ITUPublications
Recommendations

International Telecommunication Union
Standardization Sector

Recommendation

ITU-T G.7703 (2021) Amd. 1 (11/2022)

SERIES G: Transmission systems and media, digital systems and networks

Data over Transport – Generic aspects – Transport network control aspects

---

Architecture for the automatically switched optical network

Amendment 1

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
General	G.7000–G.7099
Transport network control aspects	**G.7700–G.7799**
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T G.7703

Architecture for the automatically switched optical network

Amendment 1

Summary

Recommendation ITU-T G.7703 describes the reference architecture and requirements for the automatically switched optical network (ASON) as applicable to connection-oriented circuit or packet transport networks. This reference architecture is described in terms of the key functional components and the interactions between them.

Amendment 1 aligns with Recommendation ITU-T G.7701 (2022), which specifies common control aspects for both ASON and software defined networking (SDN) architecture. This amendment refers to Recommendation ITU-T G.7701 common clauses.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T G.8080/Y.1304	2001-11-29	15	11.1002/1000/5639
1.1	ITU-T G.8080/Y.1304 (2001) Amd. 1	2003-03-16	15	11.1002/1000/6292
1.2	ITU-T G.8080/Y.1304 (2001) Amd. 2	2005-02-22	15	11.1002/1000/7482
2.0	ITU-T G.8080/Y.1304	2006-06-06	15	11.1002/1000/8777
2.1	ITU-T G.8080/Y.1304 (2006) Cor. 1	2007-09-06	15	11.1002/1000/9175
2.2	ITU-T G.8080/Y.1304 (2006) Amd. 1	2008-03-29	15	11.1002/1000/9393
2.3	ITU-T G.8080/Y.1304 (2006) Amd. 2	2010-09-06	15	11.1002/1000/10903
3.0	ITU-T G.8080/Y.1304	2012-02-13	15	11.1002/1000/11515
4.0	ITU-T G.7703	2021-05-29	15	11.1002/1000/14637
4.1	ITU-T G.7703 (2021) Amd. 1	2022-11-13	15	11.1002/1000/15143

Keywords

Automatic switched optical network, ASON, architecture, control and management, control components.

---

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T G.7703

Architecture for the automatically switched optical network

Amendment 1

Editorial note: This is a complete-text publication. Modifications introduced by this amendment are shown in revision marks relative to Recommendation ITU-T G.7703 (2021).

1 Scope

This Recommendation specifies the architecture and requirements for the automatic switched transport network (ASON). ASON control components may be used to control any layer network technology that provides connection oriented circuit switching (CO–CS) or connection oriented packet switching (CO–PS) as defined in [ITU-T G.800].

This Recommendation describes the set of control components that are used to manipulate transport network resources in order to provide the functionality of setting up, maintaining and releasing connections. The use of components allows for the separation of call control from connection control and the separation of routing and signalling.

These control components are based on the set of common control components described in [ITU-T G.7701] and represent abstract entities rather than instances of implementable software. UML-like notation is used to describe components of the ASON architecture.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.800] Recommendation ITU-T G.800 (2016), *Unified functional architecture of transport networks*.

[ITU-T G.805] Recommendation ITU-T G.805 (2000), *Generic functional architecture of transport networks*.

[ITU-T G.7701] Recommendation ITU-T G.7701 (~~2016~~2022), *Common control aspects*.

[ITU-T G.7715] Recommendation ITU T G.7715/Y.1706 (2007), *Architecture and requirements for routing in the automatically switched optical networks.*

[ITU-T G.7718] Recommendation ITU-T G.7718/Y.1709 (2020), *Framework for the management of management-control components and functions*.

[ITU-T M.3100] Recommendation ITU-T M.3100 (2005), *Generic network information model*.

[ITU-T Y.1312] Recommendation ITU-T Y.1312 (2003), *Layer 1 Virtual Private Network generic requirements and architecture elements*.

3 Definitions

This Recommendation uses the following terms defined elsewhere:

3.1 Terms defined elsewhere

3.1.1 access group: [ITU-T G.805].

3.1.2 adaptation: [ITU-T G.805].

3.1.3 address: [ITU-T G.7701].

3.1.4 administrative domain: [ITU-T G.7701].

3.1.5 allocated (resource) label range: [ITU-T G.7701].

3.1.6 automatically switched optical network (ASON): [ITU-T G.7701].

3.1.~~6~~7 boundary resource identifier (BRI): [ITU-T G.7701].

