Recovery Amendment to E-NNI 2.0 Common Part

IA # OIF-ENNI-REC-AM-01.0

February 4, 2014

Implementation Agreement created and approved
by the Optical Internetworking Forum
www.oiforum.com

The OIF is an international non profit organization with over 100 member companies, including the world's leading carriers and vendors. Being an industry group uniting representatives of the data and optical worlds, OIF's purpose is to accelerate the deployment of interoperable, cost-effective and robust optical internetworks and their associated technologies. Optical internetworks are data networks composed of routers and data switches interconnected by optical networking elements.

With the goal of promoting worldwide compatibility of optical internetworking products, the OIF actively supports and extends the work of national and international standards bodies. Working relationships or formal liaisons have been established with CFP-MSA, COAST, Ethernet Alliance, Fibre Channel T11, IEEE 802.1, IEEE 802.3, IETF, InfiniBand, ITU-T SG13, ITU-T SG15, MEF, ONF, Rapid I/O, SAS T10, SFF Committee, TMF and TMOC.

For additional information contact:
The Optical Internetworking Forum, 48377 Fremont Blvd.,
Suite 117, Fremont, CA 94538
510-492-4040 ☞ info@oiforum.com

www.oiforum.com

Working Group: **Networking and Operations**

TITLE: **Recovery Amendment to E-NNI 2.0 – Common Part**

SOURCE:

TECHNICAL EDITOR

Remi Theillaud
Marben Products
176 rue Jean Jaures
92800 Puteaux
France
Phone: +33 (1) 79 62 10 22
Email: remi.theillaud@marben-products.com

WORKING GROUP CHAIR

Evelyne Roch
Huawei Technologies
303 Terry Fox Drive Suite 400
Ottawa, ON K2K 3J1
Canada
Phone: +1 613 595 1900 x1612
Email:evelyne.roch@huawei.com

ABSTRACT: This Implementation Agreement defines generic Signaling and Routing extensions to OIF E-NNI 2.0 to support recovery mechanisms.

List of Contributors

Thierry Marcot
thierry.marcot@orange.com

Jonathan Sadler
Jonathan.Sadler@coriant.com

Evelyne Roch
evelyne.roch@huawei.com

Richard Graveman
rfgraveman@gmail.com

Lyndon Ong
Lyong@Ciena.com

Jim Jones
jim.d.jones@alcatel-lucent.com

Fred Gruman
fred.gruman@us.fujitsu.com

Torsten Wuth
torsten.wuth@nsn.com

Ben Wright
ben.wright@metaswitch.com

Hans Martin Foisel
H.Foisel@telekom.de

1 Table of Contents

2 List of Figures

3 List of Tables

4 Introduction

Recovery from failures is a crucial feature for network operators as it helps to keep the services to their customers up and running during network outages.

While control plane-based recovery mechanisms exist in intra-domain context, there is a lack of implementation specifications in ASON inter-domain context.

In order to overcome this limitation, the carrier WG provided to the OIF community a "Guideline Document" on Control Plane Requirements for Multi-Domain Optical Transport Networks that provides requirements for Control-plane Based recovery mechanisms [OIF-CWG-CPR-01.0].

In a second step, the OIF has successfully demonstrated an exploratory control plane-based service hard-rerouting during the OIF 2009 worldwide demonstration event.

At last, this document specifies the methods and extensions necessary to provide recovery signaling support in the E-NNI 2.0 Signaling & E-NNI 2.0 Routing environments. It allows control plane-enabled networks to support recovery in multi-domain context as defined by [G.8080].

This document is an amendment to E-NNI signaling 2.0 and to E-NNI Routing 2.0, reusing the terms and concepts described in [OIF-E-NNI-sig-02.0] and [OIF-ENNI-OSPF-02.0].

4.1 Problem Statement

There are standards that define some signaling mechanisms to support recovery. More specifically IETF CCAMP WG issued some RFCs:

- [RFC4872] (end-to-end recovery in GMPLS networks) specifies signaling mechanisms to support different end-to-end protection and hard/soft rerouting schemes.
- [RFC4873] (segment recovery in GMPLS networks) specifies signaling mechanisms to support different protection and hard/soft rerouting schemes for segments of an LSP.
- [RFC4920] specifies (section 7) an alternate way for supporting end-to-end full rerouting.

However, it has been noticed that the previously mentioned signaling solutions are not applicable in the ASON E-NNI context for the following main reasons:

4.1.1 ASON Identifiers

In ASON E-NNI signaling, SESSION and SENDER_TEMPLATE identifiers change across each E-NNI. Indeed, the RSVP ASSOCIATION object defined in [RFC4872] can be used to associate LSPs across different RSVP sessions, and [RFC4873] relies on such a capability to associate LSPs for two different purposes:

- Recovery;
- Resource sharing.

The RSVP ASSOCIATION object identifies an association using the association type, the association source and an association identifier (scoped by the association type and source).

When dealing with multiple nested recovery domains, the recovery domain within which each association is performed needs to be specified. This is further detailed in section 6.1.1 and section 6.1.2.

4.1.2 Failure notification

[RFC4872] and [RFC4873] rely on PathErr messages, and optionally Notify messages, being sent, respectively, to the ingress node or to the upstream branch node when a transport plane failure is detected.

In ASON networks, it is not always possible for a node detecting a failure to send such a Notify message directly to the ingress or upstream branch node, if one or multiple E-NNI interfaces must be crossed to reach it.

As a consequence failure notification is performed in a chain across domain border nodes or abstract nodes.

4.1.3 Reversion

[RFC4872], [RFC4873] and [RFC4920] do not define reversion for the “full rerouting” case.

4.2 Scope

ITU-T [G.8080] defines recovery mechanisms (protection or hard/soft rerouting) across a "recovery domain"; such a domain may encompass a single or multiple carriers.

Thus, [G.8080] recovery requires some kind of coordination between the call endpoints. **Inter-domain recovery mechanism will kick-off only when a domain could not recover a call segment by itself**, either:

- Because that domain policies chose not to provide recovery for that subnetwork call segment;
- Because that domain tried but failed to recover that subnetwork call segment itself;

Inter-domain recovery techniques must co-exist with intra-domain recovery techniques.

The scope of this Implementation Agreement is to overcome the limitations mentioned in section 4.1. More specifically the solution addresses the following carrier requirements:

- Support of different recovery schemes as per [RFC4427] within ASON architecture. More specifically, the following recovery mechanisms will be supported by this amendment:
    - 1+1 uni-directional protection (local protection switching, without any attempt to coordinate both ends);
    - 1+1 bi-directional protection (the protection switching is coordinated by a transport plane APS);
    - Shared-mesh restoration without extra-traffic; the recovery is initiated by the control plane;

      This amendment does not investigate how to assure that working connections are SRLG diverse or not.
    - Hard-rerouting[1]; the recovery is initiated by the control plane.

  This amendment supports connection setup and administrative rerouting for all above recovery mechanisms. Connection activation and de-activation for 1+1 protection mechanisms is out of scope (no control plane involvement).

[1] This amendment supports hard-rerouting as defined in section 5.1 including both revertive and non-revertive hard-rerouting.

- Support of recovery domains. There are two types of recovery domains: recovery domains corresponding to a routing area and E-NNI scoped recovery domains consisting of multiple E-NNI links between two E-NNI border nodes.
- Hard and Soft (administrative) rerouting control in this environment:
    - Failure reporting and/or Crankback information
- Resource re-use in hard/soft rerouting scheme where the resources allocated on the working path may be reused by the recovery path.
- Reversion where the normal traffic returns to (or remains on) the working path when it has recovered from the failure.

This amendment addresses path protection; link protection is out the scope of this document.[2]

Ethernet protection is for further study.

4.3 Merits to OIF

This amendment to [OIF-E-NNI-sig-02.0] is a key step towards the implementation of an open inter-domain interface that allows offering recovery for end-to-end services. This activity supports the overall mission of the OIF.

4.4 Working Groups

Networking & Operations Working Group
Carrier Working Group
Interoperability Working Group

4.5 Relationship to other Standards Bodies

Whenever possible, this amendment reuses solutions provided by other standards bodies (in particular protocol solutions from IETF CCAMP and ASON Recovery domain architectures from ITU). Adaptation of those solutions to the OIF E-NNI context is done because it is deemed pertinent. As far as possible, endorsement from other standards bodies will be sought.

[2] Path protection vs. link protection is referring to layer-specific entities: for instance, VC-4 vs. multiplex section for the SDH layer.

4.6 Keywords

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

5 Terminology & abbreviations

5.1 Definitions

The terminology in this document follows [G.8080] and [G.8081]. This amendment borrows from [OIF-CWG-CPR-01.0] some terms and definitions.

Routing Area: A routing area is defined by a set of subnetworks, the SNPP links that interconnect them, and the SNPPs representing the ends of the SNPP links exiting that routing area. A routing area may contain smaller routing areas interconnected by SNPP links. The limit of subdivision results in a routing area that contains a subnetwork.

Subnetwork: A topological component used to effect routing of a specific characteristic information. In G.8080, a subnetwork is bounded by subnetwork points.

Control Domain: A type of transport domain where the criterion for membership is the scope of a control plane component that is responsible for the transport resources within the transport domain.

Transport Domain: A transport domain is a set of transport resources that are grouped as a result of some criteria. This grouping is established by operator policies. An example is the G.805 administrative domain.

Routing Control Domain: A type of control domain where the criterion for membership is a common routing capability[3]. It may contain zero or more routing domains.

Protection: In protection, resources between the protection endpoints are established before failure, and connectivity after failure is achieved by switching at the protection end-points.

Recovery: Is a mechanism by which a connection or a connection segment which has failed is recovered. Recovery may be accomplished by protection or hard rerouting mechanisms.

Recovery Domain: A type of routing control domain where the criterion for membership is a common recovery capability. The common recovery capability can be based on protection and/or hard/soft rerouting mechanisms.

Working Connection: The connection over which normal traffic is transported when there is no failure or operator command preventing this.

[3] I.e., everything in scope of what a routing controller can do.

Recovery Connection: The protection or hard/soft rerouting connection used to transport normal traffic when the working connection fails, or when an explicit operator command is issued to switch the traffic.

Rerouting: Connection rerouting is a special case of Call Modification and it refers to modification of the path used for a specific connection of an established call. As such, the connection segment traversing a recovery domain will be modified to use a different connection segment (may not be fully diverse from the original path if resource reuse is possible) within the same recovery domain. It should be noted however that recovery domains may be nested.

Rerouting may be performed by:

- **Hard Rerouting** is in response to a network event causing the failure of an existing connection; the new route becomes the Current Path of the connection. It is a failure recovery function in a recovery domain that attempts to create another connection to the destination at the edge of the recovery domain. This is performed in response to the failure of an existing connection, and in case of non-revertive recovery, the recovery connection replaces the connection that contained the failure.
- **Administrative Rerouting** or **Soft Rerouting** is a rerouting operation in response to a management plane request, generally for administrative purposes. The original connection is not taken out of service until the recovery connection is established.

Nested recovery domains: Recovery domains are types of control domains. As such, they inherit the containment property of control domains. When a hierarchy of recovery domains exists, those domains are contained one in another, and are called nested domains. The following convention is used throughout that document: hierarchical levels are numbered (e.g., N, N+1, N-1…) in such a way that a level is assigned a higher number than the lower levels it contains. A lower level recovery domain (level N) is completely contained within a single higher level recovery domain (level N+1).

Source Node: The node initiating the end-to-end call/connection.

Destination Node: The node terminating the end-to-end call/connection.

The source and destination nodes are defined in relation to the call/connection.

Domain Ingress Node (DIN): The Signaling Controller responsible to setup the call segment across the local recovery domain (i.e. the recovery domain being considered) between domain ingress and domain egress or destination nodes in the transport plane respectively.

Domain Egress Node (DEN): The Signaling Controller terminating the local recovery domain call/connection segment and continuing with an UNI or E-NNI call segment to subsequent node.

Border Node: Transport plane node where at least one UNI or E-NNI link is attached and defined in relation to the transport network resources associated with the domain.

Transit Node: A node that implements a portion of a call/connection segment within a recovery domain. Any failure detected is reported towards the domain ingress and egress nodes.

Repair Node: A transport plane node that is the ingress node of the rerouted connection segment. The Repair Node may be the source node, the recovery domain ingress border node, or a transit node. Domain egress border node as Repair Node is for further study.

Rendezvous Node: A transport plane node that is the egress node of the re-routed connection segment. The connection segment between the Repair Node and the Rendezvous Node is re-routed. The other connection segments are re-used. The Rendezvous Node can be the egress domain border node, a transit node or the destination node. Ingress domain border node as a rendezvous point is for further study.

Nominal Path or Working Path: Path computed for the connection according to its constraints and operator policy. It is assumed that the connection uses this path unless there are failures in the network along the path.

Recovery Path: Connection Path used during a network failure event. This path may be pre-computed or it may be computed after a failure has occurred.

Current Path: Path used at a given moment by the connection. Ideally the current path is the working path. However, after a failure in the network, the current path may be the recovery path.

Resources sharing and re-use of resources: "resources sharing" is used in different standards to express the ability to share common resources between LSPs for sharing:

- recovery resources when protecting disjoint primary working LSPs (E.g. for shared meshed restoration),
- resources between recovery and working LSPs during a hard or soft rerouting process.

To avoid any confusion between these different cases, this amendment will use the term “resource sharing” to refer to the first case; it will use the term "re-use of resources" to refer to the second case when the resources from the working connection are re-used for its corresponding recovery connection over E-NNI link(s).

