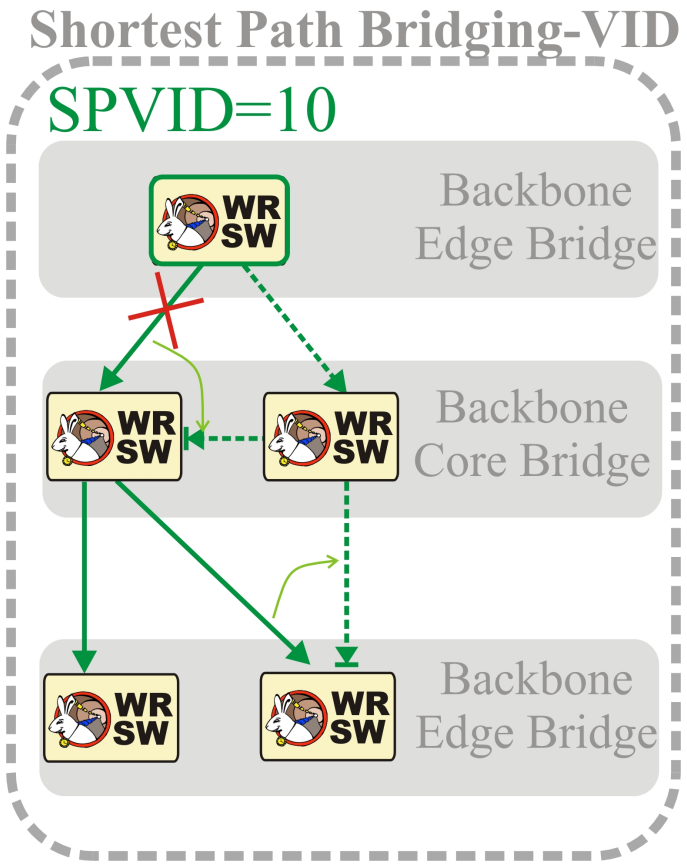
Shortest Path Bridging – VID for WR (SPBV-WR)

- Shortest Path Tree
- Backup tree – ports blocking on reception
- Single port forwarding from source
- H/W switch-over to path equally or more distant to the root
- More backup trees/ports possible (supported by H/W)
- Not fully congruent – is it a problem ?



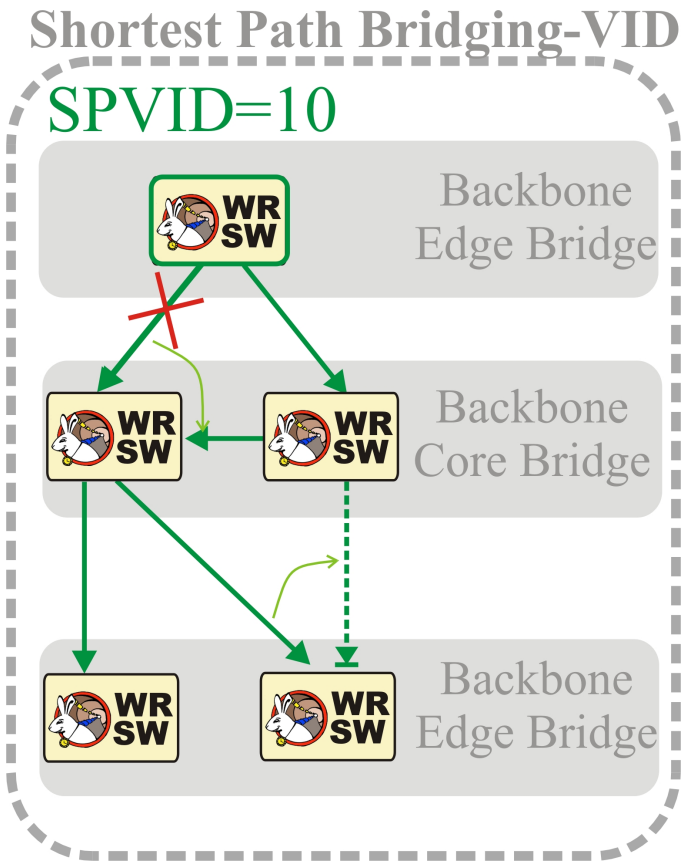
SPBV-WR: failure use case

- Link failure



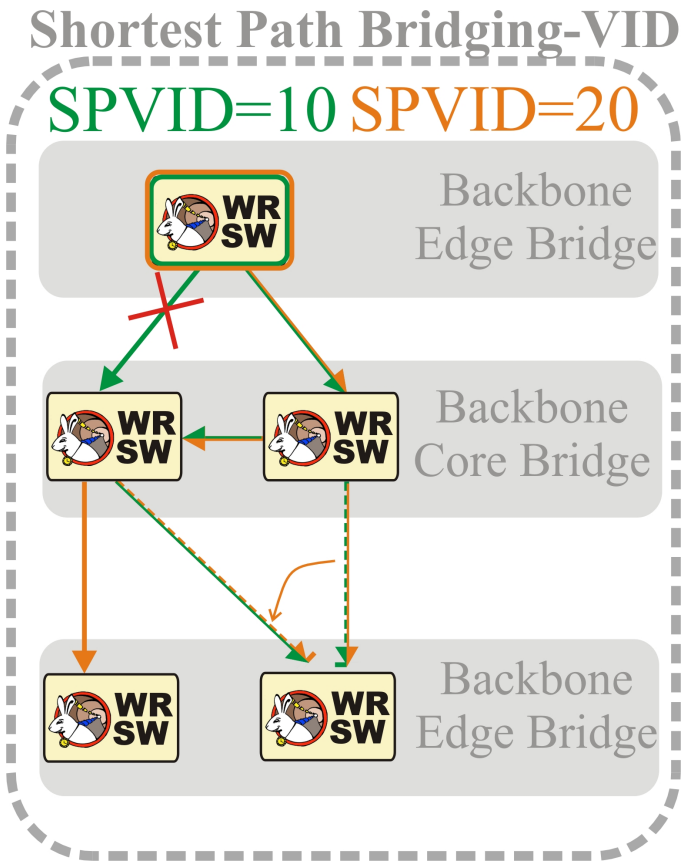
SPBV-WR: failure use case

- Link failure
- H/W switch-over to backup port



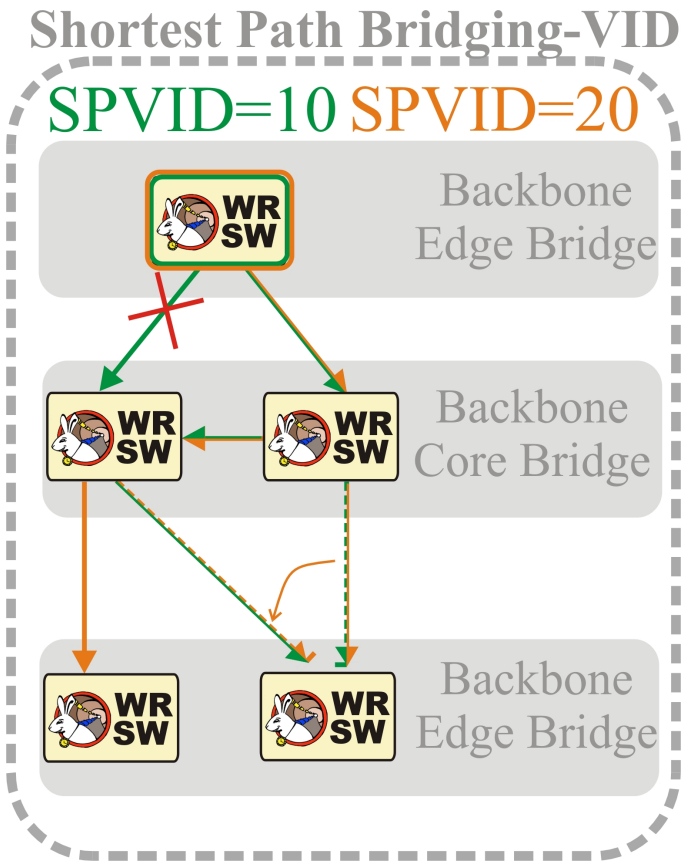
SPBV-WR: failure use case

- Link failure
- H/W switch-over to backup port
- New Shortest Path Tree installation on new SPVID