3.1.~~7~~8 call: [ITU-T G.7701].

3.1.~~8~~9 call admission control: [ITU-T G.7701].

3.1.~~9~~10 call control: [ITU-T G.7701].

3.1.1~~0~~0 call segment: [ITU-T G.7701].

3.1.1~~1~~1 characteristic information: [ITU-T G.80~~5~~50].

3.1.1~~2~~2 component: [ITU-T G.7701].

3.1.1~~3~~3 configured (resource) label: [ITU-T G.7701].

3.1.1~~4~~4 connection: [ITU-T G.805].

3.1.1~~5~~5 connection point (CP): [ITU-T G.805].

3.1.1~~6~~6 connection termination point (CTP): [ITU-T -M.3100].

3.1.1~~7~~7 control domain: [ITU-T G.7701].

3.1.1~~8~~8 discovery agent (DA): [ITU-T G.7701].

3.1.20~~19~~ layer network: [ITU-T G.805].

3.1.2~~0~~0 link: [ITU-T G.805].

3.1.2~~1~~1 link connection: [ITU-T G.805].

3.1.2~~2~~2 link resource manager (LRM): [ITU-T G.7701].

3.1.2~~3~~3 policy: [ITU-T G.7701].

3.1.2~~4~~4 potential (resource) label range: [ITU-T G.7701].

3.1.2~~5~~5 potential SNPs: [ITU-T G.7701].

3.1.2~~6~~6 protocol controller (PC): [ITU-T G.7701].

3.1.2~~7~~7 recovery domain: [ITU-T G.7701].

3.1.2~~8~~8 route: [ITU-T G.7701].

3.1.30~~29~~ routing area (RA): [ITU-T G.7701].

3.1.3~~0~~0 routing domain: [ITU-T G.7701].

3.1.3~~1~~1 routing level: [ITU-T G.7701].

3.1.3~~2~~2 service level agreement: [ITU-T G.7701].

3.1.343 subnetwork: [ITU-T G.805].

3.1.354 subnetwork connection: [ITU-T G.805].

3.1.365 subnetwork point (SNP): [ITU-T G.7701].

3.1.376 subnetwork point pool (SNPP): [ITU-T G.7701].

3.1.387 subnetwork point pool link (SNPP link): [ITU-T G.7701].

3.1.398 trail: [ITU-T G.805].

3.1.4039 transitional SNPP link: [ITU-T G.7701].

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 assigned SNPs: Those SNPs out of the set of potential SNPs that have been assigned to a particular connection. This means that the associated label corresponds to a configured label.

3.2.2 E-NNI: A bidirectional interface between control components belonging to different domains.

3.2.3 hard rerouting: A failure recovery function in a rerouting domain that attempts to create another connection to the destination at the edge of the rerouting domain. This is performed in response to the failure of an existing connection, and the rerouted connection replaces the connection that contained the failure.

3.2.4 I-NNI: A bidirectional interface between control components belonging to one or more domains having a trusted relationship.

3.2.5 interface: In the context of this Recommendation, interfaces represent logical relationships between automatic switched transport network (ASON) control components, and are defined by the information flow between these components. Such a relationship allows distribution of these control components to support different equipment implementations and network architectures.

3.2.6 multi-homed: A user is considered to be multi-homed when there are two or more subnetwork point pool (SNPP) links connecting the access group container to the network.

3.2.7 name: A name, or identifier, is a location-independent string with respect to both a source and a destination. If a string is the name of a destination, it remains unchanged if the destination moves. It is valid regardless of the source attempting communication with the destination.

3.2.8 permanent connection: A type of connection that is established within an automatic switched transport network (ASON) control domain by an external management and control (MC) system. The MC components within the ASON control domain cannot create, delete or modify a permanent connection.

3.2.9 restoration: Restoration is the action of replacing a connection in a call by rerouting the connection.

3.2.10 soft permanent connection (SPC): The concatenation of a permanent connection at the edge of the network with a switched connection within the automatic switched transport network (ASON) control domain to provide an end-to-end connection.

3.2.11 soft rerouting: A function that reroutes a connection for administrative purposes. The original connection is not taken out of service until the rerouted connection is established.

3.2.12 supplementary services: Within a transport network, supplementary services are considered to be the set of services that are provided to end users over and above connection management.

3.2.13 switched connection (SC): This type of connection is established on demand by the communicating end points within an automatic switched transport network (ASON) control domain

using a dynamic protocol message exchange in the form of signalling messages. These messages flow across the UNI, I-NNI or E-NNI within the control domain.

