Carrier Ethernet Service Protection over UNIs and E-NNIs

Zehavit Alon
Nurit Sprecher

January 2010

Nokia Siemens Networks

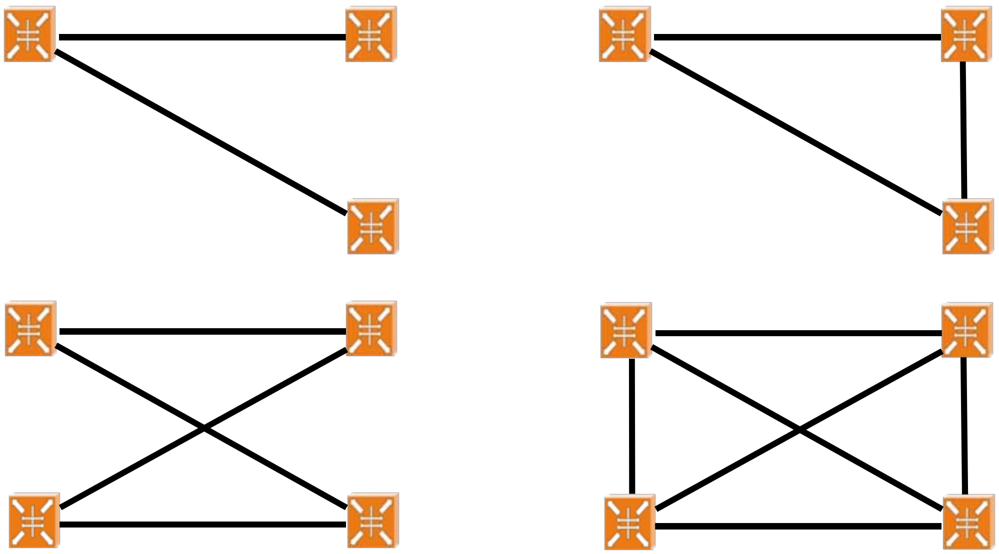
Purpose

The purpose of this presentation is to clarify and emphasize some of the attributes and characteristics of the service protection mechanism over UNI/E-NNI proposed in http://www.ieee802.org/1/files/public/docs2009/new-alon-UNI-ENNI-protection-09-09-v01.ppt

It is assumed that the audience is familiar with the main ideas of the mechanism.