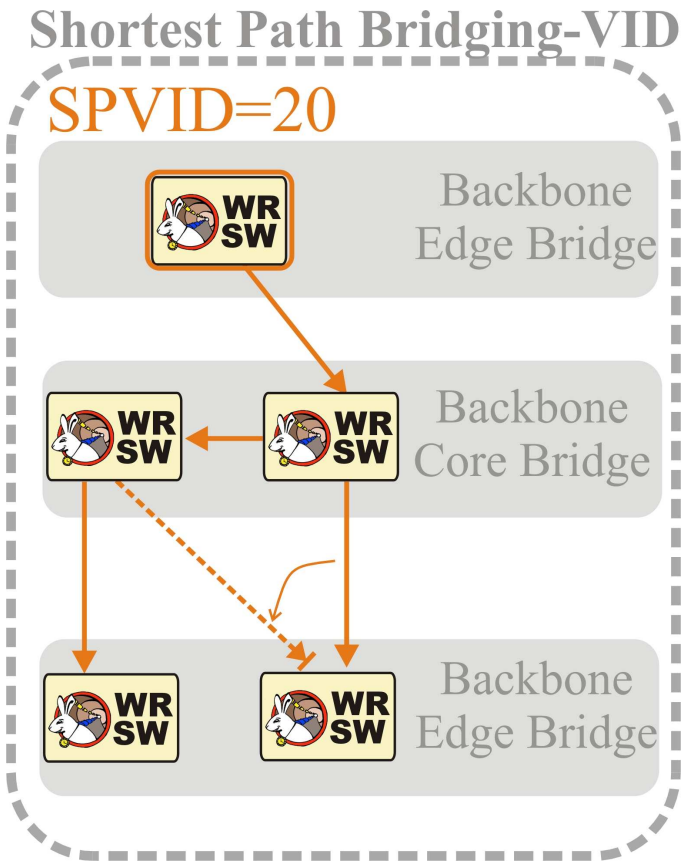
SPBV-WR: failure use case

- Link failure
- H/W switch-over to backup port
- New Shortest Path Tree installation on new SPVID
- When ready, starting to forward on SPVID



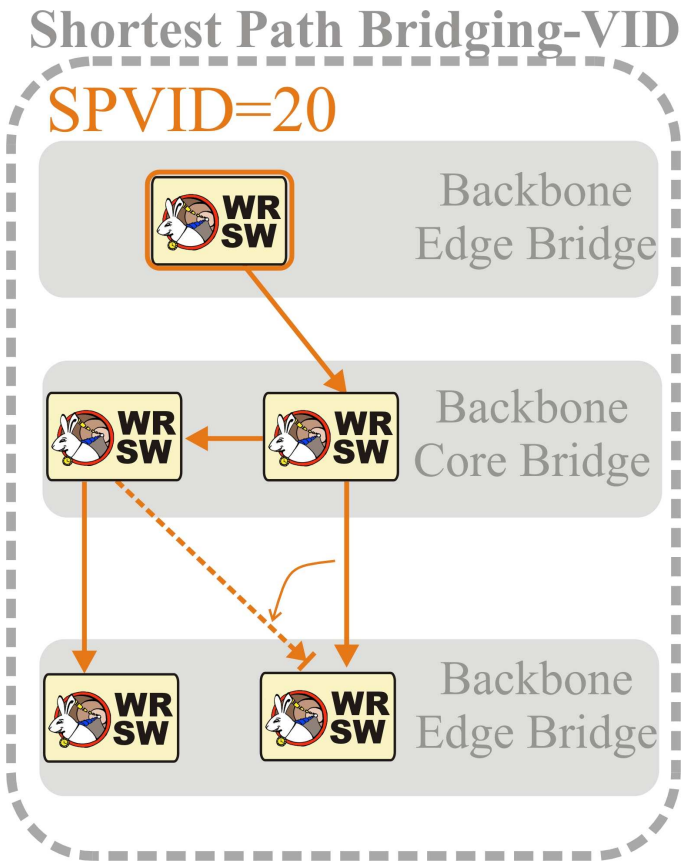
SPBV-WR: failure use case

- Link failure
- H/W switch-over to backup port
- New Shortest Path Tree installation on new SPVID
- When ready, starting to forward on SPVID
- Remove old SPVID



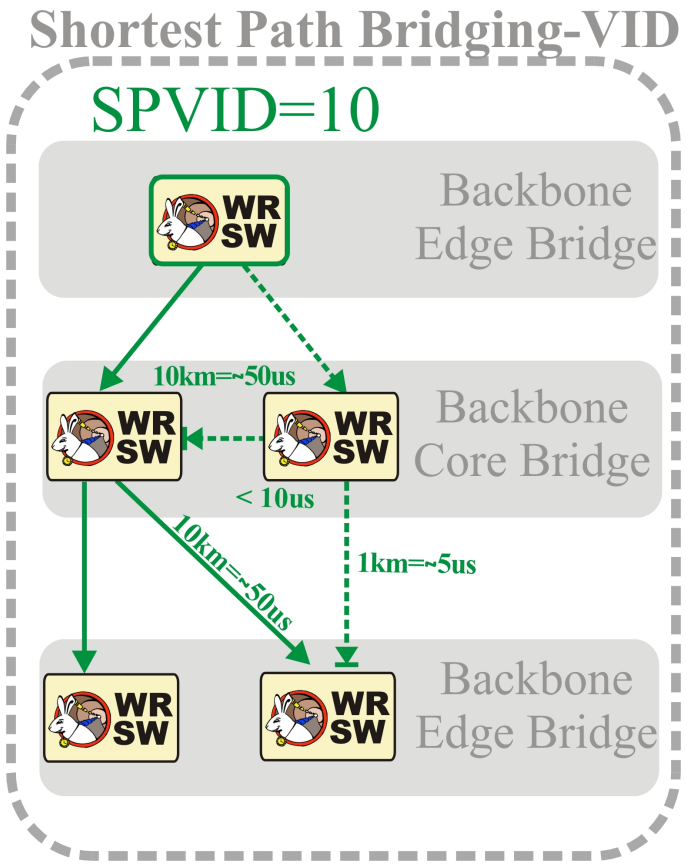
SPBV-WR: failure use case

- Link failure
- H/W switch-over to backup port
- New Shortest Path Tree installation on new SPVID
- When ready, starting to forward on SPVID
- Remove old SPVID
- **Does the standard allow this ?**