5.2 Abbreviations

APS	Automatic Protection Switching
ASON	Automatically Switched Optical Network (see [G.8080])
B&R	Bridge and Roll
BwAR	Bridge with Automatic Roll
BwER	Bridge with Explicit Roll
NCC	Network Call Controller (see [G.8081])
CCAMP	Common Control and Measurement Plane (IETF working group)
CoS	Class of Service
DIN	Domain Ingress Node
DEN	Domain Egress Node
E-NNI	External Network-Network Interface (see [G.8080])
eNNI-D	The logical control plane entity that terminates E-NNI signaling in the downstream direction with respect to control plane initiation
eNNI-U	The logical control plane entity that terminates E-NNI signaling in the upstream direction with respect to control plane initiation
GoS	Grade of Service
GMPLS	Generalized MPLS
LSP	Label Switched Path
MPLS	Multiprotocol Label Switching
OTN	Optical Transport Network
RA	Routing Area (see [G.8080])
SCN	Signaling Communication Network (see [G.8081])
SDH	Synchronous Digital Hierarchy
SNPP	Subnetwork Point Pool (see [G.8081])
SRLG	Shared Risk Link Group

UNI	User Network Interface (see [OIF-UNI-02.0], [G.8080])
WTR	Wait To Restore

6 Multi-domain recovery

6.1 Architecture

This amendment specifies a solution that stays as close as possible to GMPLS recovery specifications. Ease of interworking between GMPLS and ASON network domains is therefore sought.

In this amendment, we assume a source routed signaling flow across the Routing Areas and across the routing levels. Hierarchical signaling flow is not in the scope of this document. Please refer to appendix V of ITU-T Rec. G.8080/Y.1304 (06/2006) for further details.

A recovery domain can be scoped:

- Either to a routing area (the recovery domain is congruent with its associated routing area);
- Or to an E-NNI interface (the recovery domain includes all E-NNI links between two nodes).

See Appendix B: ASON architecture and recovery domains for more details.

Path computation of a recovery domain (N) should know the boundaries of its nested recovery domains (N-1), so that it avoids computing a detailed path within those domains: this may prevent successful diverse path computation in a nested recovery domain (if that nested recovery domain selects a recovery mechanism that requires multiple diverse paths). This rule will be implicitly fulfilled if a nested recovery domain is abstracted as an abstract node, or a set of border nodes and abstract links.

The mechanism that path computation needs to discover the boundaries of a nested recovery domain is out of scope for this amendment.

For each of its local links, a Call Controller (NCC) MUST know the recovery domains which they belong to. A link may not belong to any recovery domain. A link may belong to more than one recovery domain. In Figure 1 below:

- E-NNI link 1 is an inter-domain link with respect to RA:A and RA:A.4, it does not belong to any recovery domain;
- E-NNI link 2 belongs to RA:A.4 recovery domain,
- E-NNI link 3 belongs to RA:A recovery domain, and to the recovery domain scoped to the E-NNI interface between node 8 and node 14.

Therefore, when it processes a connection request crossing such links, a Call Controller is able to tell if its role is a Domain Ingress Node (DIN), a Domain

Egress Node (DEN) or a Transit Node in each recovery domain the links belong to.

If I-NNI interfaces are used inside a routing area, the way a request is processed inside the domain and the way the request is passed from the DIN to the DEN is out of the scope of this specification.

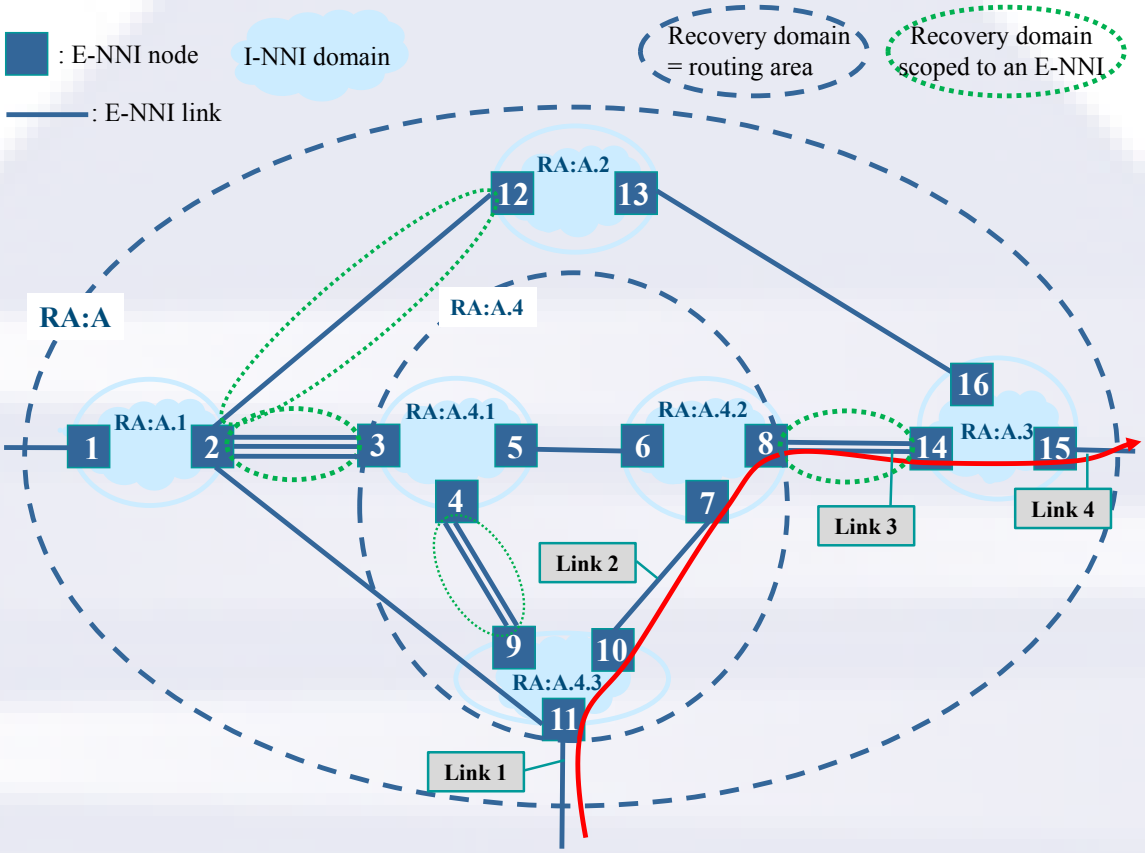


Figure 1: Example of nested recovery domains

6.1.1 Recovery information

A recovery domain may support multiple recovery mechanisms. Under local domain policy, the DIN may assign one or multiple to a given connection.

This section refers to the set of information the DIN needs to specify and to exchange with the DEN as "recovery information set".

In order to handle recovery mechanisms (protection and/or hard/soft rerouting) within a recovery domain, the DIN and the DEN have to agree about the activation of the service across this domain for each call / connection. To achieve this, the following recovery information is required:

One is the **recovery type** information which unambiguously identifies the mechanism used to recover from a data plane failure;

The second one is the **association** information which is needed to bind the recovery connection with the working connection.

Section 6.1.3 requires the specification of a **service level**, private to each nested recovery domain. This implementation agreement refers to it as "domain local service level".

A **recovery domain identifier** is required to identify the recovery domain to which such recovery information set applies. For instance, the DEN of that recovery domain must remove such recovery information set when it forwards a connection request out of its domain.

For a given connection within each recovery domain, such recovery information set is required by each requested recovery mechanism.

From source to destination a connection request that transits through multiple recovery domains, will be tied to as many recovery information sets as recovery mechanisms selected in the domains it had crossed.

6.1.1.1 Additional recovery information for shared-mesh restoration

Within a recovery domain where shared-mesh restoration has been selected, the recovery connection setup signaling message should specify the resources (nodes and links, including abstract ones) used in that domain by the associated working connection.

This information will allow all Call Controllers participating to the setup of the recovery connection in that domain to choose whether or not some resources can be shared between this recovery connection and other recovery connections: this choice is based on domain policies and may rely on the fact that all recovery connections sharing resources are associated to working connections that are resource-disjoint in the local recovery domain.

6.1.2 Nested recovery domains

A connection request that transits through multiple nested recovery domains specifies as many recovery information sets as recovery mechanisms selected at domain entrances it has crossed so far:

- When it enters a recovery domain, a recovery information set is added to the connection request for each recovery mechanism selected by the DIN.
- When it exits a recovery domain, the associated recovery information set(s) is removed from the connection request by the DEN.

A node which belongs to one recovery domain will process (see section 6.2 for signaling details) only the recovery information set(s) relevant to that domain (C.f. node 15 in Figure 1).

A node which belongs to multiple nested recovery domains (each of them contained in a hierarchical manner) will process as many recovery information sets as selected recovery mechanisms in the domains it belongs to.

In Figure 1, the connection is established from node 11 to node 15. The Signaling Controller associated to node 11 is the DIN of the recovery domain RA:A. It is also the DIN of the nested recovery domain RA:A.4. (Note it is also the DIN of the nested I-NNI domain RA:A.4.3). Assuming that at least one recovery mechanism is selected in each RA (according to each recovery domain policies), the Signaling Controller associated to node 11 will process at least two recovery information sets (one relevant to RA:A and useable by the Signaling Controller associated to destination node 15 and one relevant to RA:A.4 and usable by the Signaling Controller associated to node 8).

Similarly, note that the Signaling Controller associated to a node may be the DEN for multiple recovery domains; and therefore may need to process multiple recovery information sets (e.g. node 11 in Figure 1).

6.1.3 Service mapping

Different requests received by the DIN may request different CoS/GoS to be provided. To distinguish between these grades of service, the OIF G-UNI object defined in [OIF-UNI-02.0-RSVP] provides a Service Level sub-object. This sub-object contains a 8-bit value (range 0-255) which references a specific service definition, which may be used by the DIN to identify the specific policy to be applied to the connection.

However, the OIF does not dictate how a carrier should map a service level to a particular recovery mechanism. Even within one carrier network, each recovery domain may implement its own mapping.

The mapping may result in a single or multiple recovery mechanisms combined together over that domain. The mapping depends on the local operator policy and on the recovery capabilities of that domain.

The policy applied at the ingress of a nested domain and the corresponding mapping between the domain local service level and the recovery mechanism may take into account the recovery mechanism already used in the upper level domains; therefore, for each recovery domain, there may be a signaling mechanism for coordination with nested domains. The DIN of each recovery

domain (N) may specify in the signaling message exchanged across its domain, which domain local service level should be provided by its nested recovery domains (N-1); this domain local service level may be different from the one used in the signaling message exchanged across domain (N+1). This domain local service level must not be specified by a node outside of recovery domain (N). This domain local service level is set depending on the recovery mechanism chosen in domain (N), it may change within each nested recovery domain, and may therefore be different from the invariant end-to-end UNI service level.

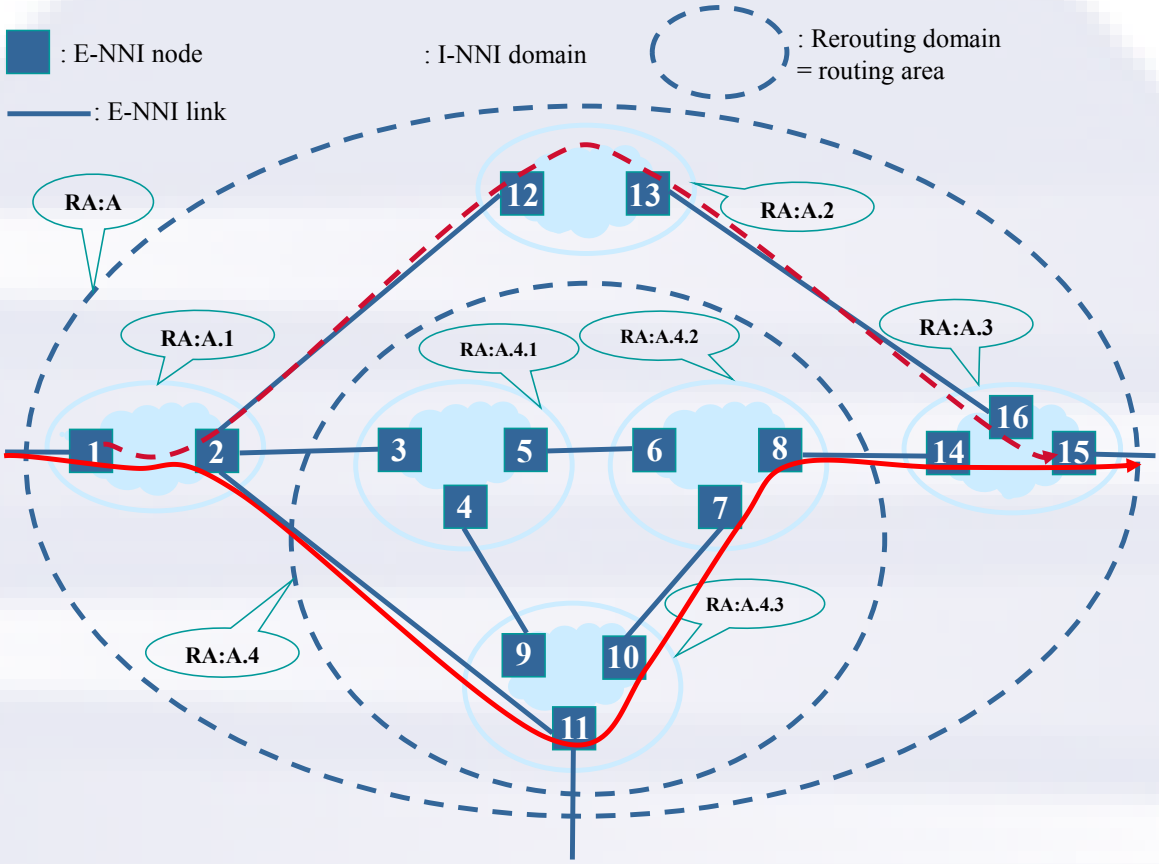


Figure 2: Example of coordination of recovery mechanisms through nested domains

For instance, when considering **Figure 2**:

When the DIN associated with node 1 receives the request with a "high availability" domain local service level, it may decide following the policy of domain RA:A to create a 1+1 protected connection. Two paths are computed over RA:A between node 1 and node 15 as shown by the red lines. The DIN associated with node 1 may therefore decide to change the domain local service level requested to lower nested domains in domain RA:A (i.e. nested domains A.1, A.2, A.3 and A.4), so that:

- such domains will not waste resource in providing a "high availability" service too;
- such domains will not defeat the 1+1 protection mechanism in place in domain RA:A by allowing hard-rerouting mechanisms to recover failures in lower domains.