3.2.14 termination connection point (TCP): For the purposes of this Recommendation, a termination connection point represents the output of a trail termination function or the input to a trail termination sink function. (Note that in [ITU-T G.805] the TCP refers to the binding between two points.)

3.2.15 third party signalling: A third party signalling entity is a party that acts on behalf of a user and exchanges information between the user and the management and control (MC) components for the purpose of connection supervision.

3.2.16 E-NNI boundary resource identifier (BRI): The E-NNI SNPP link may be assigned an identifier for the network call controllers to specify E-NNIs. These identifiers must be globally unique and are assigned for the automatically switched optical network (ASON). Multiple identifiers may be assigned to the SNPP link.

3.2.17 UNI BRI: The UNI SNPP link may be assigned an identifier for the calling party call controller and network call controller to specify destinations. These identifiers must be globally unique, are assigned for the automatically switched optical network (ASON), and may be in a 1:n or n:1 relationship with SNPP links.

3.2.18 user-network interface (UNI): A bidirectional signalling interface between the control components of a service requester and a service provider.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AGC	Access Group Container
ASON	Automatic Switched Transport Network
BRI	Boundary ~~r~~Resource ~~i~~Identifier
CC	Connection Controller
CCC	Calling party Call Controller
CI	Characteristic Information
CP	Connection Point
CTP	Connection Termination Point
DA	Discovery Agent
DCN	Data Communication Network
E-NNI	External Network-Network Interface (reference point)
FCAPS	Fault, Configuration, Accounting, Performance, Security
FP	Framing Procedure
GFP	Generic Framing Procedure
~~id~~ID	I~~i~~dentifier
I-NNI	Internal Network-Network Interface (reference point)
LRM	Link Resource Manager
MC	Management and Control
MI	Management Information

MO Managed Object

NCC Network Call Controller

NMI Network Management Interface

PC Protocol Controller

RC Routing Controller

SCN Signalling Control Network

SNC Subnetwork Connection

SNP Subnetwork Point

SNPP Subnetwork Point Pool

SPC Soft Permanent Connection

SC Switched Connection

SDN Software-Defined Networking

TAP Termination and Adaptation Performer

TCP Termination Connection Point

UML Unified Modelling Language

UNI User-Network Interface (reference point)

VPN Virtual Private Network

5 Conventions

This Recommendation uses the diagrammatic conventions defined in [ITU-T G.800] to describe the transport resources.

This Recommendation uses the diagrammatic conventions defined in [ITU-T G.7701] to describe controller components.

6 Overview

The purpose of the automatic switched optical network (ASON) management and control (MC) components is to:

– facilitate fast and efficient configuration of connections within a transport layer network to support both switched and soft permanent connections;

– reconfigure or modify connections that support calls that have previously been set up;

– perform a restoration function.

A well-designed control architecture should give service providers control of their network, while providing fast and reliable call set-up. The control architecture should be reliable, scalable, and efficient. It should be sufficiently generic to support different technologies, differing business needs and different distribution of functions by vendors (i.e., different packaging of the control components).

An ASON control domain is composed of different components that provide specific functions including that of route determination and signalling. The control components are described in terms that place no restrictions regarding how these functions are combined and packaged. Interactions among these components, and the information flow required for communication between components are achieved via interfaces.

This Recommendation deals with the control components and the interaction between the control components, the transport resources and other fault, configuration, accounting, performance, security (FCAPS) functions, as shown in the high level view in Figure 6-1. The other FCAPS functions and transport layer networks are specified in other ITU-T Recommendations and are outside the scope of this Recommendation. However, the other FCAPS functions are shown alongside SDN and ASON MC systems which are part of the management-control continuum.

Within each instance of an ASON MC system in Figure 6-1, are MC components, of which some are components detailed in [ITU-T G.7701]. These are represented by the yellow triangle symbol and it those components may be used in other MC systems. Each instance of an ASON MC system has in its scope a set of transport resources over which it supports connection management in a layer network. Management functions recur and a management function instance can manage a set of other management function instances. This is depicted in the network administration role that shows the functions "Configuration of MC components and system" and "Assignment of transport resources to a MC system" The first function would be used to instantiate an ASON MC system and its components, and the second function assigns transport resources to the scope of that ASON MC system.