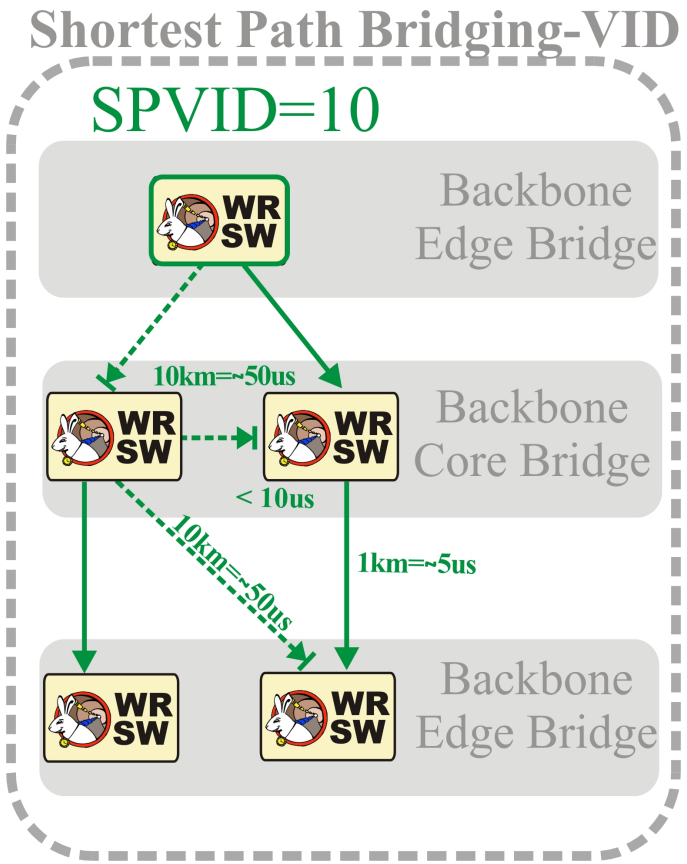
SPBV-WR: new link metrics

- Shortest Path $\neq$ Shortest Delay
- Precise knowledge of link delay
- New metric reflecting link delay (upper bound latency of switch)



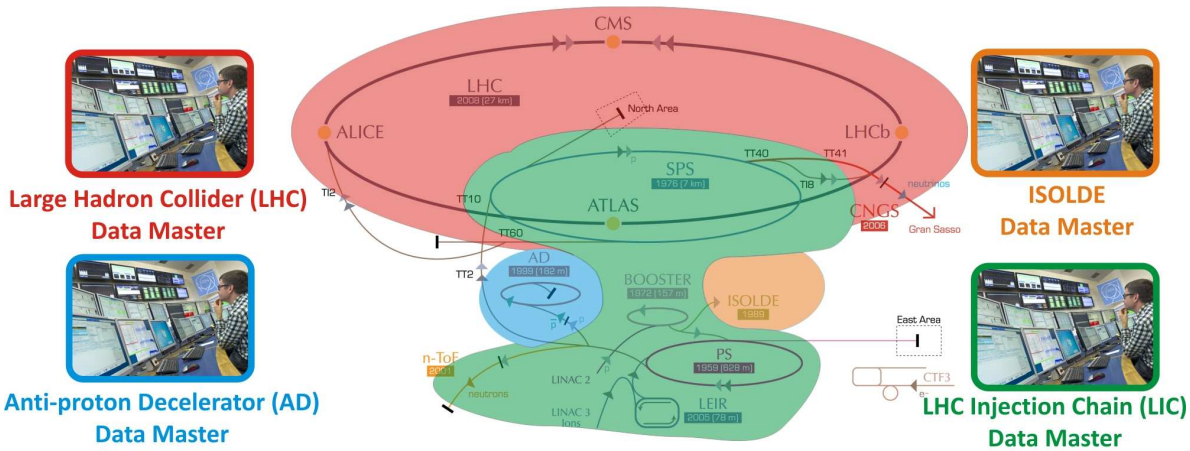
SPBV-WR: new link metrics

- Shortest Path $\neq$ Shortest Delay
- Precise knowledge of link delay
- New metric reflecting link delay (upper bound latency of switch)
- Effectively: Shortest Delay Tree



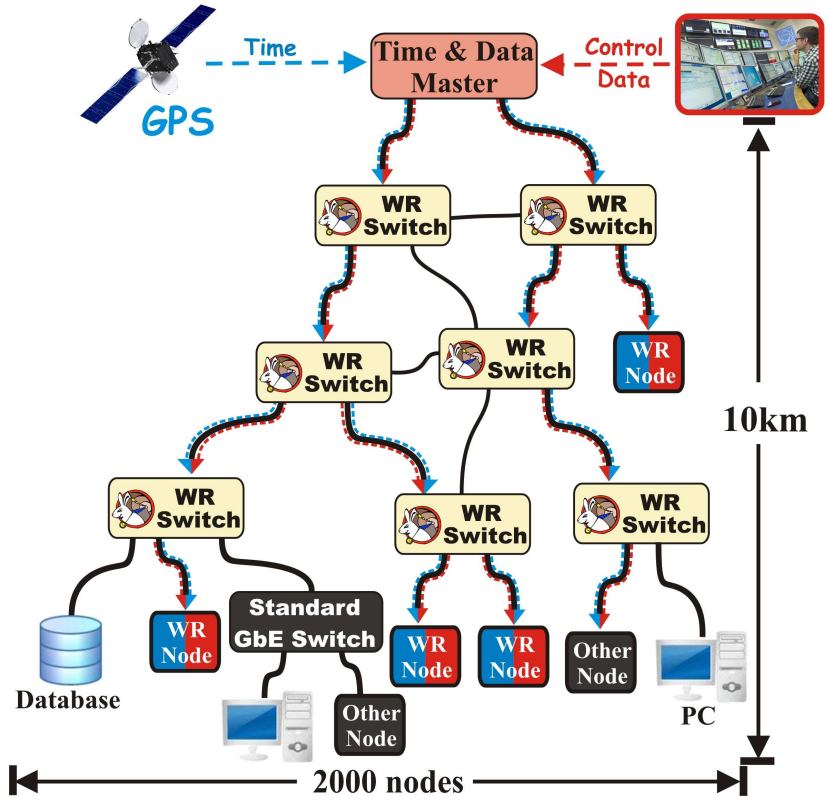
Outline

WR-based Control and Timing System (concept)