The DIN associated with node 1 may set the domain local service level to “unprotected”.

6.2 Detailed signaling

6.2.1 Detailed signaling for connection setup

The following steps describe the process for connection handling (see also **Figure 4** and **Figure 5** that respectively illustrate those steps for protection/pre-planned rerouting and hard rerouting cases; both figures are based on Figure 3 reference network):

At connection setup, if the source UNI-N belongs to one or multiple recovery domains, the source node Signaling Controller may add in the connection request the recovery information relevant to each recovery domains:

- The highest recovery domain to be considered is the one containing both the source UNI-N and the destination UNI-N; note that such a recovery domain may not exist, in that case end-to-end recovery is not possible;
- Next, for all other lower recovery domains, as specified above, the Signaling Controller component at the next lower level adds in the request the recovery information relevant to this lower level recovery domain (E.g. RA:A4 in Figure 1).

  This mechanism recurs until the lowest level recovery domain is reached.

Each time a call / connection request is entering a new recovery domain; the DIN adds to the request the recovery information sets that are relevant to this recovery domain (the recovery domain that is traversed by the call from domain border ingress and egress nodes).

Therefore, a call / connection request may specify a list of recovery information sets.

When a call / connection request reaches a DEN, it will process the recovery information sets relevant to its recovery domain:

- associates the working and the recovery connections,
- if necessary performs the recovery actions.

As specified above, if the node is the DEN of multiple recovery domains, the node will process also the recovery information sets relevant to the next upper levels.

When a connection request exits a recovery domain, the DEN removes from the list the recovery information set associated to that recovery domain.

Unless it is the DIN or the DEN of the relevant recovery domain, a node MUST not modify any recovery information set in a message.. Therefore, passing

through a domain (using I-NNI or E-NNI signaling) is completely transparent from outside this domain.

At the destination UNI-N, all remaining recovery information sets are processed by the Signaling Controllers of each level. There should be as many left as the number of recovery mechanisms used in the domains the destination UNI-N belongs to.

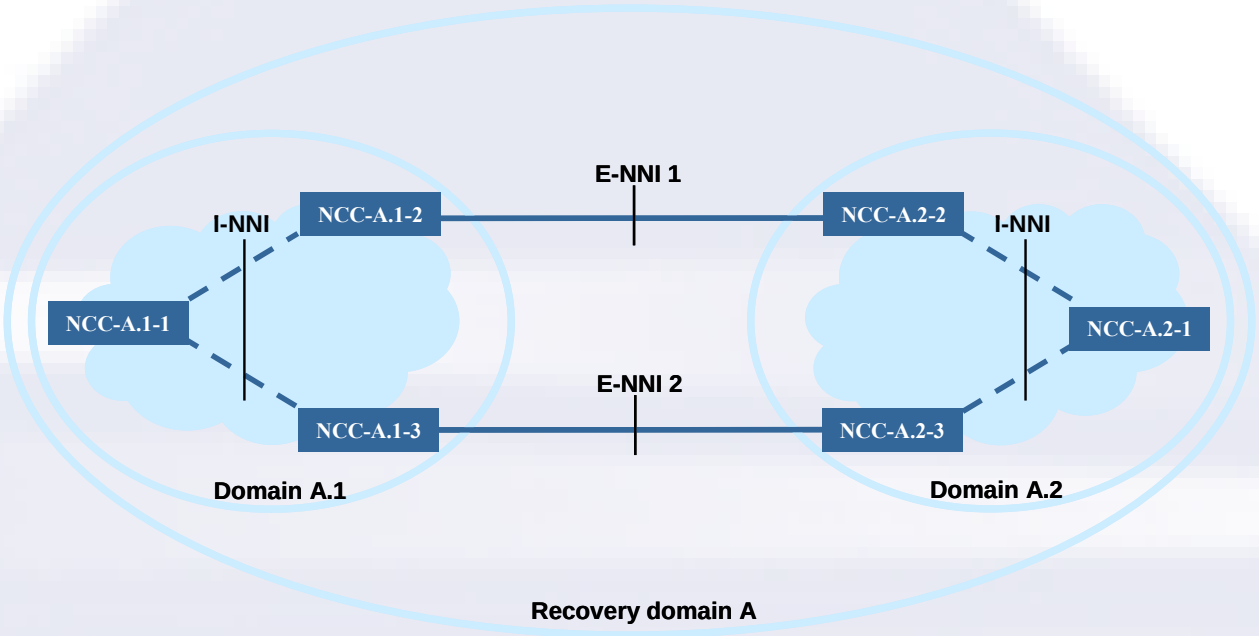


Figure 3: Recovery – Multiple E-NNIs

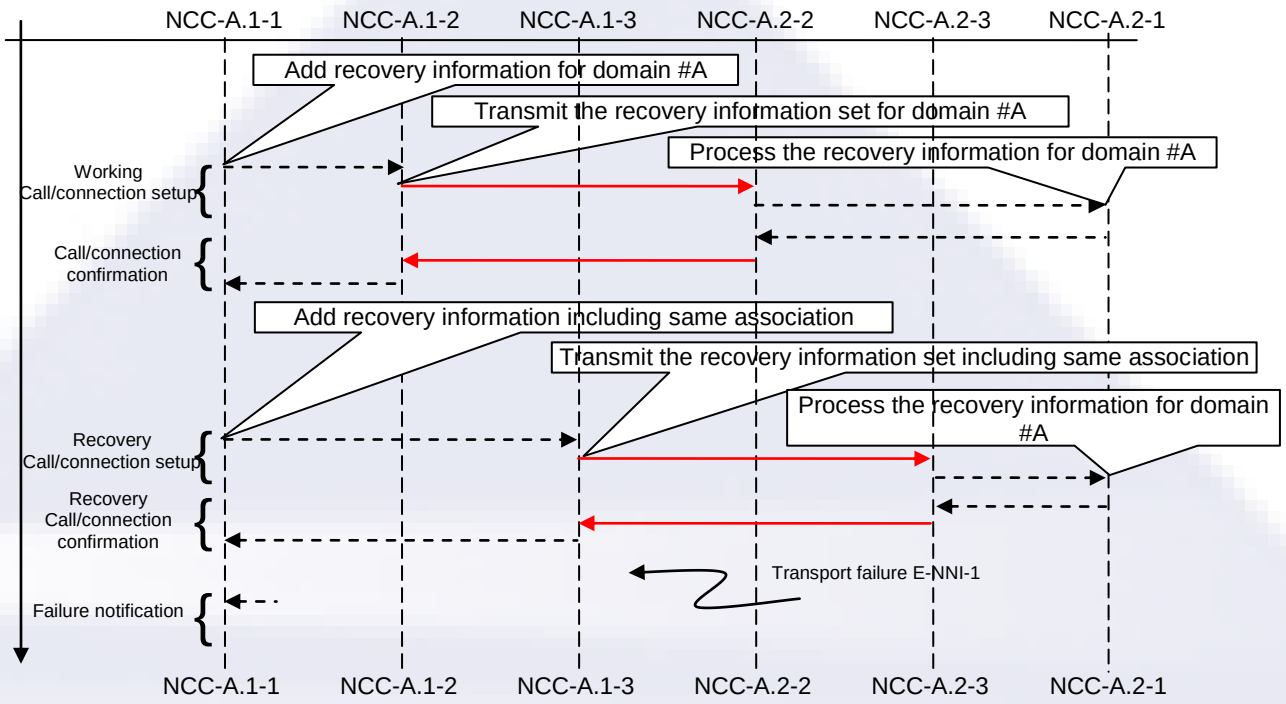


Figure 4: Example of protection or pre-planned rerouting signaling processing

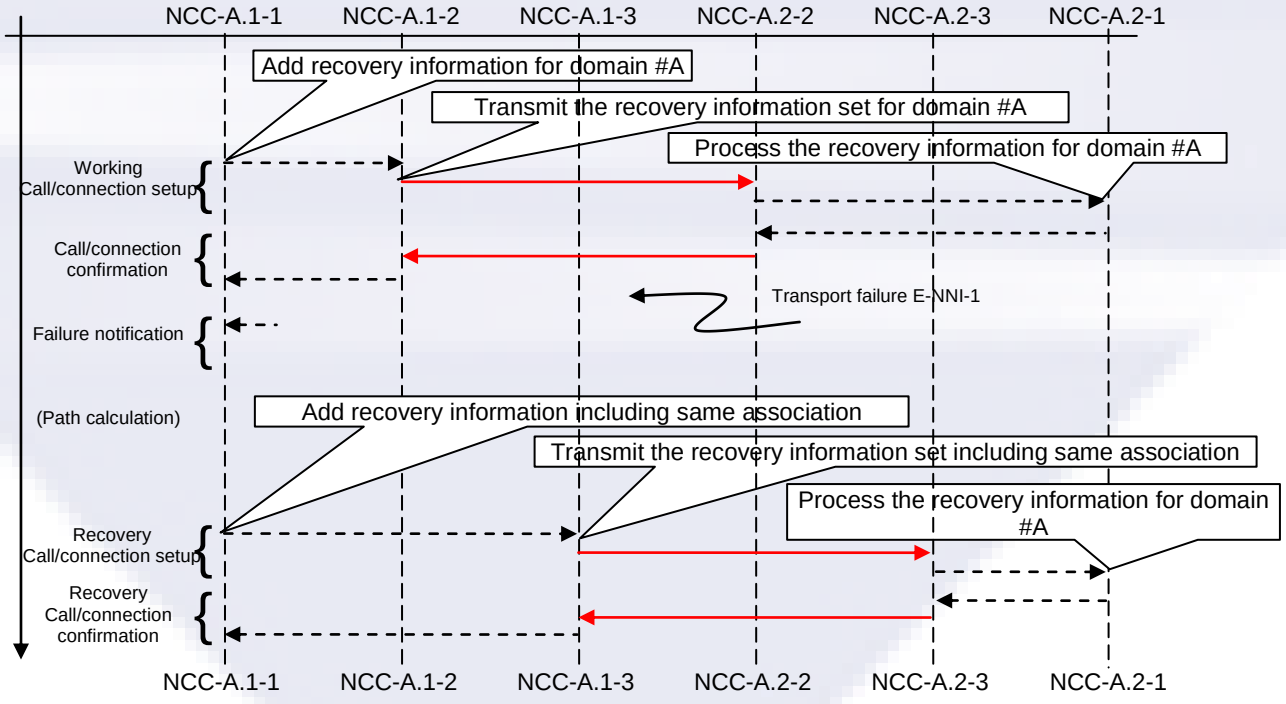


Figure 5: Example of hard rerouting signaling processing

6.2.2 Failure notification

This amendment does not investigate whether failure notification may be provided by the transport plane itself (this may be switching technology specific). If the transport plane can provide failure notification, care must be taken not to trigger current level recovery too soon, while some lower level recovery procedures are in progress.

The remaining text in that section assumes that failure notification is handled by the control plane and is forwarded upstream to the failure.

In case of source node failure, no recovery can be performed; therefore failure notification is not required.

Notification messages should identify the failure and the recovery domain where this failure occurred.

As long as the notification message is forwarded within the same recovery domain (N); this message explicitly identifies the failed resource (node, link, bundle…). Note that if the failure occurs on a link that belongs to a bundle and if all other component links do not have enough available bandwidth to support all impacted connections, the notification message may identify the bundle itself; otherwise it may identify the component link, and the bundle will not be excluded from any path computation.

If the notification message needs to be forwarded at an upper level recovery domain (N+x) then the message may no longer explicitly identify the failed resource: it must identify the resource exposed in the upper recovery domain (N+x). If abstraction is used, notification messages must identify the abstracted resource or the complete lower level recovery domain (N). The latter information may be used to prevent the N+x DIN to re-use this resource or domain in the new path computation.

When a Signaling controller receives a notification message reporting a failure at level (N), it will check whether its role with regard to each reported connection is:

- a DEN. It will forward the notification message to its corresponding DIN;
- a DIN at domain N, see the procedure specified in steps a) b) c) below;
- a DIN at domain N+x, the same procedure as specified in steps a) b) c) below applies in scope of domain N+x;
- a DIN at domain N-x, it must forward the notification message upstream (because the notification message is intended for the DIN of the higher recovery domain (N)).

a) Failure inside a recovery domain of level N (node (other than a DIN) or link failure)

The DIN at level N must be informed of the failure.

The failure notification progresses upstream "hop-by-hop" across E-NNI interfaces of the same level (N) and across lower level recovery domains (N-x).

Under local domain policy, the DIN may first decide to attempt a recovery inside the recovery domain of level N.

If the attempt is successful, no more failure notification is required.

If the attempt is unsuccessful or if the feature is not supported, the Signaling Controller must notify the failure at the DIN at level N+1. If a transport plane node belongs to multiple nested recovery domains, the associated DIN at level N+1 may be hosted together with the associated DIN at level N. In this case, the notification is internally sent to the N+1 DIN. Otherwise, the failure notification progresses upstream "hop-by-hop" across E-NNI interfaces at N+1 level and across recovery domains between these E-NNI interfaces.

This recurs up to the originating Signaling Controller.

b) Failure at the border of a recovery domain of level N (border nodes failure or inter-domain link failure (link entering or exiting that domain))

The DIN at level N+1 must be informed of the failure. The failure notification progresses upstream "hop-by-hop" across E-NNI interfaces at N+1 level and across recovery domains of the same level (N).

> For instance, in **Figure 6**, a failure occurs on the E-NNI link between node 10 and node 7. This link is located at the border of two (N) recovery domains: A.4.3 and A.4.2.
>
> The failure impacts a connection that was signaled from node 3 to node 8 in recovery domain A.4. The failure notification will progress upstream, starting from node 10, across two (N) recovery domains: A.4.3 and A.4.1, and across one (N+1) E-NNI interface: between node 9 and 4, until it reaches the DIN of the (N+1) recovery domain A.4 (node 3).

Under local domain policy, the DIN at level N+1 may first decide to attempt a recovery inside the recovery domain of level N+1.

If the attempt is successful no more failure notification is required.