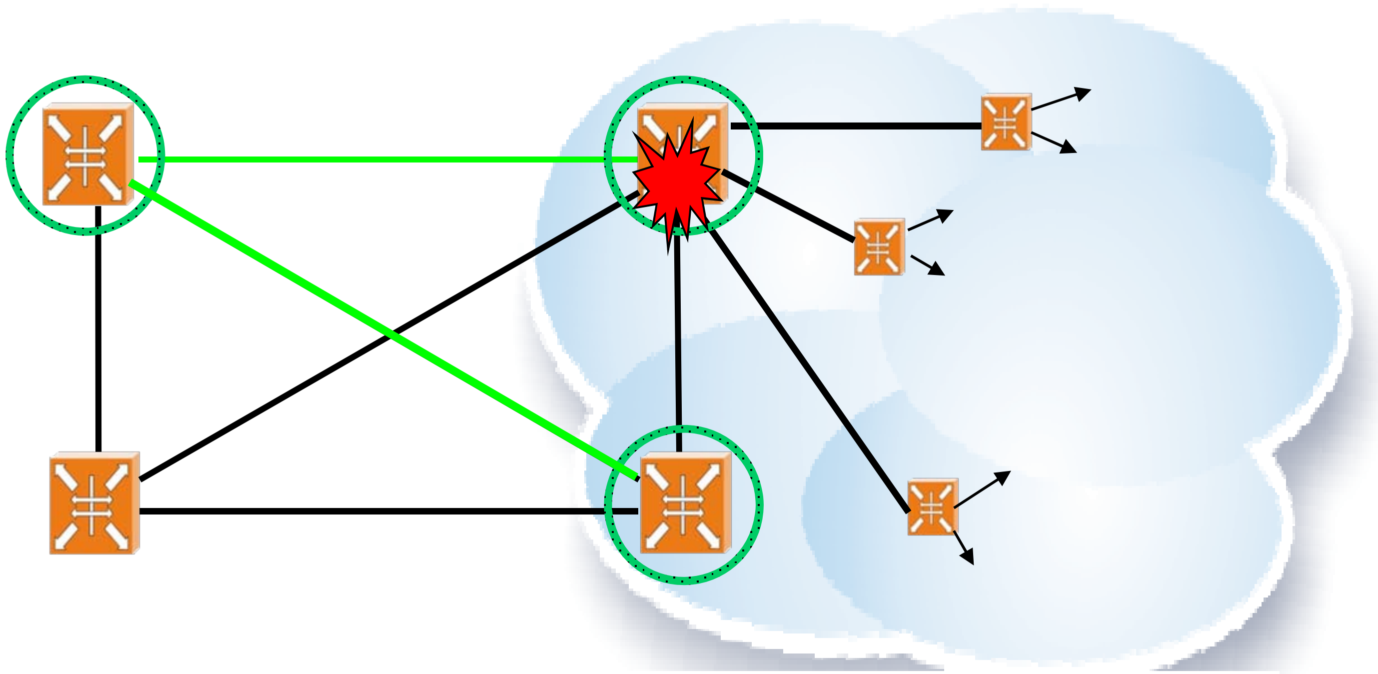
Recap

The subject of Carrier Ethernet Service Protection over network-to-network interfaces (UNIs and E-NNIs) was discussed and the following constructs were recommended:



Mechanism highlights
Failures and repairs (1)

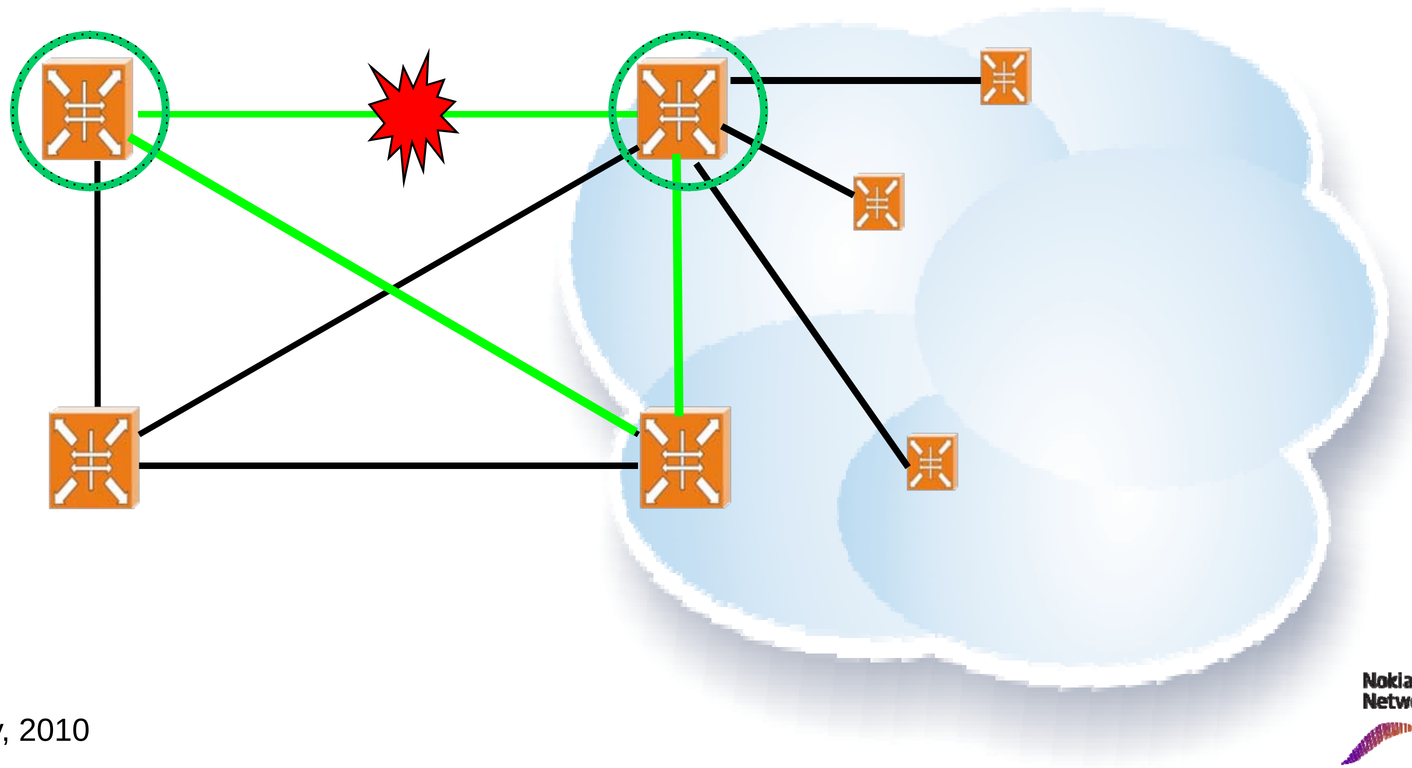
When a Traffic Gateway (TG) node fails, another node becomes a TG; changes in the attached network (to which the failed node belongs) are inevitable (regardless of whether or not an internal link is present).