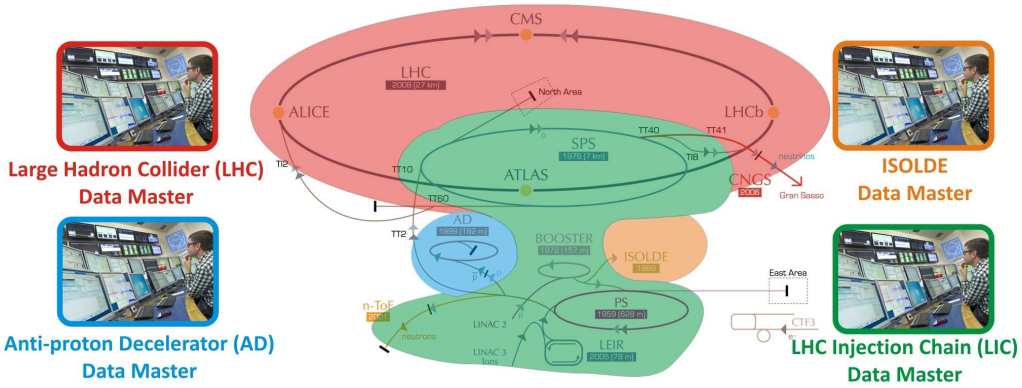


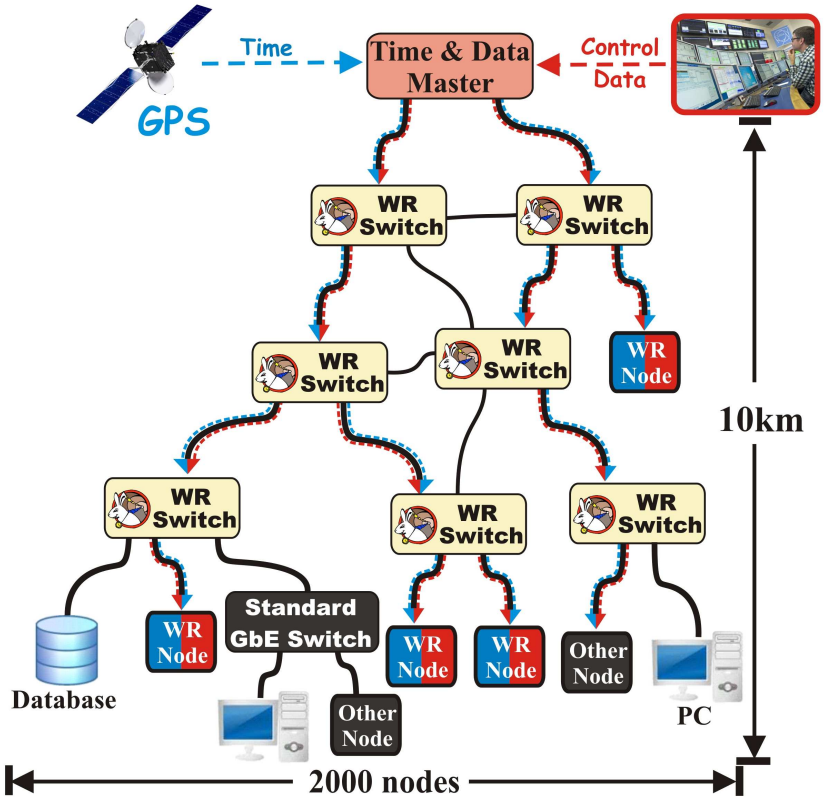
- 4 accelerator networks
- Separate **Data Master (DM)** for each network
- LIC Data Master communicates with other DMs and control devices in their networks
- Broadcast/multicast of **Control Messages**

WR-based Control and Timing System (concept)

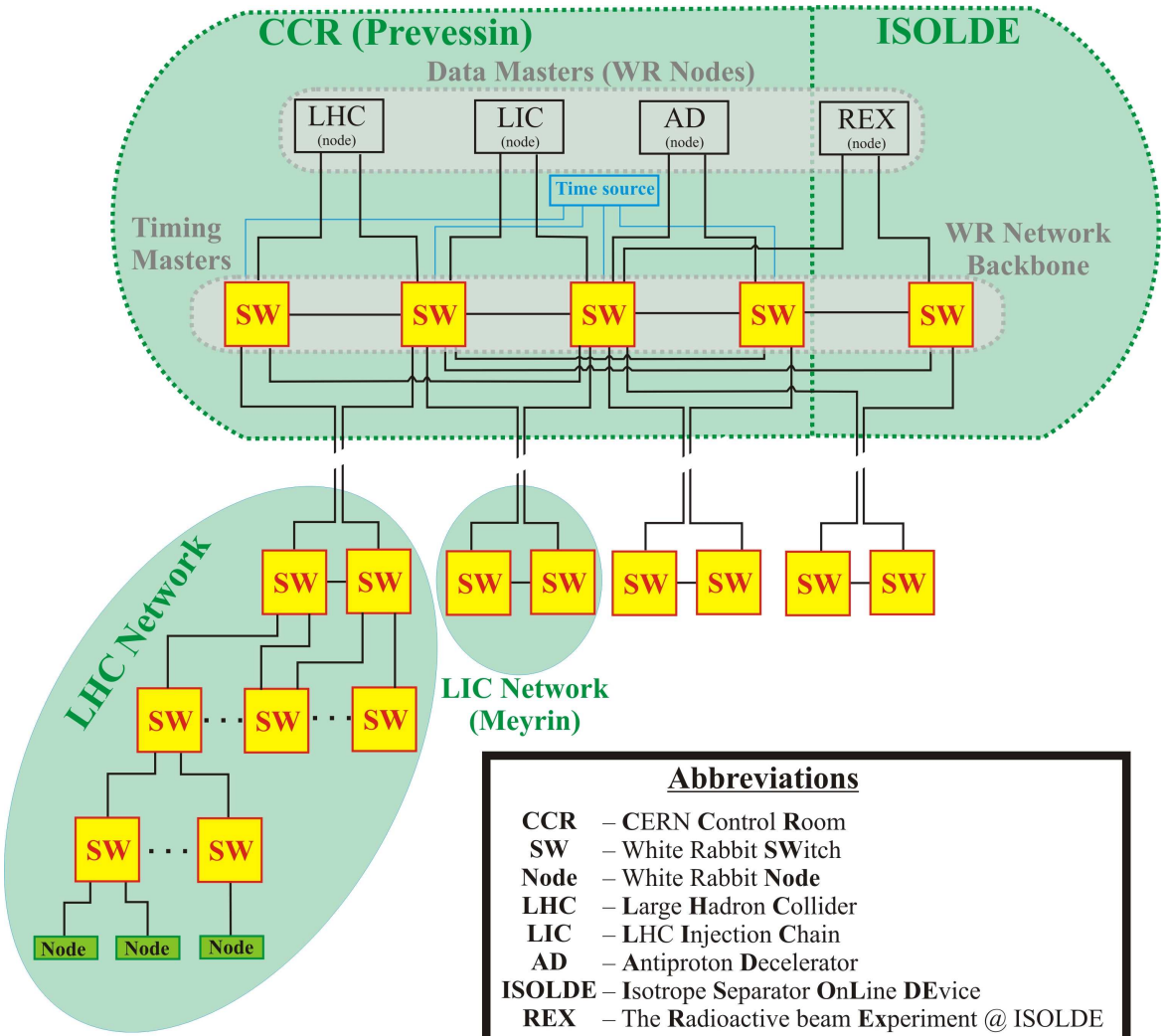


WR-based Control and Timing System (concept)