If the attempt is unsuccessful or if the feature is not supported, the Signaling Controller must notify the failure at the DIN at level N+2. If a transport plane node belongs to multiple nested recovery domains, the associated DIN at level N+2 may be hosted together with the associated DIN at level N+1. In this case, the notification is internally sent to the N+2 DIN. Otherwise, the failure

notification progresses upstream "hop-by-hop" across E-NNI interfaces at N+2 and across recovery domains between these E-NNI interfaces.

This recurs up to the originating Signaling Controller.

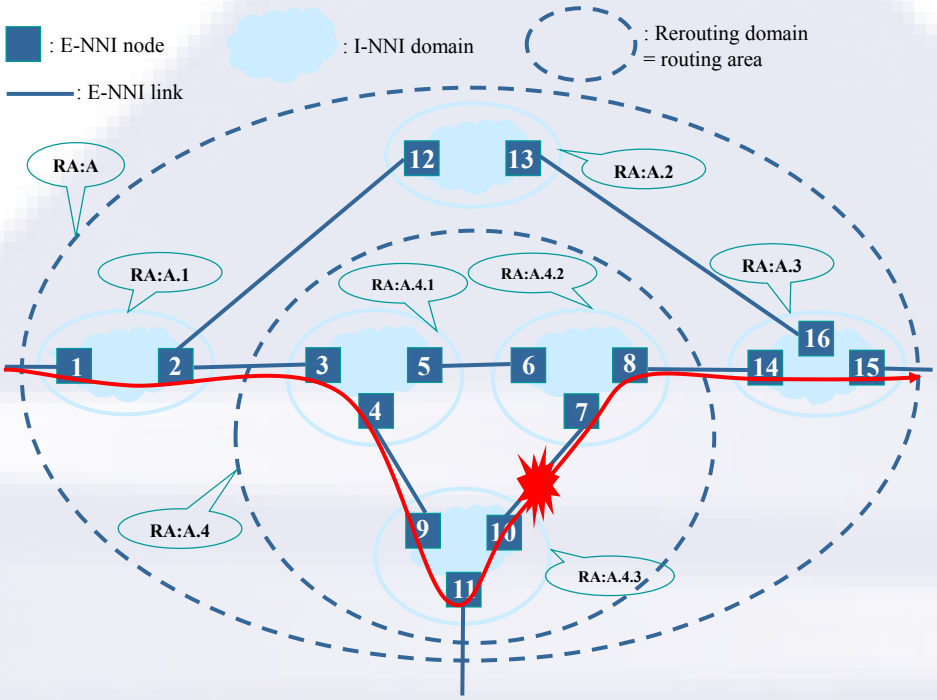


Figure 6: Domain entry/exit link failure

c) Destination node fails

No recovery can be performed. However, the failure notification will be sent upstream to the source call controller (across E-NNI of the global domain). If the source node is aware of the destination node failure, then the source call controller should not initiate a recovery process.

6.2.2.1 Failure identification for domain abstractions

Lower level areas may contain resources that are abstracted and thus not visible to an upper level area.

Lower level areas may use namespaces that are not visible to an upper level area.

Since the E-NNI signaling must contain information that is visible to the E-NNI routing process, the error information must be provided using the names from upper area. This handling is defined for the following modes of domain abstraction.

a) Exposed Interior

An exposed interior does not have any abstraction in the topology provided to the upper area. Therefore, the upper area topology has a link that exactly matches the link that has faulted in the lower area.

Since the information provided in the upper area regarding the exposed interior may use different addressing than is used in the lower area, the edge nodes for the lower area need to be able to perform the translation of lower area link identifiers into the <NodeID, ifIndex> information carried in the error information provided across the E-NNI link.

b) Exposed Borders

An exposed border representation does not have any abstraction in the topology for the border nodes, but it does have abstraction for the links and nodes interior to the domain. Since a failure for non-border links/nodes does not have a directly corresponding entity in the upper level that should be avoided by future path computation, the DIN border node needs to identify what to provide in the failure notification.

c) Abstract Node

An abstract node's interior does not contain any information that needs to be externally viewed when performing crankback.

The use of topology abstraction (in cases b and c) to represent a domain in a hierarchy of domains allows to reduce the amount of topological information exchanged between those domains. Due to the distortion between the real topology and the abstracted one, a loss of accuracy exists when an internal event (E.g. failure, failure clearance…) has to be reported outside of a particular abstracted domain.

If the DIN needs to report upstream an internal failure, it can either identify:

- The whole domain (therefore, recovery in the upper level will avoid the domain);
  If any possible path still exists through the excluded domain, this choice may lead to a non-optimized recovery path.
- Some abstracted resource(s) or some E-NNI link(s) (therefore, recovery in the upper level may go through the same domain again).
  For that second case, the lack of accurate information may lead to excessive delay in the hard rerouting process especially when resources exhaustion appears (one example is provided in Appendix A: Failure notification and resources exhaustion inside a domain). To avoid this

situation, it is required that the DIN includes in the failure notification the right information to avoid several unsuccessful re-attempts.

- For instance, in the exposed borders case, since the original explicit route received by the ingress node references the abstract link that was used to identify a path should go through the domain, it may be enough to identify this abstract link in the failure notification. Consequently, the failure notification provided across the E-NNI link should contain the <NodeID, ifIndex> for the downstream interface associated with the abstract link on the ingress border node.
  However, this may still result in crankback, if a path computation entity in the upper domain simply prunes that abstract link and if another abstract links path exists between the same ingress and egress border nodes.

- For instance, in the abstract node case, it may be enough to avoid the entrance link (or exit link) to the abstract node. Therefore, the ingress node should respond with failure information containing the ingress (or egress) link to the Abstract node that was used by the failed connection.

6.2.2.2 Multiple notifications per failure

In a data-plane with bi-directional connections and upstream OAM (i.e. RDI[4]), it is possible to limit the generation of notifications to just the node immediately upstream[5] from the fault, or to have both the upstream and downstream[6] nodes send notifications towards the DIN.

When the recovery domain DIN receives multiple notifications, it needs to determine if it needs to react to latter ones. This is especially important to keep the DIN from interrupting the recovery operation being performed as a result of receiving the first notification.

Since the notifications are actually coming from domains which may be represented in the E-NNI by abstract topologies, a domain may employ different strategies to determine what information should be placed to identify a failure. This has the potential to make it impossible for the DIN to correlate different notifications reporting the failure of the same link.

To handle this case, a mechanism needs to be provided to reduce the number of notifications being sent for a single fault. This is done by preventing the

[4] such as SONET/SDH Remote Defect Indication
[5] Upstream must be understood from a control plane signaling flow standpoint.
[6] Upstream and downstream must be understood from a control plane signaling flow standpoint.

downstream node from generating a notification. For this reason, nodes that have the ability to generate notifications when downstream from a fault MUST have a configuration option to inhibit sending these notifications.

6.2.3 Detailed signaling for connection recovery

In the hierarchy of recovery domains, it is required that the recovery mechanism occurs at the lowest recovery domain first.

a) Protection

In case of protection, the DIN and DEN must have bound the working connection with its protecting connection prior to the failure. Protection switching is performed at the protection end points to which the DIN and DEN are associated.

b) Hard rerouting

In case of hard rerouting, when failure occurs inside the (N) recovery domain under consideration, the nodes adjacent to the failure detect and report the failure to the DIN. I-NNI failure reporting is out the scope of this document, but since recovery may span multiple domains, it is assumed that failure reporting is supported by I-NNI domains.

The DIN initiates a recovery connection set up request to its associated Routing Controller with the same connection end-points at the edge of the (N) recovery domain than previously used and possibly the exclusion constraint. Upon receiving the computed path, the Signaling Controller initiates the recovery connection setup.

c) Recovery escalation

A recovery escalation process must be started if the recovery is not possible inside the (N) recovery domain under consideration (E.g. multiple failures, lack of resources, node isolation, egress border node failure…) or if the egress inter-domain link or the ingress border node of the downstream domain fails.

The DIN is in charge of starting the recovery escalation. Once the impossibility to recover the failure is detected by the DIN (the way the DIN detects this impossibility is out the scope of this document), it must report the recovery failure to the DIN at level N+1:

- If the DIN at level N is the DIN at level N+1 too (the transport plane node belongs to multiple nested recovery domains), then the notification is internally sent to the N+1 DIN;

- Otherwise, the DIN at level N sends a notification to the DEN of the upstream domain at level N. The upstream DEN must forward the message to its corresponding DIN. The message progresses from routing domain to routing domain of equal levels up to the DIN that hosts the (N+1) upper level recovery domain Signaling Controller.

The upper level DIN applies the recovery mechanism relevant to its recovery domain and performs step a) or b). If the recovery fails, the step c) occurs once again (until the global level recovery domain is reached).

6.2.4 Re-use of resources during hard or soft rerouting

It may be desirable for the recovery connection to reuse the resources allocated to the failed/rerouted connection in upstream or downstream domains.

Resource re-use avoids the allocation of distinct resources (E.g. on an unaffected E-NNI link re-used in the end-to-end path) for both the failed connection and the recovery one. This may be an important requirement considering that the failed connection may not be released, for future reversion purpose. However, at the time such a reversion occurs, it may not be hitless, since the resources assigned to the recovery connection may have to be disconnected first, before being re-assigned to the original connection. If hitless reversion is desired, then:

- Either the recovery connection will have to be setup across disjoint resources (no resource re-use).
- Or bridge and roll is implemented at bridge and merge nodes (repair and rendezvous nodes).

Other resource re-use cases are for further study: for instance one issue is that the recovery connection may enter a domain through a different point that the failed/rerouted connection, making resource re-use hard to achieve within that domain.

7 Abstract Messages and Attributes

7.1 Abstract Messages

E-NNI 2.0 Signaling IA Connection Setup abstract messages are used to support the setup of working and recovery connections. This amendment defines additional attributes for Connection Setup abstract messages.

E-NNI 2.0 Signaling IA Connection Notification abstract message is used to support failure notification. This amendment defines additional attributes for the Connection Notification abstract message.

This amendment defines two additional abstract messages to support the activation of recovery and working connections (respectively for recovery and reversion purposes).

Abstract Messages		Reference
Activation Messages	ActivationRequest	7.1.3.1
	ActivationIndication	7.1.3.2

This amendment defines two additional abstract messages to support permanently moving a working or recovery connection.
The Make Persistent abstract message allows an operator to make permanent the rerouted connection – working or recovery connection - after a hard or soft rerouting. It is used for hard or soft rerouting of any connection belonging to any recovery mechanisms..
It is also used for the last step of the non-revertive hard-rerouting recovery mechanism (see section 9.5).

Abstract Messages		Reference
Make Persistent Messages	MakePersistentRequest	7.1.4.1
	MakePersistentIndication	7.1.4.2

7.1.1 Connection Setup Messages

7.1.1.1 Connection Setup Request

Table 1 lists the different attributes added by this amendment to the Connection Setup Request message:

Message: ConnectionSetupRequest Scope: Global Direction: eNNI-U → eNNI-D			
Attribute	**Type**	**Scope**	**Call/Conn**
Recovery	Optional	Recovery domain	Connection

Table 1: Connection Setup Request abstract message

7.1.1.2 Connection Setup Indication

Table 2 lists the different attributes added by this amendment to the Connection Setup Indication message:

Message: ConnectionSetupIndication Scope: Global Direction: eNNI-U → eNNI-D			
Attribute	**Type**	**Scope**	**Call/Conn**
Recovery	Optional	Recovery domain	Connection

Table 2: Connection Setup Indication abstract message

7.1.2 Connection Notification Messages

7.1.2.1 Connection Notification

This message is used for the following purposes:

- Failure notification;
- End-of-failure notification;
- In shared-mesh restoration, when a recovery connection has been activated to recover one working connection, to notify all other recovery connections that share some common resources with the recovery connection that has just been activated (i.e., whose associated working connections are no longer protected). In the same way, when a recovery connection has been deactivated, to notify all other recovery connections that share some common resources that associated working connections are protected again.

This amendment does not add any new attributes to the Connection Notification message

Message: ConnectionNotification Scope: Local Direction: eNNI-D → eNNI-U or eNNI-U → eNNI-D			
Attribute	**Type**	**Scope**	**Call/Conn**
Recovery	Optional	Recovery domain	Connection

Table 3: Connection Notification abstract message

7.1.3 Connection Activation Messages

7.1.3.1 Connection Activation Request

The Connection Activation Request message is used by the eNNI-U to transmit a recovery or a working connection activation request towards the eNNI-D.

This message is used to activate the recovery connection upon notification of a failure impacting the working connection.
This message is used to activate the working connection upon notification that the failure impacting the working connection has been cleared.

In the case of hard-rerouting, the recovery connection is established after failure notification. There exist two alternatives:

1. If the working connection is failed (e.g., a failure notification has been received), it is possible to optimize the signaling exchange for hard rerouting: the (recovery) connection setup request and indication messages perform both the establishment and activation of the recovery connection;
2. If the soft rerouting is triggered by a user switchover request, in order to minimize traffic impact, a two-steps signaling exchange may be preferred: recovery connection establishment, then activation.

Table 4 shows the abstract message and its content.

Message: ConnectionActivationRequest Scope: Global Direction: eNNI-U → eNNI-D			
Attribute	**Type**	**Scope**	**Call/Conn**
Initiating NCC PC ID	Mandatory	Local	Connection
Terminating NCC PC ID	Mandatory	Local	Connection
Connection Name	Mandatory	Local	Connection
Call Name	Mandatory	Global	Call
SNP ID	Optional	Local	Connection
Recovery	Optional	Recovery domain	Connection

Table 4: Connection Activation Request abstract message

7.1.3.2 Connection Activation Indication

The Connection Activation Indication message is used by the eNNI-D to respond to a recovery or a working connection activation request towards the eNNI-U. That response may be negative in case of activation failure.

Table 5 shows the abstract message and its content.