The TG is the node that conveys traffic from the network to the interconnected zone, and from the interconnected zone to the network.

Mechanism highlights
Failures and repairs (2)

When a link between TG nodes (connecting the attached networks) fails, a bypass route may be established between the slave nodes (if the internal link is present) to preserve the TGs. Changes in the attached network are avoided.



Protection mechanism
General (1)

- The protection mechanism operates per VLAN (or group of VLANs).
- The nodes' role is asymmetric in the construct.
  - One of the networks is the owner of the control nodes which control the link over which traffic will be conveyed.
  - The other network is the owner of the slave nodes. A control node issues explicit orders to the slave nodes that are to become TGs. A slave will not become a TG without an explicit order.
- Connectivity between the networks is **MANDATAORY**.
- A minimum of two links is required for link protection.
- A minimum of two nodes is required for node protection. When node protection is not required, one node is sufficient.

Mechanism highlights
General (2)

- Connectivity between the control nodes and connectivity between the slave nodes is **OPTIONAL**.
  - In the absence of connectivity between the control nodes, each control node discovers its peers' state by acknowledging the states of the attached slaves. (If one of the slaves is a TG, one of the control nodes must be a TG, as a slave becomes a TG if, and only if, it receives an explicit request from a control node.)
  - A bypass cannot be created when there is no connectivity between the slave nodes, but protection switching will provide the required protection. However, when protection switching occurs, the protection switching event will cause topology changes in the attached network.

Mechanism highlights
Node states

- Each node is responsible for its own state. A new state is calculated on the basis of the current state plus control messages that are received (or not) on the links. A node starts in standby state (does not convey traffic).
- Only one node at a time* is active in each network (one slave and one control node), i.e. the node acts as a TG.
- Only one port at a time is active (sends and receives frames) in each TG node that connects the networks, i.e. a frame is sent and received by a maximum of one port of a TG node.
- Thus:
  - A loop cannot be created.
  - Packets are sent only once. (Packets cannot be duplicated.)
  - Packets may get lost.
  - In certain circumstances, connectivity between the adjacent networks may be lost for a short while (less than 50 ms).