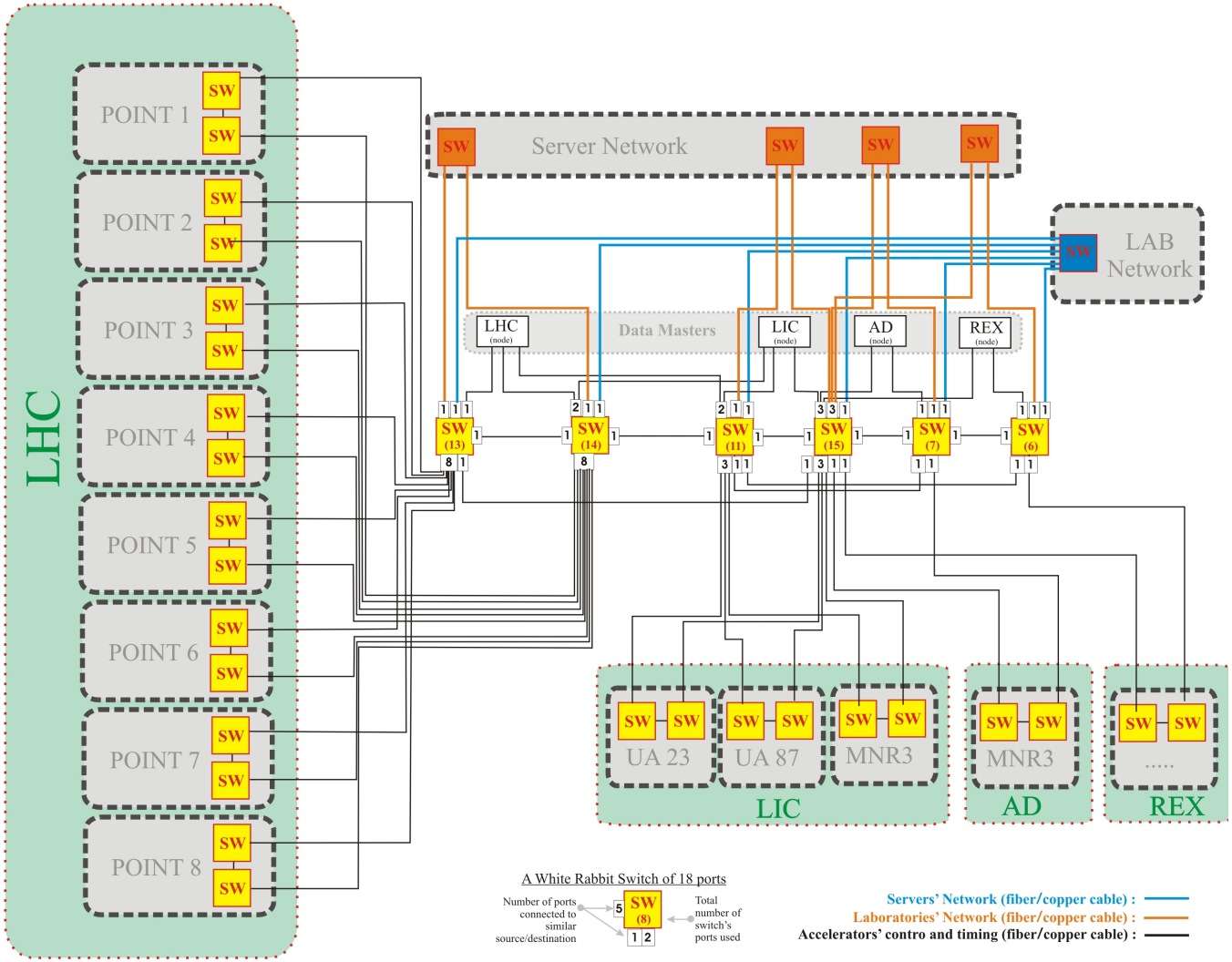




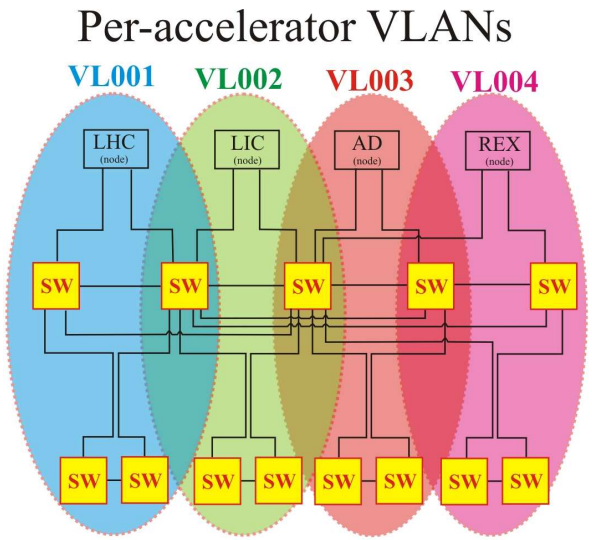
Accelerator Networks

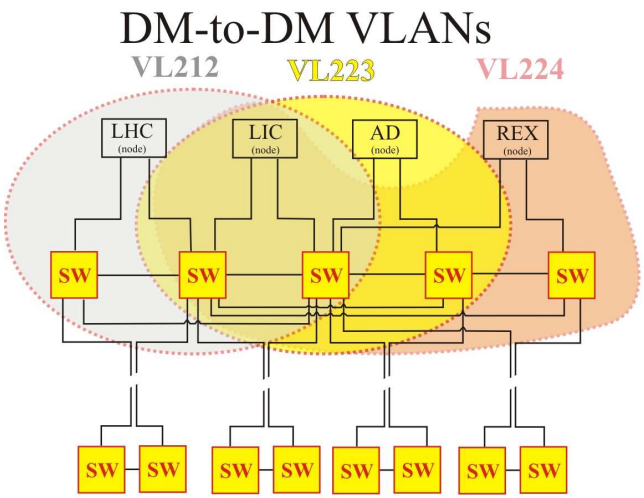


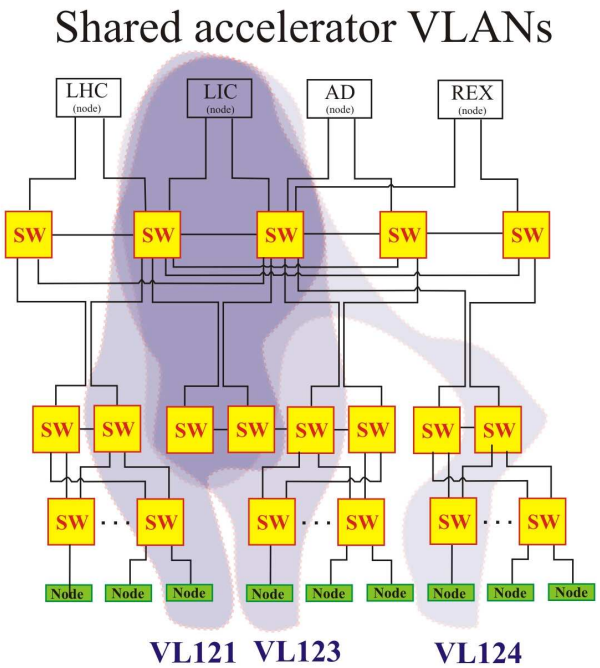
Accelerator and Auxiliary Networks



Traffic distribution: VLANs + multicast



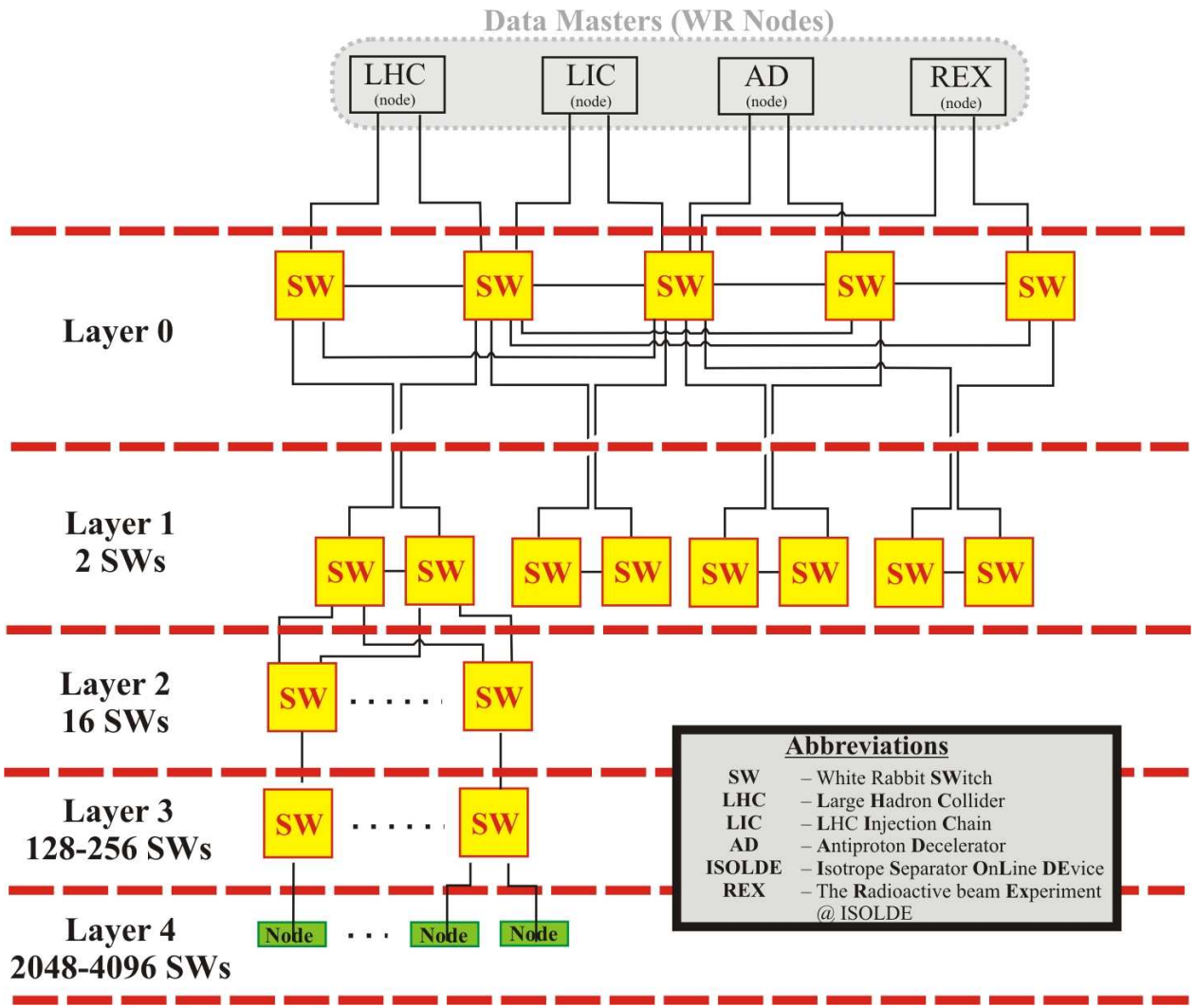




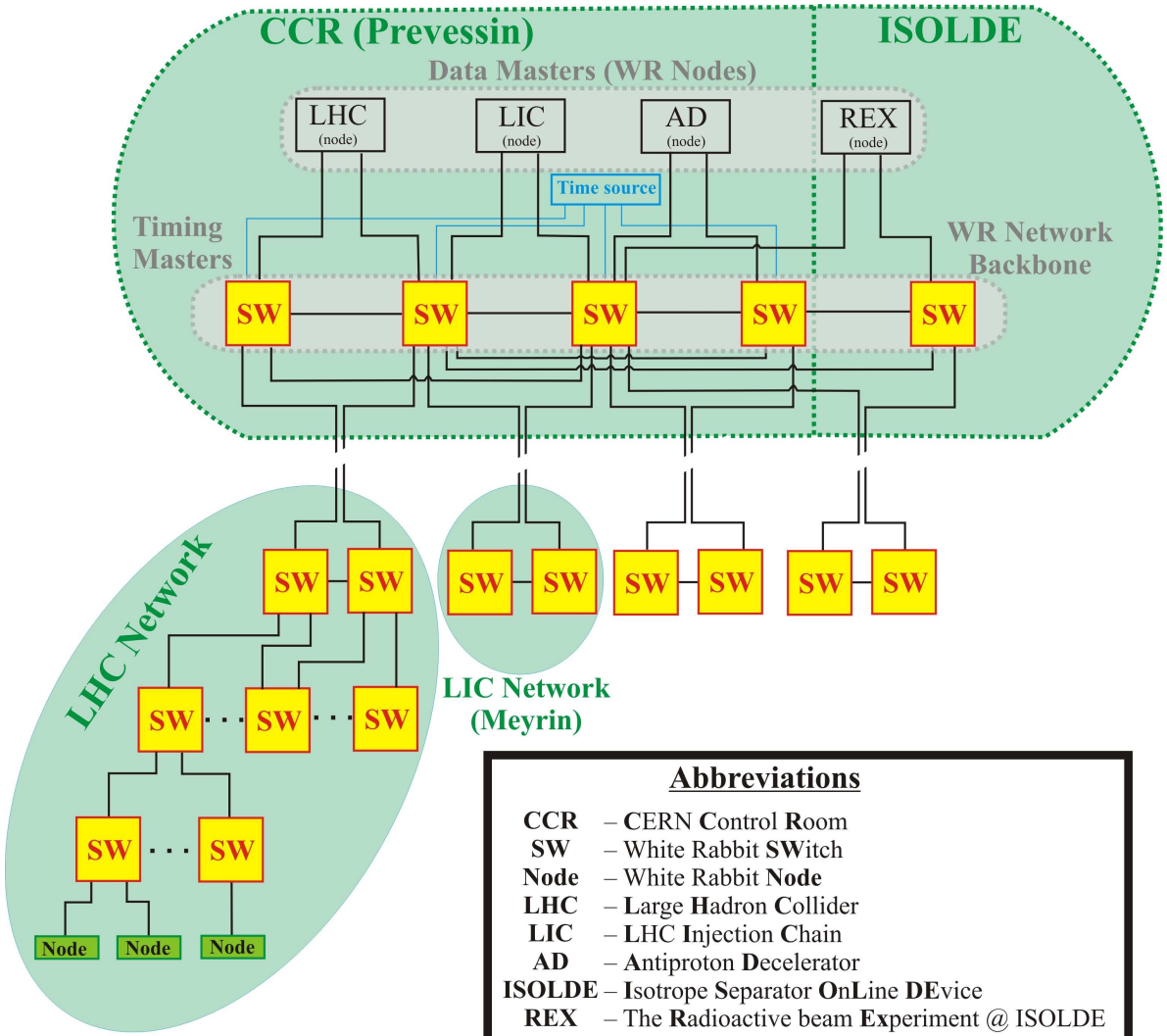
Abbreviations

SW	– White Rabbit **SW**itch	**AD**	– Antiproton **D**ecelerator
LHC	– Large **H**adron **C**ollider	**ISOLDE**	– **I**sotrope **S**eparator **O**n**L**ine **DE**vice
LIC	– LHC **I**njection **C**hain	**REX**	– The **R**adioactive beam **Ex**periment @ ISOLDE