Message: ConnectionActivationIndication Scope: Global Direction: eNNI-D → eNNI-U			
Attribute	**Type**	**Scope**	**Call/Conn**
Initiating NCC PC ID	Mandatory	Local	Connection
Terminating NCC PC ID	Mandatory	Local	Connection
Connection Name	Mandatory	Local	Connection
Call Name	Mandatory	Global	Call
Connection Status	Mandatory	Local	Connection
Error Code	Optional	Global/ Local	Connection
Recovery	Optional	Recovery domain	Connection

Table 5: Connection Activation Indication abstract message

7.1.4 Make Persistent Messages

7.1.4.1 Make Persistent Request

The connection Make Persistent request message is used by the eNNI-U to transmit a recovery or a working connection make persistent request towards the eNNI-D.

Table 6 shows the abstract message and its content.

Message: MakePersistentRequest Scope: Global Direction: eNNI-U → eNNI-D			
Attribute	**Type**	**Scope**	**Call/Conn**
Initiating NCC PC ID	Mandatory	Local	Connection
Terminating NCC PC ID	Mandatory	Local	Connection
Connection Name	Mandatory	Local	Connection
Call Name	Mandatory	Global	Call
SNP ID	Optional	Local	Connection
Recovery	Optional	Recovery domain	Connection

Table 6: Make Persistent Request abstract message

7.1.4.2 Make Persistent Indication

The connection Make Persistent indication message is used by the eNNI-D to respond to a recovery or a working connection make persistent request towards the eNNI-U.

Table 7 shows the abstract message and its content.

Message: MakePersistentIndication Scope: Global Direction: eNNI-D → eNNI-U			
Attribute	**Type**	**Scope**	**Call/Conn**
Initiating NCC PC ID	Mandatory	Local	Connection
Terminating NCC PC ID	Mandatory	Local	Connection
Connection Name	Mandatory	Local	Connection
Call Name	Mandatory	Global	Call
SNP ID	Optional	Local	Connection
Error Code	Optional	Global/ Local	Connection
Recovery	Optional	Recovery domain	Connection

Table 7: Make Persistent Indication abstract message

7.2 Abstract Attributes

For alignment with [G.7713], this amendment defines a single abstract attribute: the Recovery attribute. Its purpose is to include all recovery-related information, detailed in the following sections.

7.2.1 Recovery mechanism type

This attribute specifies the mechanism or combination of mechanisms that have been selected to support recovery within the recovery domain. This amendment considers (see section 4.2):

- 1+1 uni-directional protection;
- 1+1 bi-directional protection;
- Hard rerouting
- Soft-rerouting
- Shared-mesh restoration without extra-traffic;

7.2.2 Recovery association

This attribute allows associating a recovery connection with the working or recovery connection for which it provides recovery.

7.2.3 Connection purpose:

This attribute is used to distinguish the purpose of the connection: it specifies whether a connection is a recovery connection or a working connection.

7.2.4 Recovery domain identifier

This attribute identifies a recovery domain:

- If the recovery domain is congruent to a routing area, the recovery domain identifier is the routing area ID;
- If the recovery domain is scoped to an E-NNI interface, the recovery domain identifier is an identifier assigned by the eNNI-U signaling controller ID.

7.2.5 Domain local service level

This attribute is introduced in section 6.1.3. It specifies the service level that will be specified in the Connection Setup message of connections (working and recovery) signaled across a recovery domain. This service level is chosen every time a connection enters a recovery domain (N) by the DIN of this recovery domain, based on domain policies and the recovery mechanism selected for this recovery domain. The domain local service level (N) will be used by the DIN of every nested recovery domains (N-1), along with each recovery domain (N-1) policies, to select a recovery mechanism.

7.2.6 Working connection route (for shared-mesh restoration)

This attribute is introduced in section 6.1.1.1. It is specified in the connection setup message of a recovery connection. Its purpose is to provide the route followed by the associated working connection.

8 Control and transport planes coordination during switching

In connection-oriented technologies, three different approaches exist for moving traffic from the working path to the recovery path. Which approach is chosen is based on:

- what mechanisms are supported by the data-plane,
- the amount of time that is acceptable for traffic outage when non-complete failures occur (e.g. unidirectional faults).

There is no requirement that the source and destination nodes in a recovery domain use the same data plane approach for moving traffic. When the approaches used for the source and destination nodes are not the same, the amount of traffic outage that occurs is the only behavior that is impacted.

The approaches are discussed in the three following subsections. **Table 8** summarizes which approach(es) can be used for each recovery mechanism.

	Cross-connect change (XC change)	Bridge and Roll (B&R)	
		Bridge with Automatic Roll (BwAR)	Bridge with Explicit Roll (BwER)
1+1 uni-directional protection	Not applicable	Required	Not applicable
1+1 bi-directional protection	Not applicable	Supported	
Shared-mesh restoration	Supported	Supported	
Hard-rerouting (revertive)	Supported	Supported	
Hard-rerouting (non revertive)	Supported	Supported	

Table 8 : Recovery mechanisms and control and transport planes coordination approaches

8.1 Cross-connect change (XC change)

This approach is perhaps the simplest to implement, as it only requires the Repair Node and Rendezvous Node to trigger a cross connect deletion and cross connect addition.

Note that this approach does not support 1+1 protection.

8.2 Bridge and Roll (B&R)

The bridge function places a copy of the normal traffic signal onto a recovery connection. The roll function controls the selector position.

8.2.1 Bridge with Automatic Roll (BwAR)

This approach has perhaps the least amount of impact on any traffic still being carried on the working path (e.g. in the case of a uni-directional fault). The roll action is triggered by the highest priority:

- local condition (transport plane signal fail or signal degrade),
- state (Wait-To-Restore),
- Or externally initiated command (operator command).

Note that this approach is required for 1+1 uni-directional protection (but may also be used for other recovery mechanisms such as 1+1 bi-directional protection if an attempt is made to coordinate the two ends so that both have the same bridge and selector settings).

8.2.2 Bridge with Explicit Roll (BwER)

With this approach, the roll action is triggered by the highest priority:

- control plane request (indeed, the reason to introduce BwER is that, in case the data plane does not support automatic roll, a control plane protocol mechanism is needed to trigger the roll action),
- Or externally initiated command (operator command).

This approach has a similar traffic impact as Bridge with automatic Roll for shared-mesh restoration and hard-rerouting; the transition to the new path in some situation would have lower traffic impact as the change to the path is occurring in advance of a data plane failure (e.g. revertive or non-revertive soft-rerouting).

The traffic impact is however less than a Cross-connect change.

Compared to the Bridge with automatic Roll, a larger amount of coordination is needed between the DIN and DEN.

Note that this approach does not support 1+1 uni-directional protection.

8.3 Transport plane switching during recovery connection setup

Note that for hard-rerouting mechanisms, recovery connection setup and recovery connection activation (section 8.4) can be combined for faster failure recovery.

Figure 7 shows the different stages of transport plane switching during the setup of the recovery connection.

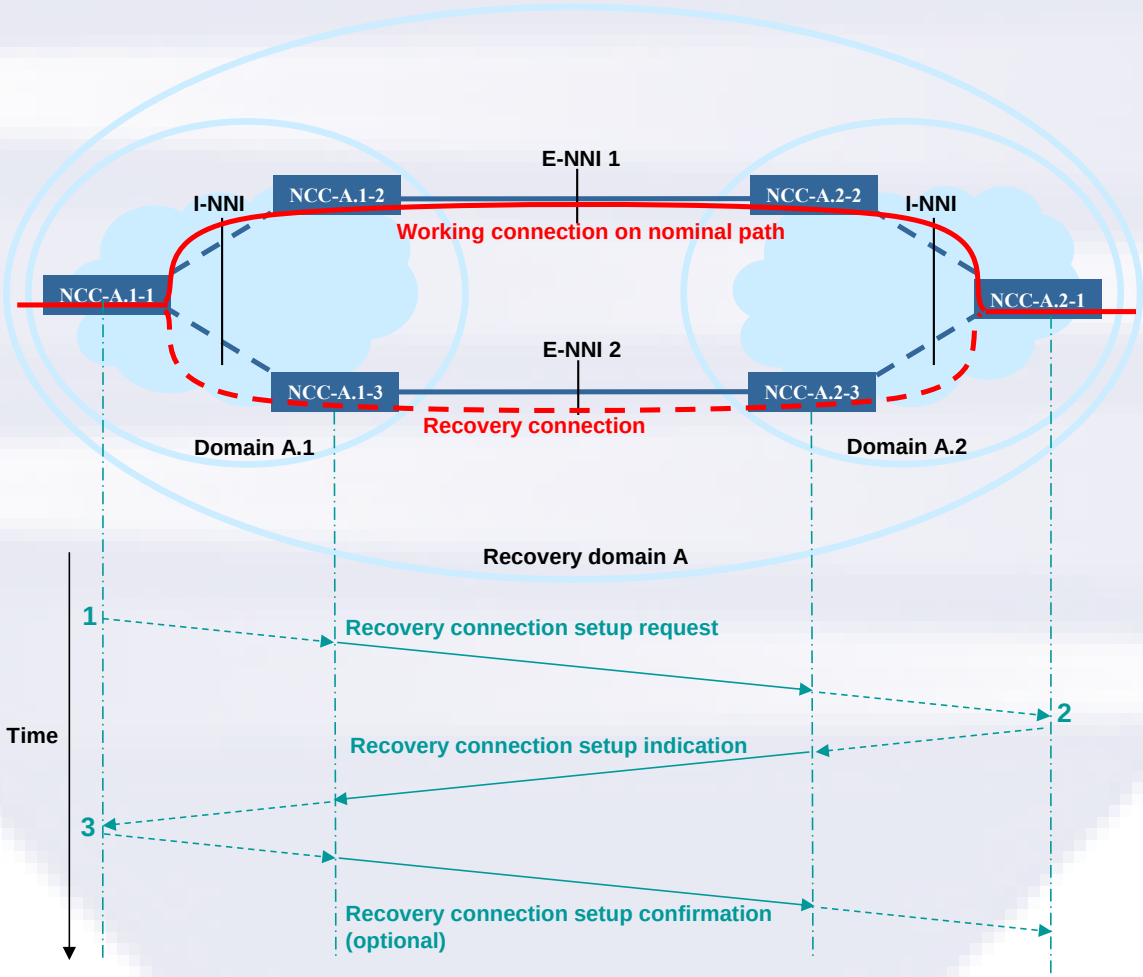


Figure 7: Transport plane switching during recovery connection setup

The successive stages are as follows (the numbers match those of the timeline above):

1. Establish recovery connection (it may be triggered before or after reception of a failure notification or a user-triggered switchover). Setup bridge (if applicable)
    - For B&R, TX to both
    - For XC change, do nothing at this stage
2. If recovery connection accepted, setup bridge (if applicable).
    - For B&R, TX to both
    - For XC change, do nothing at this stage
3. Optional confirmation stage – no cross-connect change

Note that the recovery connection is fully cross-connected at intermediate nodes for all recovery mechanisms (1+1 protection, revertive and non-revertive hard-rerouting) but shared-mesh restoration.

8.4 Transport plane switching during recovery connection activation

Figure 8 shows the different stages of transport plane switching during the activation of the recovery connection.

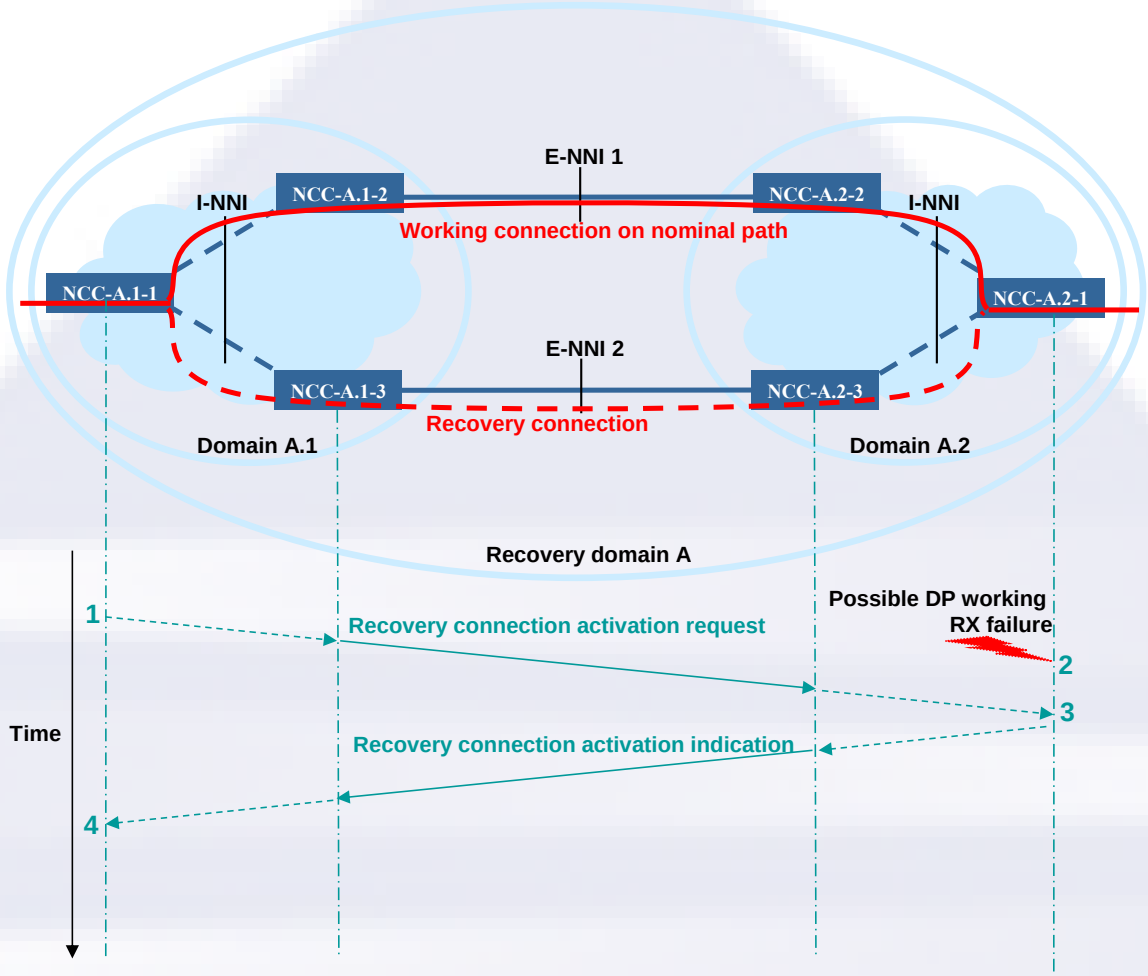


Figure 8: Transport plane switching during recovery connection activation

The successive stages are as follows (the numbers match those of the timeline above):

1. A failure notification or a user-triggered switchover has been received. Activate the recovery connection
    - For B&R, RX to recovery connection
    - For XC change, TX/RX set to recovery connection
2. Optimization if a data plane failure has been detected on the working connection, perform step 3 immediately (without receipt of a Recovery connection Activation Request).