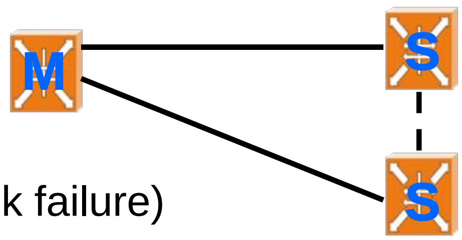
•**The only exception is detailed in Corner case in the backup slides.**

Mechanism highlights
Possible configurations

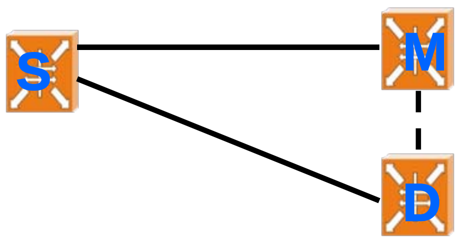
Possible connectivity options (static configuration only):

- One master, one slave
  (No protection)

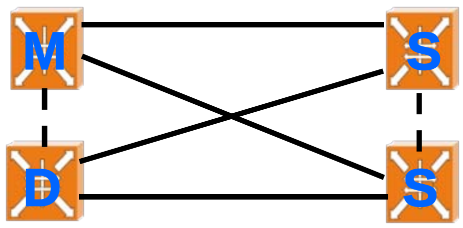
- One master with two slaves
  (Master node is a single point of failure; preserves TG during link failure)



- One master, one deputy, and one slave
  (Slave node is a single point of failure)



- One master, one deputy, and two slaves
  (No single point of failure)



Mechanism highlights
Node functionality (1)

Each link and each node can fail and recover. On recovery, the node adjusts to the current protection state of the whole construct, which it learns from received messages.

- When the master recovers, it immediately becomes a TG in either node revertive-mode, or in node non-revertive mode if the deputy is not a TG. Otherwise it becomes standby.
  - If there is an internal link, the active control node can be detected directly. If there is no internal link, it can be detected indirectly from the slave's state.
- When the deputy recovers, it becomes standby. If the master is not a TG, the deputy becomes a TG.
- A slave becomes a TG only when a specific request is received from a control node. A recovered slave will remain in standby state until specifically asked to become a TG.
- When the link configured as working recovers, a control node may choose to use this link to convey traffic (in link revertive-mode).

Mechanism highlights
Node functionality (2)

- A maximum of one node in each network (one control node and one slave node) may be a TG, i.e. conveys traffic to and from the interconnected zone.
- A maximum of one port connecting the networks may be active on each node, i.e. conveys traffic to and from the interconnected zone.
  - The internal port may also be active, however the internal port transmits traffic to the other slave in the same network.
- The transition between states is deterministic and unambiguous. (For the flow chart and transition table, see the Flow Chart and Transition table slides.)

Delay effects

- In all node types, traffic stops immediately whenever a failure is detected.
- A control node chooses a single link to convey traffic. If the slave is unaware of the control node's decision, traffic received on the chosen link will be discarded.
- A slave node accepts requests to transmit traffic from one control node only. If it receives a request from both control nodes, it either ignores one request or changes the state of both of its ports to standby.
- A delay may only result in the construct requiring a longer than expected recovery time.
  - The maximum expected recovery time, including failure detection without delays, is 9 times the control message interval, i.e. less than 30ms when the control messages are sent every 3.3ms. (For timing issues, see the Timing slide.)

Thank You

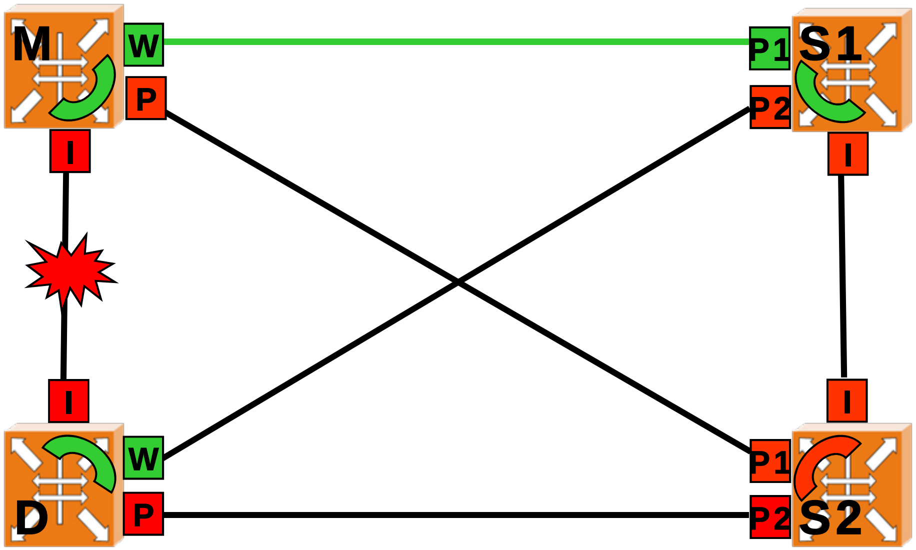
zehavit.alon@nsn.com
nurit.sprecher@nsn.com