Not shown in this figure is the data communication network (DCN), which provides the communication paths to carry signalling and management information (MI). The DCN is a separate set of transport resources also controlled by an MC system. The details of the DCN, other MC systems and the transport resources are outside the scope of this Recommendation. Functions pertaining to the control components are described in this Recommendation. Management of the MC components is specified in [ITU-T G.7718].

ASON deployment will occur within the context of commercial operator business practices and the multi-dimensional heterogeneity of transport networks. These business and operational considerations lead to the need for architectural support of, for example, strong abstraction barriers to protect commercial business operating practices, segmenting transport networks into domains according to managerial and/or policy considerations, and inherent transport network heterogeneity (including control and management). The domain notion embodied in the [ITU-T G.805] definition of administrative domain and the Internet administrative regions (e.g., autonomous systems) has been generalized in the architecture to express differing administrative and/or managerial responsibilities, trust relationships, addressing schemes, infrastructure capabilities, survivability techniques, distributions of control functionality, etc. Domains are established by operator policies and have a range of membership criteria, as exemplified above.

ASON supports connection services (see Annex A) through the automatic provisioning of end-to-end transport connections across one or more domains. This involves both a service and connection perspective:

– The service (call) perspective is to support the provisioning of end-to-end services while preserving the independent nature of the various businesses involved.

– The connection perspective is to automatically provision "path layer" connections (in support of a service) that span one or more domains.

Connection state information (e.g., fault and signal quality) is detected by the transport network resources and provided to the control components.

ASON carries (distributes) link status (e.g., adjacency, available capacity and failure) information to support connection set-up/release and restoration.

Detailed fault management information or performance monitoring information is transported within the transport network (via the overhead/OAM) and via the DCN.

The interconnection between and within domains is described in terms of reference points. As domains are established via operator policies, inter-domain reference points are service demarcation

points for a single service layer (i.e., points where call control is provided). The exchange of information across these reference points is described by the multiple abstract interfaces between control components. A physical interface is provided by mapping one or more abstract component interfaces to a protocol. Multiple abstract interfaces may be multiplexed over a single physical interface. The reference point between a user and a provider domain is the user-network interface (UNI), which represents a user-provider service demarcation point. The reference point between domains is the external network-network interface (E-NNI), which represents a service demarcation point supporting multi-domain connection establishment. The reference point within a domain is an internal network-network interface (I-NNI), which represents a connection point supporting intra-domain connection establishment. The information flows across these reference points are further described in clause 11.

The MC components may also be subdivided to allow the segregation of resources for example between virtual private networks (VPNs). If the resources are dedicated to independent domains, then no reference points are provided between these domains.

Separate descriptions are provided for the interactions between the:

– MC components and transport networks; and the

– MC systems and transport network resources resulting from the addition of the MC components for connection management and connection monitor configuration.

This Recommendation encompasses the control of transport layer network connections, including inter-layer interactions arising from requests for capacity in server layers.