DM	– **D**ata **M**aster		

Network Layers

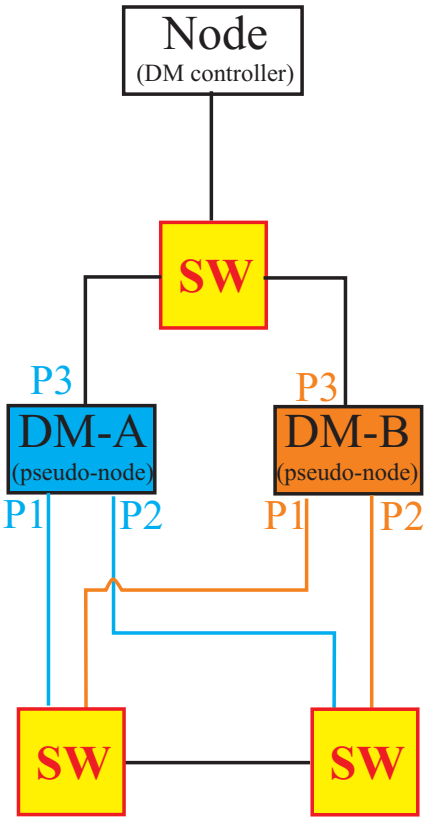


Data Masters



Multicast for redundant controllers (Data Masters)

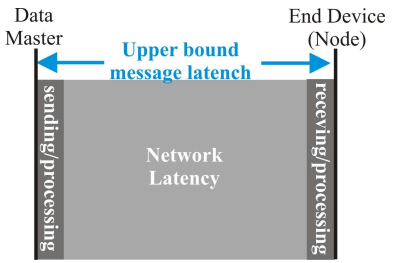
- Broadcast (unregistered multicast) communication: DM-to-nodes
- Multicast communication: nodes-to-DM
- Multicast address used for Data Masters (DM-A and DM-B)
- Seamless switch over between DMs: time-triggered synchronous reconfiguration of Layer 1 switches
- Nodes send data to multicast address: both DMs receive data
- No need for network reconfiguration when switching/changing DMs