Backup slides

Exception

Two TGs with active ports in the same network

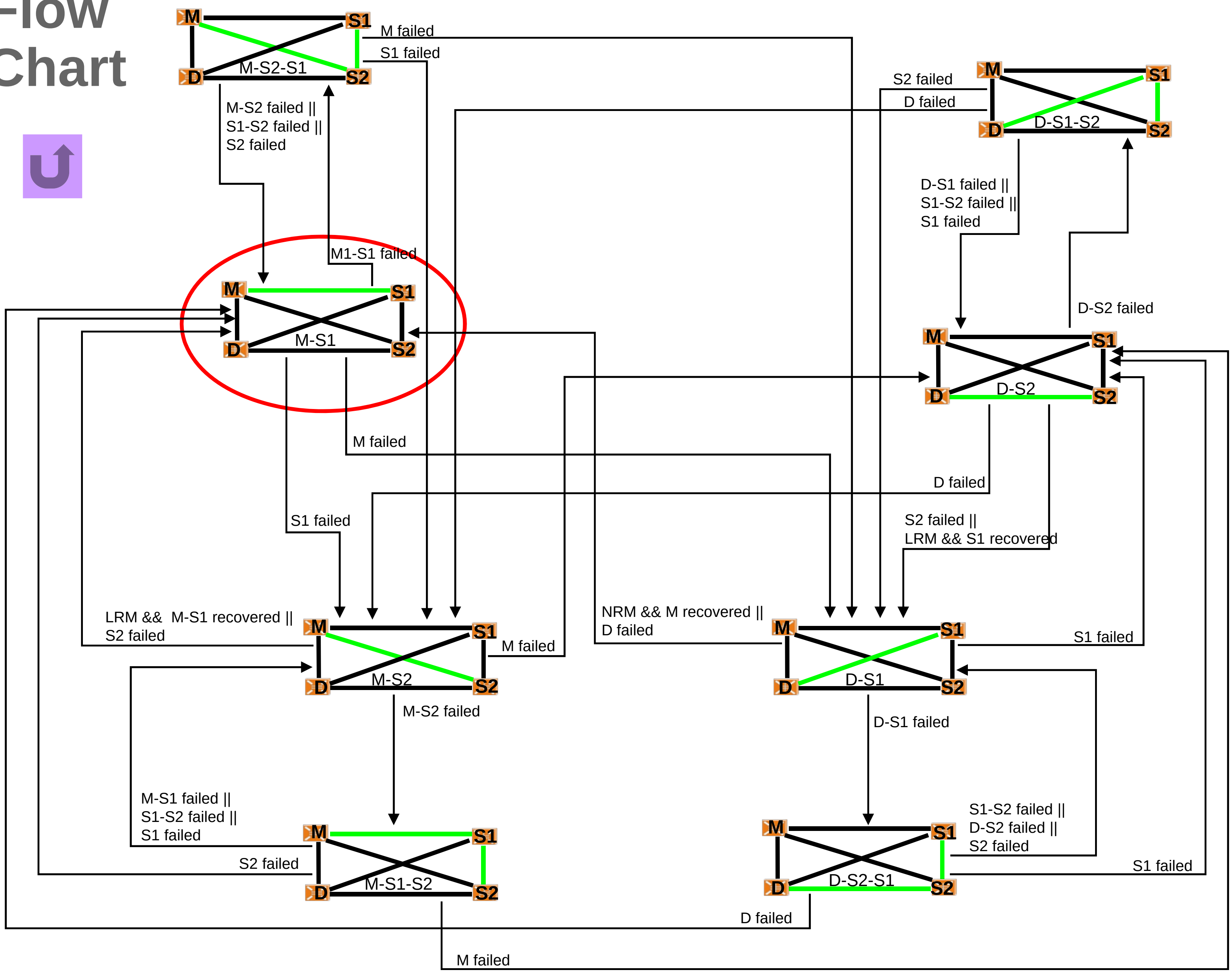
A scenario may occur where both control nodes are active for a short while. This is only possible when an internal connection between the control nodes fails. If this happens, the slave acting as the Traffic Gateway ignores the duplicated packets, since only one active port is allowed.