3. Activate the recovery connection
    - For B&R, RX to recovery connection
    - For XC change, TX/RX set to recovery connection

    Note that for shared-mesh restoration recovery mechanism, the recovery connection is fully cross-connected at intermediate nodes.

8.5 Transport plane switching during working connection activation (reversion)

Figure 9 shows the different stages of transport plane switching during the activation of the working connection.

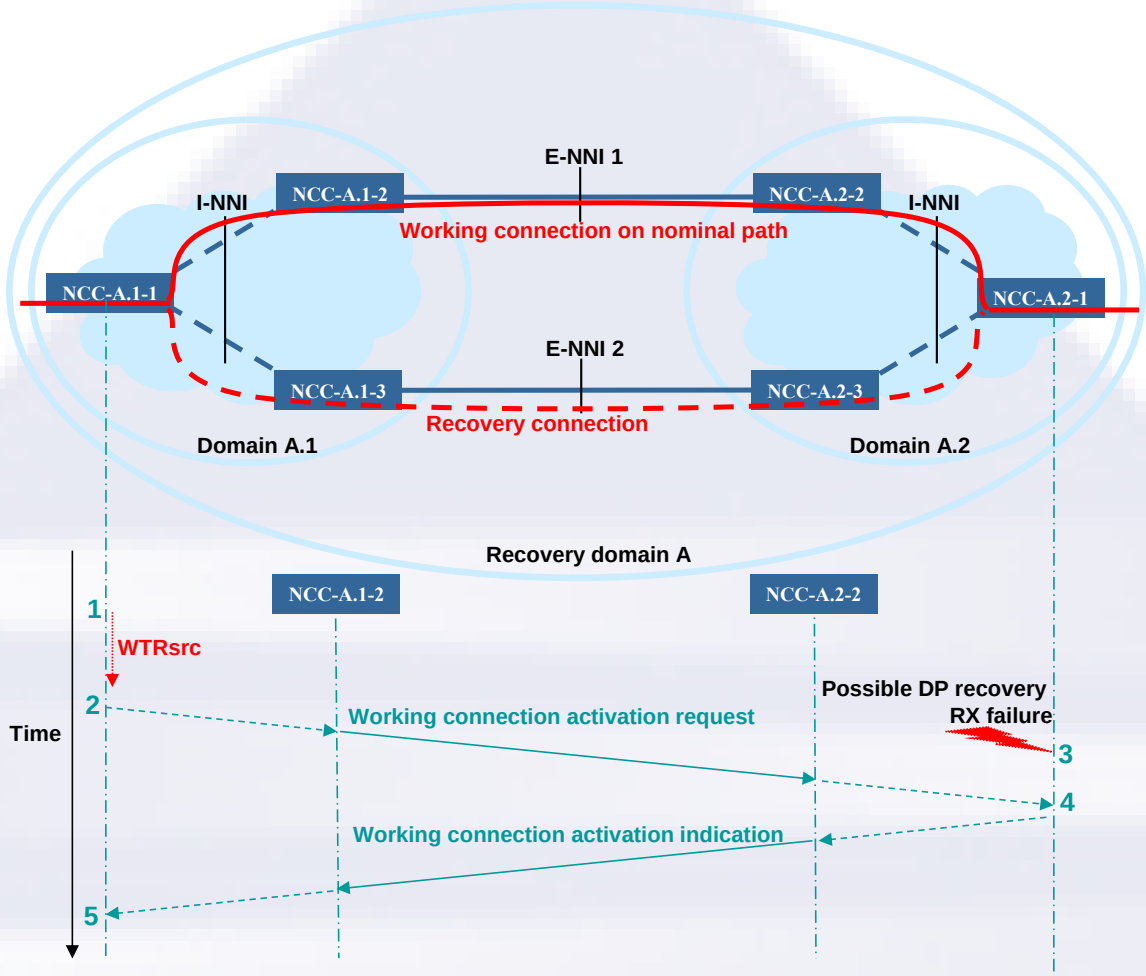


Figure 9: Transport plane switching during working connection activation

The successive stages are as follows (original conditions: traffic is flowing on the recovery path; if Bridge and Roll is available, traffic is TX on both paths but RX is set to recovery path):

1. All the failures have been cleared. Start the Wait to restore (WTR) timer at the source. If 1+1 uni-directional protection, start WTR at the source and destination independently.
2. Activate the working connection
   - For B&R, RX to working
   - For XC change, TX/RX set to working
3. Optimization if a data plane failure is detected on the recovery connection, perform step 4 immediately (without receipt of a Working connection Activation Request).

4. Activate the working connection
    - For B&R, RX set to working
    - For XC change, TX/RX set to working

9 Detailed signaling for connection setup and recovery

9.1 Combination of protection and hard rerouting and soft rerouting

Table 9 shows the different protection, hard rerouting and soft rerouting combination that can be achieved. The first column lists the different primary states. The other columns show whether protection or hard rerouting or soft rerouting can be combined with each primary state.

Although [G.8080] models such combinations using nested recovery domains, this implementation agreement supports them using a single recovery domain (i.e. a single recovery attribute information set is used during the signaling).

A cross (X) means that the combination is not supported.

Primary state	**Can be combined with:**			
	1+1 protection	**Shared-mesh restoration**	**Hard-rerouting (revertive or non-revertive)**	**Soft rerouting**
1+1 protection	X	X	The working and protecting legs can be rerouted independently. See 9.1.2 Or one can be rerouted only after both had failed. See 9.1.2	The working or protecting connection can be soft rerouted. See 9.1.1
Shared-mesh restoration	X	X	The working connection can be rerouted after both working and recovery had failed.	The working or recovery connection can be soft rerouted.

Hard-rerouting (non-revertive)	X	X	If the recovery connection fails, it will be torndown and another attempt can be made.	Only the working connection can be soft rerouted [7].
Revertive Hard-rerouting				Only the working connection can be soft rerouted. The recovery connection must be released or made persistent before it can be soft rerouted.
Unprotected	X	X	X	The connection can be soft rerouted
Permanent soft rerouting	X	X	The connection can be rerouted after it has been made persistent	The connection can be soft rerouted after it has been made persistent
Temporary soft rerouting	X	X	To be restorable, the soft rerouted connection must be released or made persistent	To be soft rerouted, the soft rerouted connection must be released or made persistent

Table 9: Combination protection and hard rerouting and soft rerouting

[7] There is a tiny window when both a working and recovery connections exist (rerouting in progress).

9.1.1 Combined 1+1 protection and soft-rerouting

Combination of protection and soft-rerouting mechanisms allows carriers to perform maintenance activities on the working or protecting LSPs while maintaining the 1+1 protection.

This is achieved by supporting up to four related connections that have resources reserved but of which at most two are instantiated end-to-end in the data plane.

9.1.2 Combined 1+1 protection and hard-rerouting

In order to provide higher reliability, some service levels may combine restoration and protection.

The DIN of a recovery domain supporting combined 1+1 protection and hard-rerouting may support different ways of combining 1+1 protection and hard-rerouting, and may choose one of them upon failure notification based on local policies.

For instance:

- Always on protection offers the assurance of fast protection even after successive failures by restoring the failed leg(s) of a protected service (as long as resources are available).

  When any one of the working or protecting connections is under failure condition, the recovery domain DIN triggers restoration of that connection.

  This is achieved by supporting up to four related connections that have resources reserved but of which at most two are instantiated end-to-end in the data plane.

- 2nd level restoration offers protection against dual failures in the case of protected services. It offers the option to restore the connection if both working and protection connections fail.

  The solution to second level restoration can be viewed as a subset of the solution to always on protection. The triggers for restoration differ. Also, for second level restoration, the head end is responsible for selecting a connection that is deactivated during second level restoration.

  This is achieved by supporting up to three related connections that have resources reserved but of which at most two are instantiated end-to-end in the data plane. When both the working and protecting connections are under failure condition, this triggers restoration. This limitation of at most

two instantiated connections implies there is a short window of vulnerability where the 1+1 protection is not available.

9.2 Detailed signaling for 1+1 protection recovery

Figure 10 shows the abstract messages exchange during connection setup, then connection recovery, in a recovery domain when the selected recovery mechanism is 1+1 protection. It is based on Figure 3 reference network.

Figure 10 assumes a bi-directional failure. In case of uni-directional failure, the automatic roll kicks in on one side only.

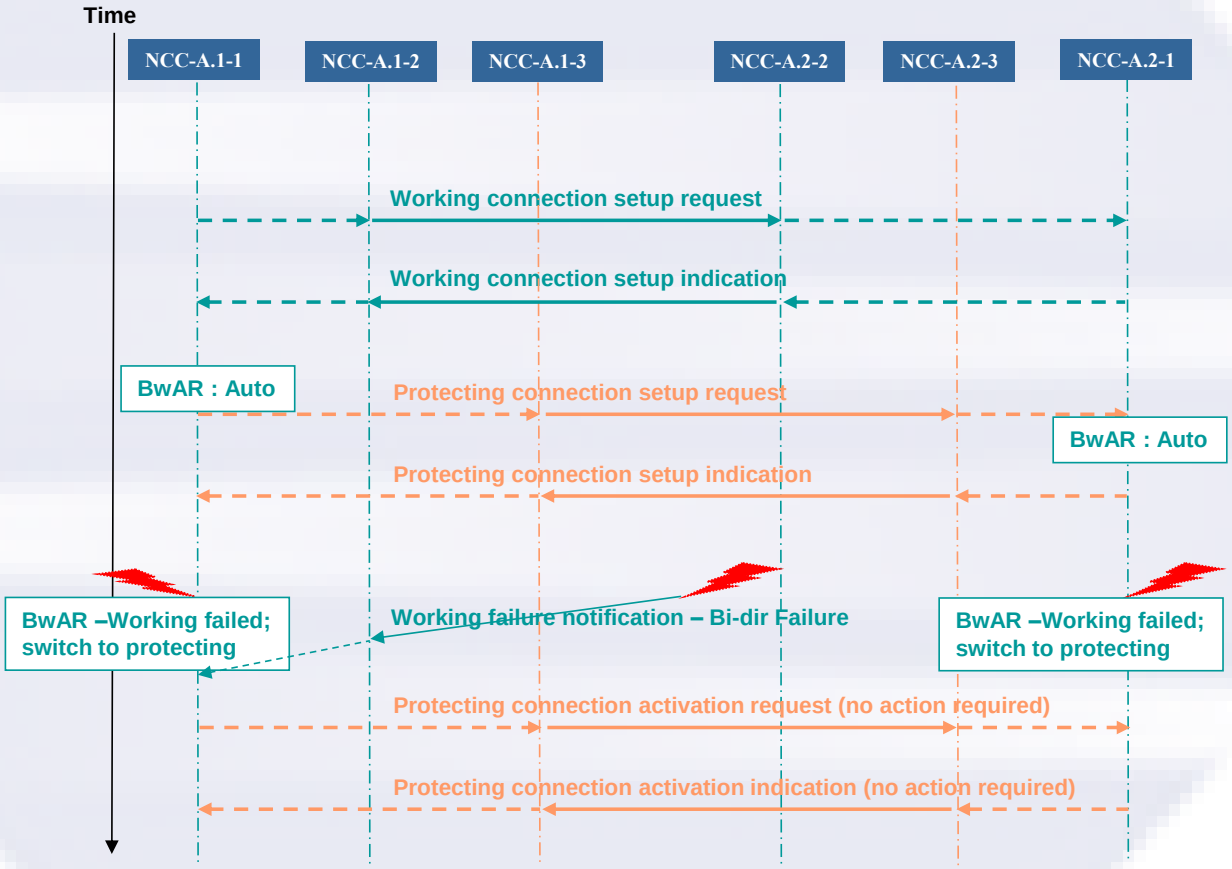


Figure 10: Abstract messages exchange for 1+1 protection recovery

9.3 Detailed signaling for shared-mesh restoration recovery

Figure 11 shows the abstract messages exchange during connection setup, then connection recovery, in a recovery domain when the selected recovery mechanism is shared-mesh restoration. It is based on Figure 3 reference network.

Some resources may be shared between multiple recovery connections (based on policies and if their associated working connections are resource-disjoints). If one recovery connection is activated, then a notification is sent to the DINs of all other recovery connections (if any) that share some resources with the one being activated: this notification will inform the DINs that those recovery connection can no longer be activated (the associated working connections can no longer be recovered in that recovery domain).

Such notifications are originated by transit NCCs inside the recovery domain (if the shared resource is a link, only by the NCC upstream – considering control plane signaling - to that link).

Similarly, when the original working connection failure is cleared and its associated recovery connection is de-activated, a notification is sent to the same DINs to inform them that the same recovery connections can be activated again (the associated working connections can be recovered again in that recovery domain).