Outline

Latency calculations for CERN

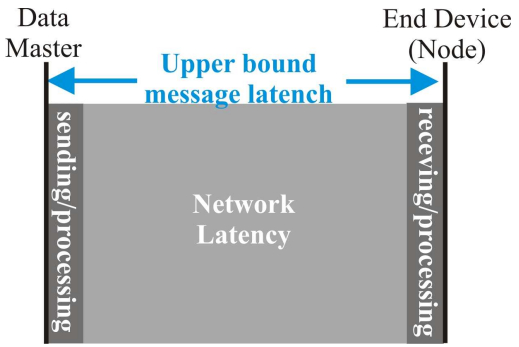
- Message latency (1000us) $\neq$ Network latency
- Tx/Rx worst case: 5000 bytes $=$(FEC)$=>$ 8 x 1500 bytes = 2x8x12us $\approx$ 200us
- FEC encoding = $\approx$ wire speed
- FEC decoding – good question
- Medium delay: 10km = $\approx$ 50 us
- Forwarding delay of 5 hops
  - $<$ 50 us if single source of High Priority
  - $<$ 50us + 2x8x12us = $\approx$ 250us if two sources of High Priority
- Worst case sum = 500 us



Requirements

Requirement	CERN	GSI
Synchronization accuracy	**up to ns**	
Upper-bound message latency	**1000us**	**200us**
Network span	**10km**	**2km**
End device number	**2000**	
Control Message size	**1200-5000 bytes**	$<$ **1500 bytes**
Data Master number	**4**	**1**
Traffic characteristics	**one-to-many**	
Number of CM lost per year	**1**	

Latency calculations for GSI



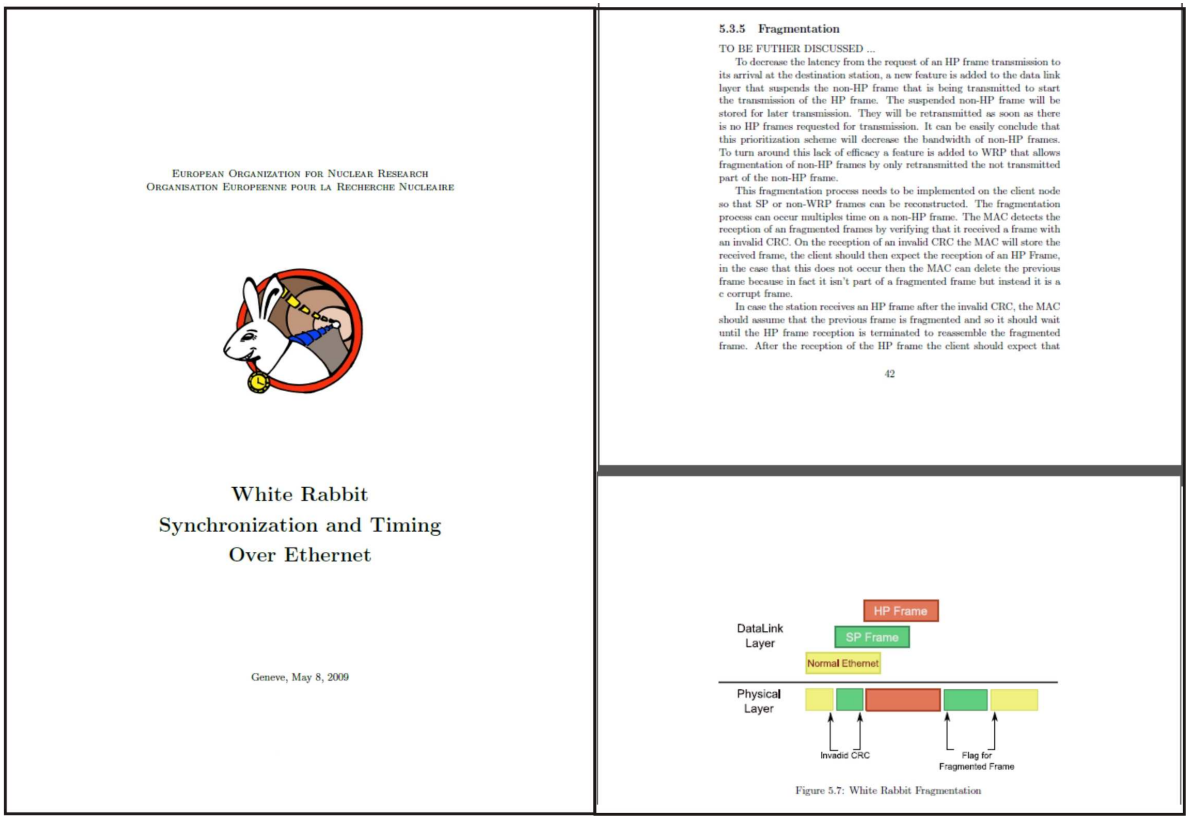
- Message latency (200us) $\neq$ Network latency
- Tx/Rx worst case: 1500 bytes $=$(FEC)$=>$ 4 x 1200 bytes = 2x4x9.4us $\approx$ 75us
- FEC encoding = $\approx$ wire speed
- FEC decoding - good question
- Medium delay: 2km = $\approx$ 10 us
- Forwarding delay of 5 hops $<$ 5x10us = 50 us
- Worst case sum = $\approx$ 135 us

Conclusions or rather questions

- Is what we are considering a crazy implementation of the standard ?
- We seem to have an extreme case of Stream Reservation:
  - large number of listeners
  - static Stream Reservation
- Path redundancy for broadcast/multicast seems especially challenging

Thank you

A twist of history



European Organization for Nuclear Research
Organisation Europeenne pour la Recherche Nucleaire

White Rabbit
Synchronization and Timing
Over Ethernet

Geneve, May 8, 2009

5.3.5 Fragmentation

TO BE FUTHER DISCUSSED ...

To decrease the latency from the request of an HP frame transmission to its arrival at the destination station, a new feature is added to the data link layer that suspends the non-HP frame that is being transmitted to start the transmission of the HP frame. The suspended non-HP frame will be stored for later transmission. They will be retransmitted as soon as there is no HP frames requested for transmission. It can be easily conclude that this prioritization scheme will decrease the bandwidth of non-HP frames. To turn around this lack of efficacy a feature is added to WRP that allows fragmentation of non-HP frames by only retransmitted the not transmitted part of the non-HP frame.

This fragmentation process needs to be implemented on the client node so that SP or non-WRP frames can be reconstructed. The fragmentation process can occur multiples time on a non-HP frame. The MAC detects the reception of an fragmented frames by verifying that it received a frame with an invalid CRC. On the reception of an invalid CRC the MAC will store the received frame, the client should then expect the reception of an HP Frame, in the case that this does not occur then the MAC can delete the previous frame because in fact it isn't part of a fragmented frame but instead it is a c orrupt frame.

In case the station receives an HP frame after the invalid CRC, the MAC should assume that the previous frame is fragmented and so it should wait until the HP frame reception is terminated to reassemble the fragmented frame. After the reception of the HP frame the client should expect that