Timing

- The master maximum initialization time will not exceed 5 intervals:
  - Understand that there is no active control node – 1 interval
  - Become TG – 1 interval
  - Activate port – 1 interval
- The deputy maximum initialization time will not exceed 7 intervals:
  - Understand that there is no active control node – 3 intervals
  - Become TG – 1 interval
  - Activate port – 1 interval
- The slave maximum initialization time will not exceed 2 intervals:
  - Become TG and activate port after receiving a command from a control node
- Overall recovery time from failure detection to full recovery will not exceed 9 intervals:
  - Master failure discovery and slave change to standby – 3 intervals
  - Deputy detects that there is no active control node and becomes active – 3 intervals
  - Slave becomes active – 1 interval

Flow Chart



M
S1
D
S2
M-S2-S1
M failed
S1 failed
M-S2 failed ||
S1-S2 failed ||
S2 failed
S2 failed
D failed
D-S1-S2
D-S1 failed ||
S1-S2 failed ||
S1 failed
M1-S1 failed
M-S1
D-S2 failed
D-S2
M failed
D failed
S1 failed
S2 failed ||
LRM && S1 recovered
LRM && M-S1 recovered ||
S2 failed
M failed
M-S2
NRM && M recovered ||
D failed
S1 failed
D-S1
M-S2 failed
D-S1 failed
M-S1 failed ||
S1-S2 failed ||
S1 failed
S2 failed
M-S1-S2
S1-S2 failed ||
D-S2 failed ||
S2 failed
S1 failed
D-S2-S1
D failed
M failed

Transition table

Facility failure / State	M	S1	D	S2	M-S1	S1-S2	D-S2	M-D	M-S2	D-S1	M recovers Node revertive mode	Link recovers Link revertive mode	M	D
M-S1	D-S1	M-S2	-	-	M-S2-S1	-	-	-	-	-	-	-	D-S1	-
D-S2	-	-	M-S2	D-S1	-	-	D-S1-S2	-	-	-	M-S2	D-S1	-	M-S1
M-S2	D-S2	-	-	M-S1	-	-	-	-	M-S1-S2	-	-	M-S1	D-S1	-
D-S1	-	D-S2	M-S1	-	-	-	-	-	-	D-S2-S1	M-S1	-	-	M-S1
M-S2-S1	D-S1	M-S2	-	M-S1	-	M-S1	-	-	M-S1	-	-	M-S1	D-S1	-
M-S1-S2	D-S2	M-S2	-	M-S1	M-S2	M-S2	-	-	-	-	-	-	D-S1	-
D-S1-S2	-	D-S2	M-S2	D-S1	-	D-S2	-	-	-	D-S2	M-S2	-	-	M-S1
D-S2-S1	-	D-S2	M-S1	D-S1	-	D-S1	D-S1	-	-	-	M-S1	D-S1	-	M-S1

Note 1: A bypass to a failed link always passes through a slave (never through a control node).

Note 2: The last two columns are for constructs that only have five links. (The control nodes are not connected.)

Note 3: The scenario where there are only four links can be reached by removing the S1-S2 connectivity, when applicable.