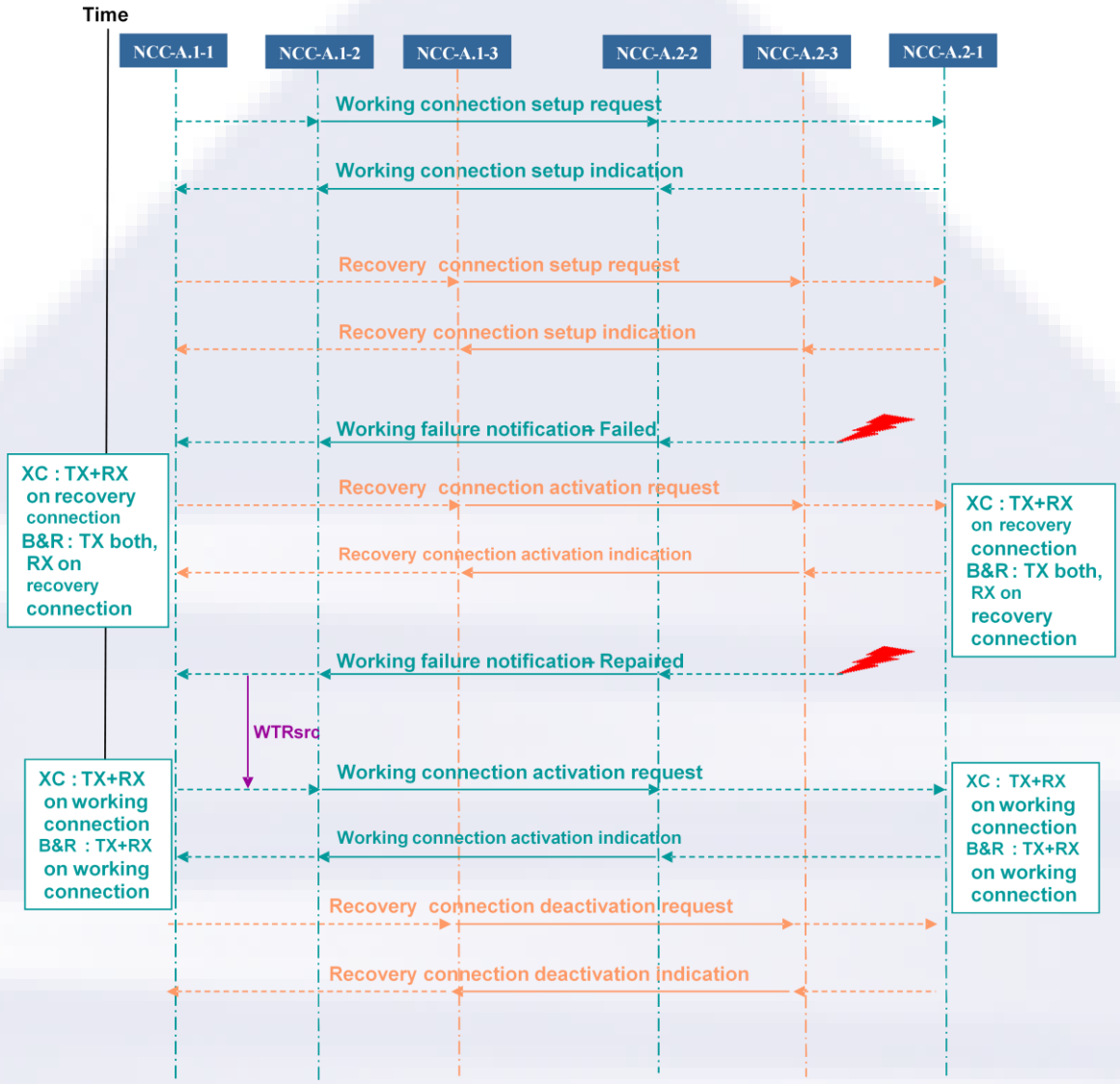


Figure 11: Abstract messages exchange for shared-mesh recovery

9.4 Detailed signaling for revertive hard-rerouting recovery

Figure 12 shows the abstract messages exchange during connection setup, then connection recovery, in a recovery domain when the selected recovery mechanism is revertive hard-rerouting. It is based on **Figure 3** reference network.

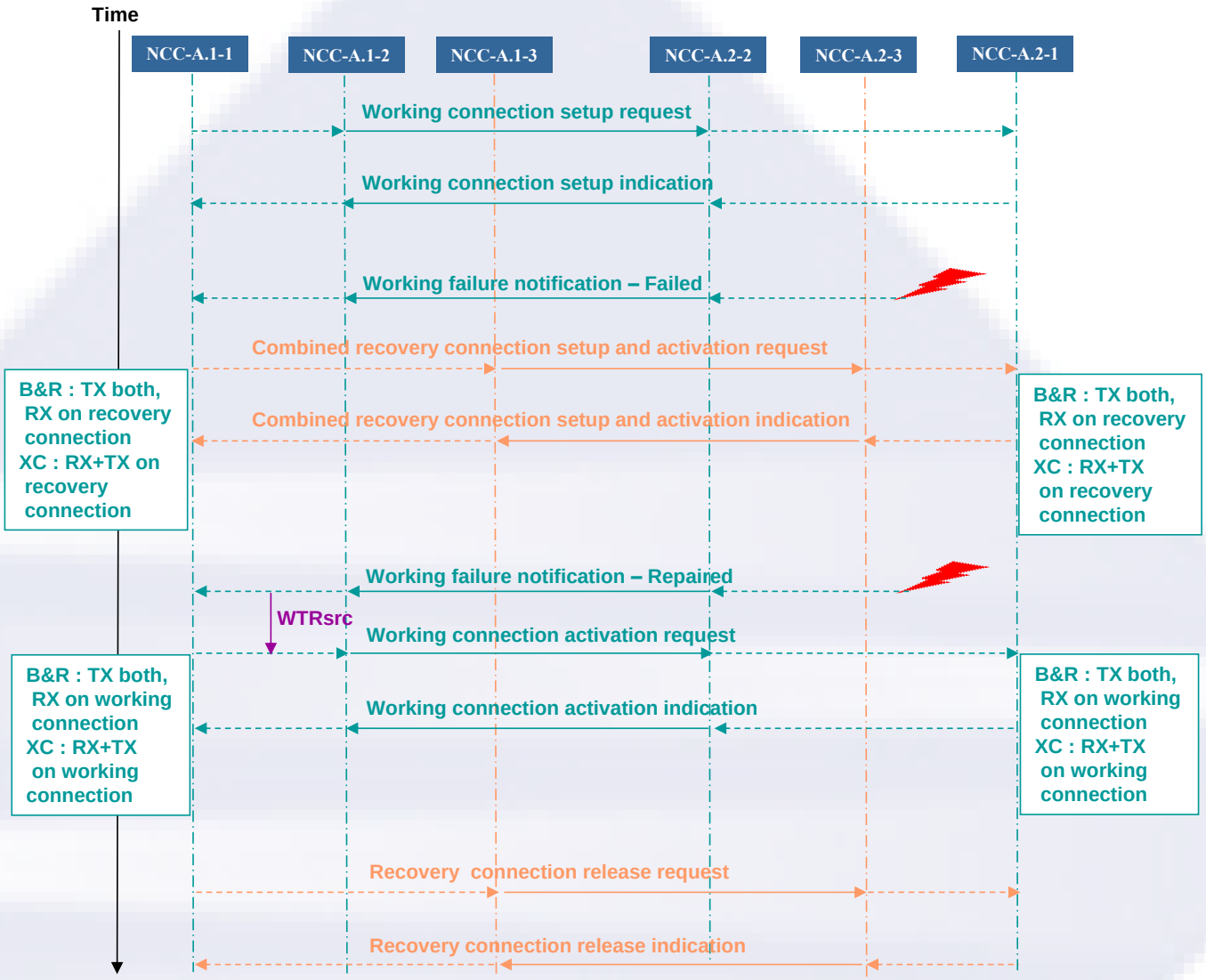


Figure 12: Abstract messages exchange for revertive hard-rerouting recovery

9.5 Detailed signaling for non-revertive hard-rerouting recovery

Figure 13 shows the abstract messages exchange during connection setup, then connection recovery, in a recovery domain when the selected recovery mechanism is non-revertive hard-rerouting. It is based on Figure 3 reference network.

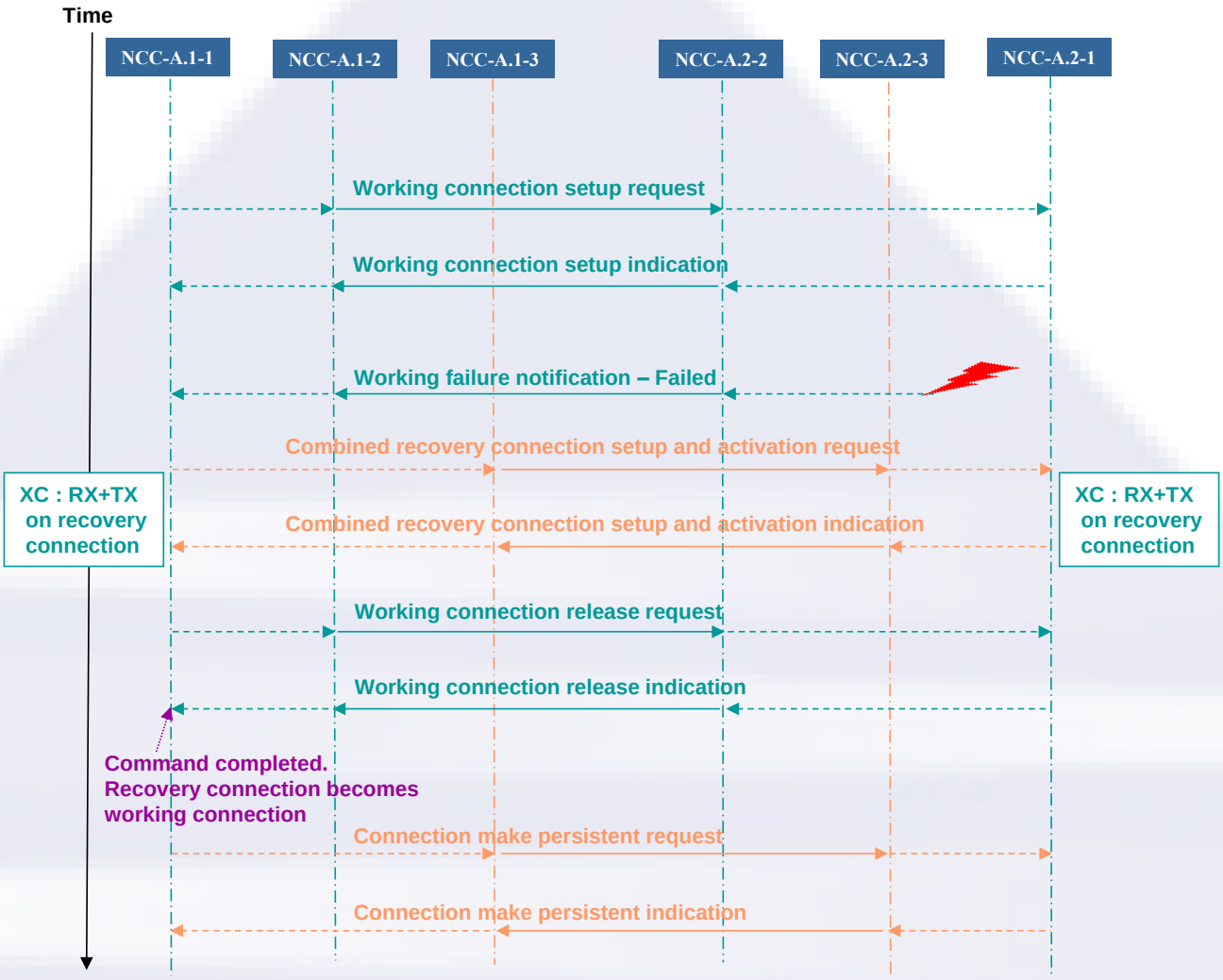


Figure 13: Abstract messages exchange for non-revertive hard-rerouting recovery

The Make Persistent abstract messages exchange allows the DEN to get the information that the working connection has been released (in case of SCN or control plane failure along the working connection path, the connection release request and/or indication may not go through). The DEN should not initiate a forced upstream deletion of the working connection if it receives a make persistent message for the recovery connection while no working connection release request has been received yet.

The last two steps (Working connection Release and Recovery Connection Make Persistent) may be performed in any order.

10 Operator commands

Operator commands are defined in linear protection standards [G.841, G873.1]. This chapter does not modify existing operator commands. There are two categories of operator commands: local and end-to-end.

- Local operator commands are not signaled end-to-end by any protocol (e.g. APS) between DIN and DEN. When symmetrical behavior is desired, they have to be applied at both DIN and DEN by the operator.
- End-to-end operator commands are signaled end-to-end by a protocol (e.g. APS) between DIN and DEN.

10.1 1+1 bi-directional protection

The following standards define operator commands priorities for SDH and OTN technologies:

- For SDH, refer to G.841 Table 8-1.

  G.841 considers that externally initiated commands (Clear, Lockout of Protection, Forced Switch, and Manual Switch) are local requests.

  G.841 does not define any APS protocol.

- For OTN, refer to G.873.1 Table 2 and Table 3.

  G.873.1 defines end-to-end commands and states. An APS protocol allows coordination between the two ends.

The control plane may be used to signal end-to-end operator commands and states to the far end, in case no APS protocol is used, or non-interoperable APS protocol implementations. Within an inter-domain context, E-NNI signaling may be extended to support this feature. This feature is outside the scope of this amendment.

10.2 Soft rerouting

Soft-rerouting can only be initiated from the DIN of a connection.

In soft rerouting, the use of Bridge & Roll-like mechanism for TDM data planes for moving traffic at both ends of the connection is highly recommended if traffic impact is to be kept to a minimum. The fact that the data plane approach may be different at the two ends of the connection does not preclude doing soft rerouting; however the traffic impact will be higher.

10.2.1 Detailed signaling for permanent soft rerouting

Figure 14 shows the abstract messages exchange during the soft rerouting of a working or recovery connection. It is based on Figure 3 reference network.

The new connection has the same role as the original one: it is a working connection if the connection being soft rerouted is a working one; it is a recovery connection if the connection being soft rerouted is a recovery one.

Figure 14: Abstract messages exchange during a permanent soft rerouting

Notes:

- The setup and activation steps can optionally be combined.
- An implementation may require an explicit operator trigger for any of the last three steps.
- The new connection may also be made persistent before the original one is released.

10.2.2 Detailed signaling for temporary soft rerouting

Figure 15 shows the abstract messages exchange during the temporary soft rerouting of a working or recovery connection. It is based on Figure 3 reference network.