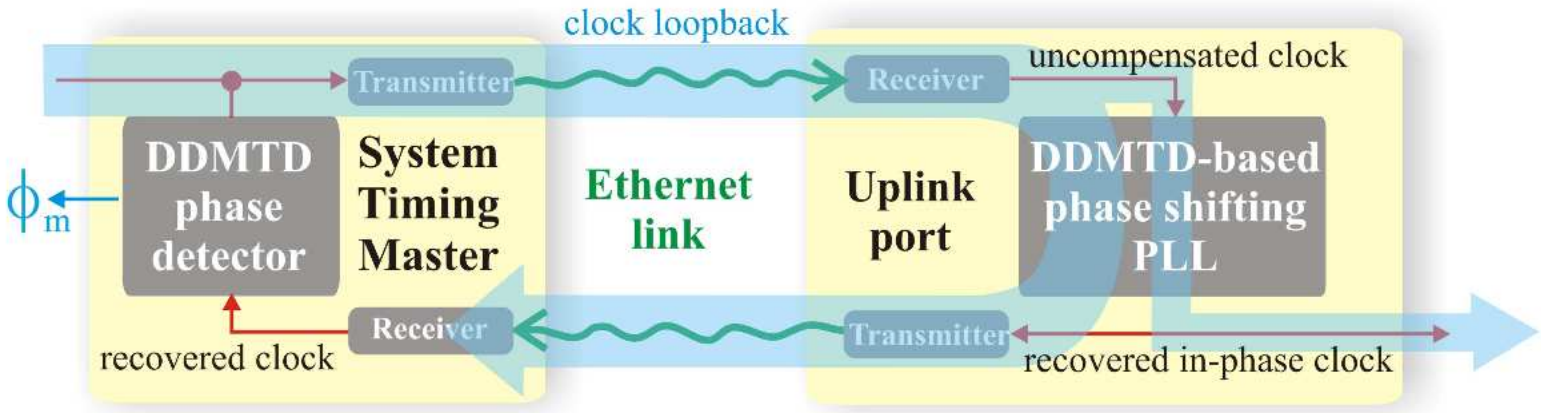
42

Figure 5.7: White Rabbit Fragmentation

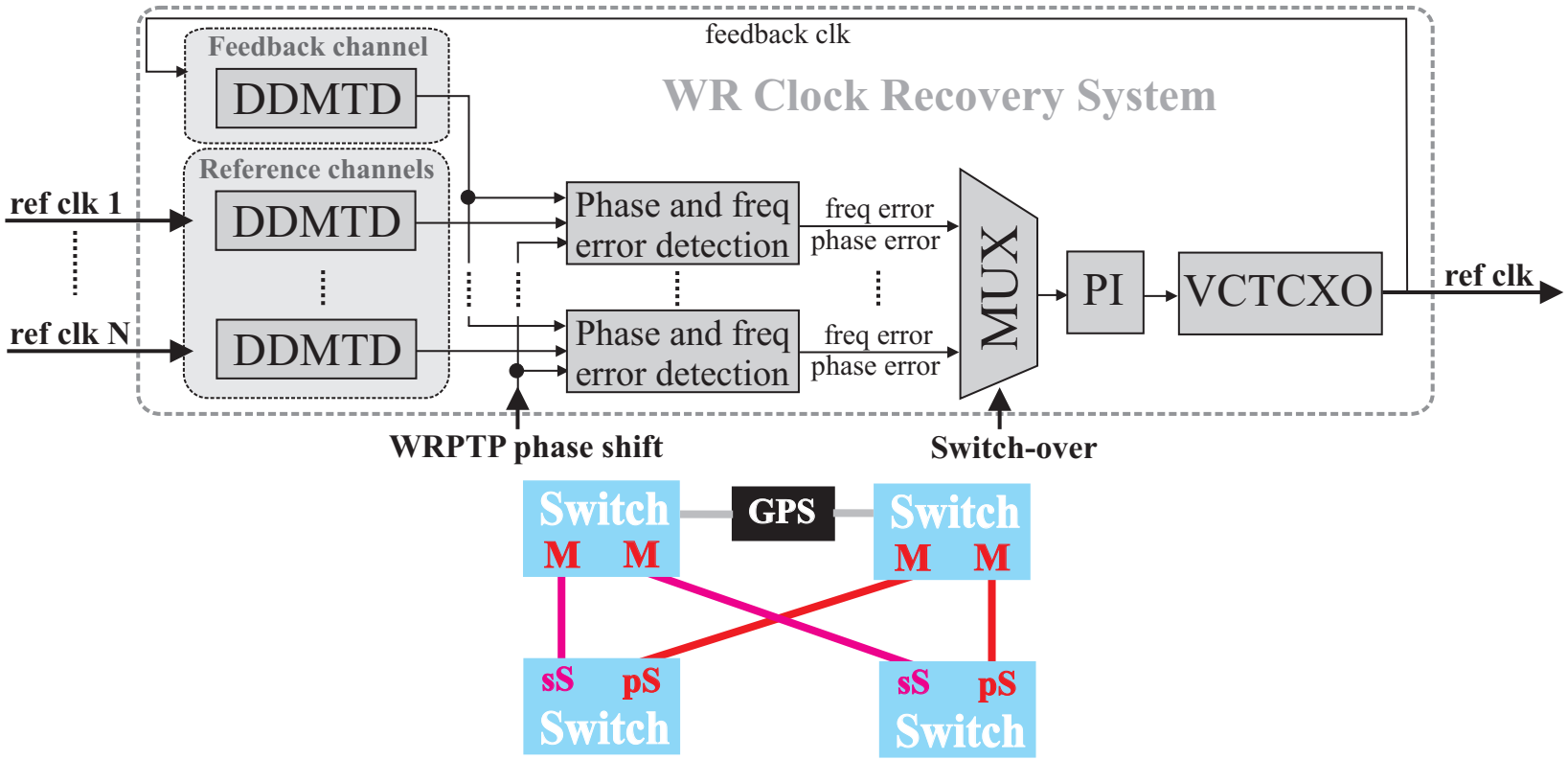
Frame segmentation (pre-emption)
in the White Rabbit Specification from 2009

Phase Tracking (DDMTD)

- Monitor phase of bounced-back clock
- Enhance PTP timestamps with phase measurement
- Phase-locked loop in the slave follows the phase changes

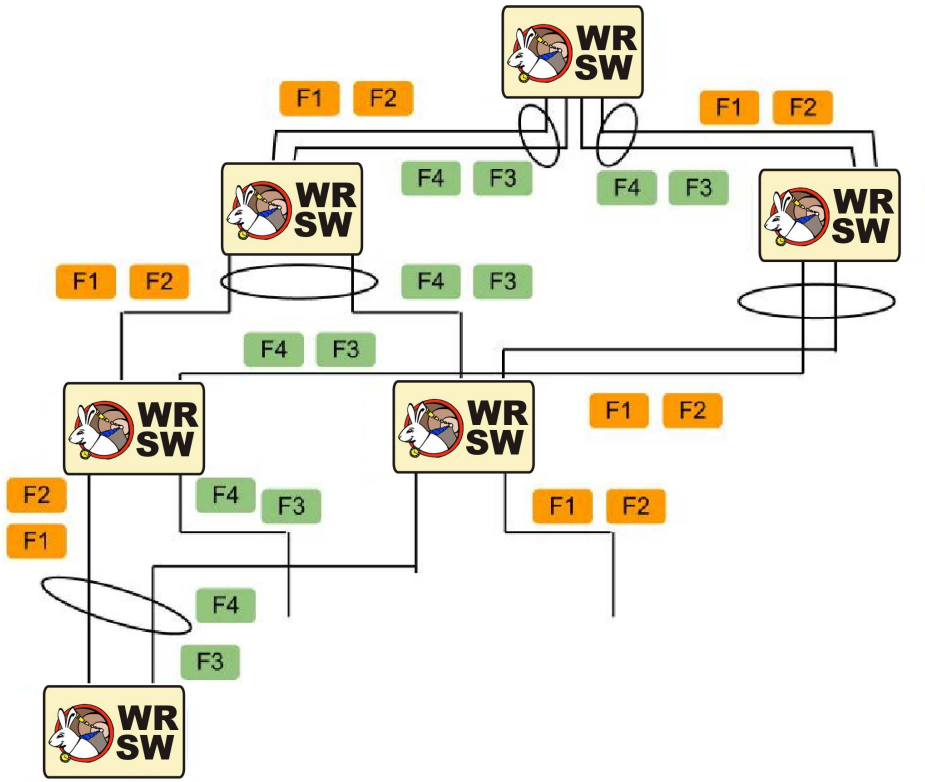


Redundancy for time distribution



Topology Redundancy: eLACP (short explanation)

Control Message encoded into 4 Ethernet Frames (F1,F2,F3,F4). Reception of any two enables to recover Control Message (*Cesar Prados, GSI).*



Courtesy of Cesar Prados