The new connection has the same role as the original one: it is a working connection if the connection being soft rerouted is a working one; it is a recovery connection if the connection being soft rerouted is a recovery one.

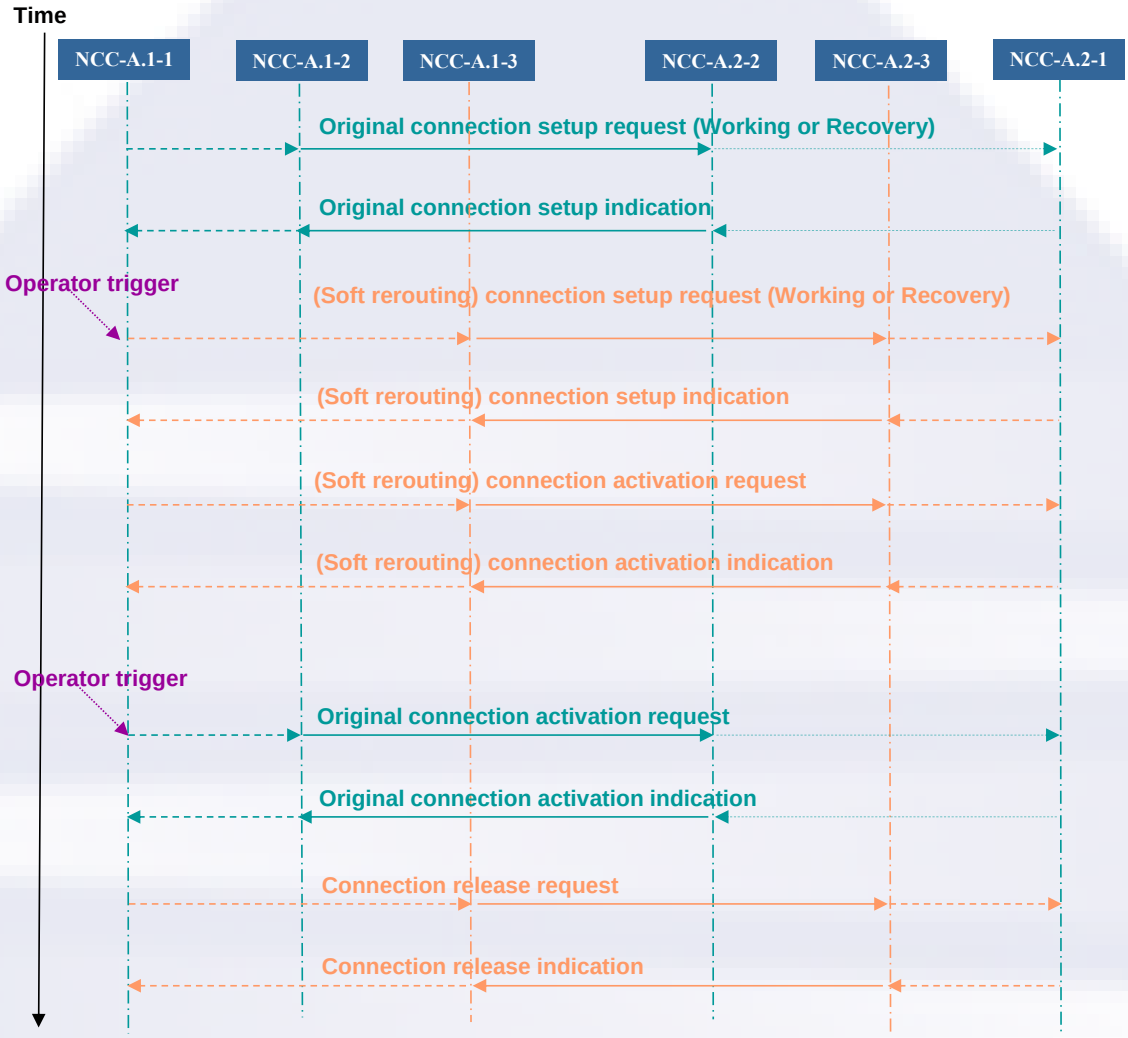


Figure 15: Abstract messages exchange during a temporary soft rerouting

Notes:

- The setup and activation steps can optionally be combined.
- An implementation may require an explicit operator trigger:
  - To activate the soft-rerouted connection;
  - To release the soft-rerouted connection.

11 Security and Logging

Forged or altered failure, recovery, escalated recovery, cleared failure, or reversion messages can cause inefficient use of data plane resources or even a denial of service condition. Exposure of such messages may also be undesirable. Methods for securing the OIF's control plane protocols are specified for the OIF's existing signaling and routing implementation agreements (see [SecExt]). This amendment introduces no new control plane protocols, so these methods apply equally to the messages defined in this IA.

If signaling used during recovery requires security and the required security associations do not exist, setting up these security associations may slow down the recovery process. Therefore, part of preparing for recovery may include making sure that all of the required security associations exist.

To verify the correct operation of the methods in this IA, the PROT@26041 logging capability [SysLog] can be turned on at one or more network elements implementing these recovery capabilities to generate a secure, time-stamped trace of control plane traffic. Vendors providing recovery and logging SHOULD provide configuration options to turn logging on or off automatically during failure and recovery conditions.

Vendors may use the methods in [SysLog] to define additional log messages at the appropriate Severity for failures, cleared failures, recovery actions, and reversion actions. These messages may contain additional information about the failed resources. Standardization of such messages is for future study.

12 References

12.1 Normative references

12.1.1 ITU-T

[G.841] ITU-T Recommendation G.841 (1998), *Types and characteristics of SDH network protection architectures*

[G.873.1] ITU-T Recommendation G.873.1 (2006), *Optical Transport Network (OTN): Linear protection*

[G.8080] ITU-T Recommendation G.8080/Y.1304 (2006), *Architecture of the Automatic Switched Optical Network (ASON)*

[G.8081] ITU-T Recommendation G.8081/Y.1353 (2009), *Terms and definitions for Automatically Switched Optical Networks (ASON)*

[G.7713] ITU-T Recommendation G.7713/Y.1704 (2006), *Distributed Call and Connection Management (DCM)*

12.1.2 OIF

[OIF-UNI-02.0] OIF Implementation Agreement, "User Network Interface (UNI) 2.0 Signaling Specification Common Part," OIF-UNI-02.0-Common, February 2008.

[OIF-UNI-02.0-R2-RSVP] OIF Implementation Agreement OIF-UNI-02.0-R2-RSVP, "RSVP Extensions for User Network Interface (UNI) 2.0 Signaling", January 2013, http://www.oiforum.com/public/documents/OIF-UNI-02.0-R2-RSVP.pdf

[OIF-E-NNI-sig-02.0] OIF Implementation Agreement, "OIF E-NNI Signaling Specification", OIF-E-NNI-Sig-02.0, April 2009

[OIF-ENNI-OSPF-02.0] OIF Implementation Agreement, "E-NNI OSPFv2-based Routing - 2.0 (Intra-Carrier)", OIF-ENNI-OSPF-02.0, July 2011

[SecExt] Optical Internetworking Forum Implementation Agreement, "Security Extension for UNI and E-NNI 2.0," OIF-SEP-03.0, November 2010.

[SysLog] Optical Internetworking Forum Implementation Agreement, "OIF Control Plane Logging and Auditing with Syslog version 1.1," OIF-SLG-01.1, July 2010.

12.1.3 IETF

[RFC4427] Mannie, E., Ed., and D. Papadimitriou, Ed., "Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4427, March 2006.

[RFC4872] Lang, J.P., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, May 2007.

[RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., Farrel, A., "GMPLS Segment Recovery", RFC 4873, May 2007.

[RFC4920] Farrel, A., Ed., Satyanarayana, A., Iwata, A., Fujita, N., and G. Ash, "Crankback Signaling Extensions for MPLS and GMPLS RSVP-TE", RFC 4920, July 2007.

12.2 Informative references

[OIF-CWG-CPR-01.0], OIF Carrier WG Guideline Document: Control Plane Requirements for Multi-Domain Optical Transport Networks.

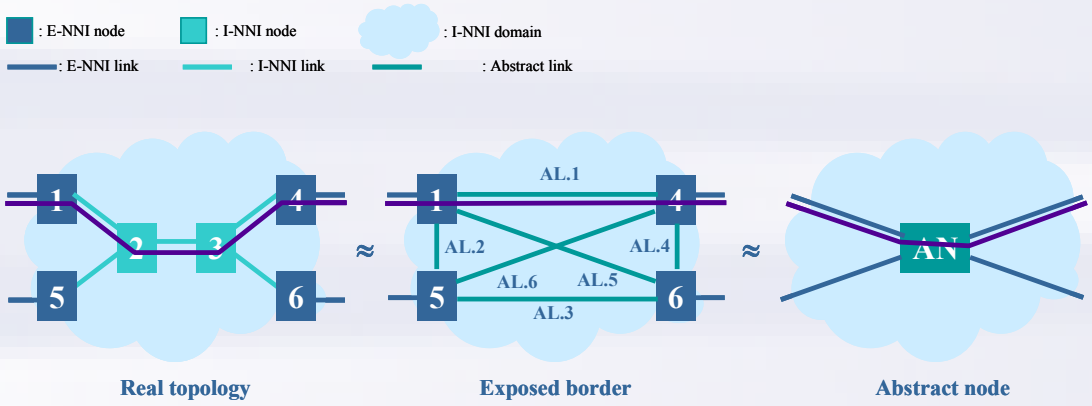
13 Appendix A: Failure notification and resources exhaustion inside a domain.

This appendix illustrates the cases where the lack of sufficient information when reporting a failure may be detrimental for the speed of hard rerouting process.

The following figures shows on the left the real topology that is represented in the upper layer either by exposed border model in the middle or by abstract node model on the right.

The exposed border model uses a full mesh of abstract links between border nodes (AL.1 to AL.6).

The connection in purple crosses the domain and is originated from a source node outside of the domain.



Case 1: Failure on link [3-4] (resource exhaustion towards node [4])
In the exposed border scenario, as per section 6.2.2.1, if node [1] report AL.1 as failed, the source node may still compute a path using {node[1]-AL.5-node[6]-AL.4-node[4]} which would fail during the signaling phase. Another re-attempt could be {node[1]-AL.2-node[5]-AL.6-node[4]} which would also fail. Depending on the lack of crankback information received earlier, it may also re-attemp with no more success {node[1]-AL.2-node[5]-AL.3-node[6]-AL.4-node[4]}.

At last, the optimal recovery path {node[1]-AL.5-node[6]} (in reality {node[1]-node[2]-node[3]-node[6]}) and other succession of domains could be calculated in order to satisfy the recovery.

In the abstract node model, as per section 6.2.2.1, if AN report the ingress E-NNI as failed, the source node is not able to compute what would be the optimal path (in reality {node[1]-node[2]-node[3]-node[6]}) and would therefore select a non-optimized path.

14 Appendix B: ASON architecture and recovery domains

The ASON architecture specifies that a recovery domain is fully contained in a subnetwork. The recovery domains, comply with the ASON architecture. Every node participating in one or multiple recovery domains may be decomposed into multiple logical entities, each entity participating in one recovery domain.

For instance, **Figure 16** shows such a possible decomposition for node 2 and node 3 as shown in Figure 1.

Figure 16 does not show how such entities are interconnected within a node. Note that such internal topology information is not shared outside that node.

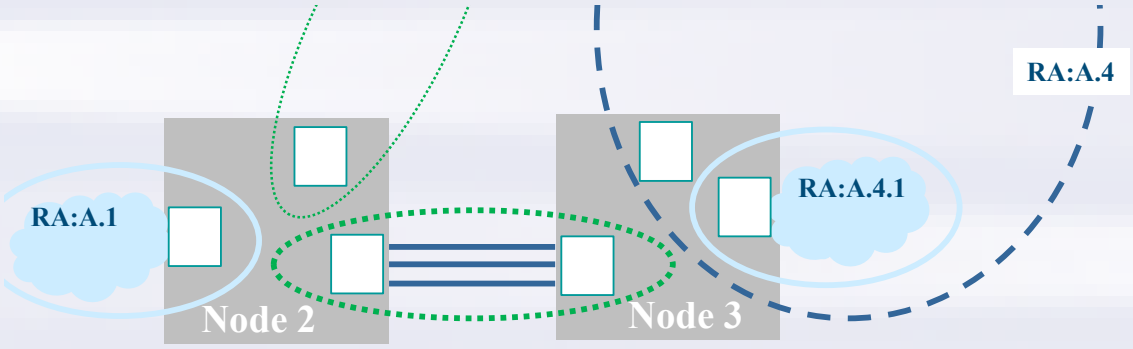


Figure 16: Recovery domain boundaries within a node

15 Appendix C: List of companies belonging to OIF when document is approved

Acacia Communications
ADVA Optical Networking
Agilent Technologies R & D
Alcatel-Lucent
Altera
AMSS
Amphenol
Anritsu
Applied Communication Sciences
AT&T
Avago
Broadcom
Brocade
Centellax
China Telecom
Ciena
Cisco
ClariPhy
Coriant
Cortina Systems
CPqD
Department of Defense
Deutsche Telekom
Emcore
Ericsson
FCI USA LLC
Fiberhome Technologies Group
Finisar
Fujikura
Fujitsu
Furukawa Electric Japan
Google
Hewlett Packard
Hitachi
Hittite Microwave
Huawei Technologies
IBM
Infinera
Inphi
Intel
JDSU

Juniper Networks
Kaiam
Kandou
KDDI R & R Laboratories
LeCroy
LSI
Luxtera
M/A-COM Technology Solutions
Marben Products
Mellanox
Metaswitch
Mindspeed
Mitsubishi Electric
Molex
MoSys
MultiPhy
NEC
NeoPhotonics
NTT
Oclaro
Optelian
Orange
PETRA
PMC Sierra
QLogic
Ranovus
Semtech
Skorpios
Sumitomo Electric
Sumitomo Osaka Cement
TE Connectivity
Tektronix
Tellabs
TELUS Communications
TeraXion
Texas Instruments
Time Warner Cable
TriQuint Semiconductor
u2t Photonics AG
US Conec
Verizon
Xilinx
Xtera Communications
Yamaichi Electronics