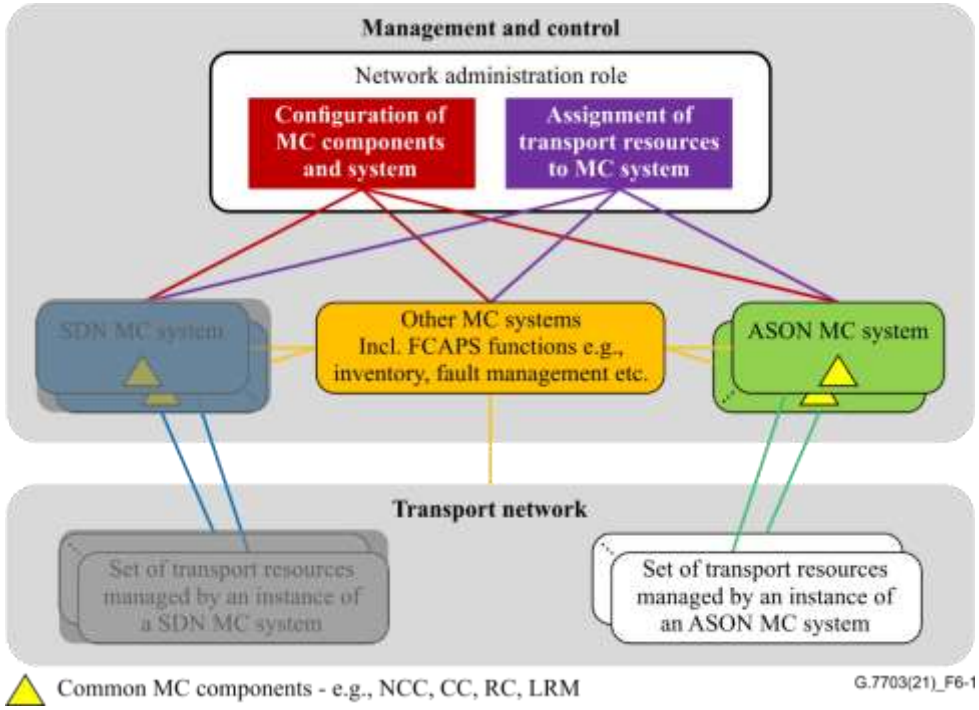


Figure 6-1 – Relationship between architectural components

6.1 Control component overview

This clause describes a reference architecture for ASON that supports the requirements in this Recommendation, identifying its key functional components and how they interact. This flexible reference architecture is intended to enable operators to support their internal business and managerial practices, as well as to bill for service usage. The ASON architecture should have the following characteristics:

– Support various transport infrastructures, such as those covered by [ITU-T G.800].

– Be applicable regardless of the particular choice of control protocol (i.e., employ a protocol neutral approach that is independent of the particular connection control protocols used).
– Be applicable regardless of how the ASON control domains are subdivided into domains and routing areas, and how the transport resources have been partitioned into subnetworks.
– Be applicable regardless of the implementation of connection control that may range from a fully distributed to a centralized control architecture.

This reference architecture describes the:

– functional components of an ASON control domain, including abstract interfaces and primitives
– interactions between call controller components
– interactions among components during connection set-up
– functional component that transforms the abstract component interfaces into protocols on external interfaces.

The detail of the interfaces of these and other components are provided in other technology specific Recommendations.

Protocol controllers are provided to take the primitive interface supplied by one or more architectural components, and multiplex those interfaces into a single instance of a protocol. In this way, a protocol controller absorbs variations among various protocol choices, and the architecture remains invariant. One or more protocol controllers are responsible for managing the information flows across a reference point.

Protocol controllers also apply rules to system interfaces to provide a secure environment for the architectural components to execute in, thereby isolating the architectural components from security considerations. In particular, they isolate the architecture from distribution decisions made involving security issues.

6.2 Call and connection control

Call and connection control are introduced in clause 6.1 of [ITU-T G.7701].

This Recommendation, ~~separates the treatment of call control and connection control, call control is only needed at domain boundaries that represent a UNI or E-NNI. Thus, within a domain or between domains that use an I-NNI it is only necessary to support procedures for connection control. Additionally,~~ call control is also provided at interlayer network call controller (NCC) boundaries. ~~The functions performed by the call controllers at domain boundaries are defined by the policies associated by the interactions allowed between the domains. Policies are established by the operator. As such, an end-to-end call is considered to consist of multiple call segments, depending on whether the call traverses multiple domains. This allows for flexibility in the choice of signalling, routing and recovery paradigms in different domains.~~

~~It should be noted that the call is the representation of the service offered to the user of a network layer, while the connections are one of the means by which networks deliver said services. There may be other entities used in supporting calls, such as service specific processes.~~

6.2.1 Call control

~~Call control is a signalling association between one or more user applications and the network to control the set-up, release, modification and maintenance of sets of connections. Call control is used to maintain the association between parties and a call may embody any number of underlying connections, including zero, at any instant of time.~~

~~Call control may be realized by one of the following methods:~~

– separation of the call information into parameters carried by a single call/connection protocol;

– separation of the state machines for call control and connection control, whilst signalling information in a single call/connection protocol;

– separation of information and state machines by providing separate signalling protocols for call control and connection control.

Call control must provide coordination of connections (in a multi-connection call) and the coordination of parties (multi-party calls). To coordinate multiple connections, the following actions need to take place in the network:

– All connections must be routed so that they can be monitored by at least one coordinating (call control) entity.

– Call control associations must be completed before connections are set up. A call may exist without any connections (facilitating complex connection rearrangements).

A call can be considered to have three phases:

Establishment

During this phase, signalling messages are exchanged between users and the network to negotiate the call characteristics. The exchange of signalling messages between the calling party and the network is known as an outgoing call. The exchange of signalling messages between the network and the called party is referred to as an incoming call.

Active

During this phase, data can be exchanged on the associated connections and call parameters may also be modified (e.g., the addition of new parties in a point-to-multi-point call, where this type of call is supported).

Release

During this phase, signalling messages are exchanged between calling and called parties and the network to terminate the call. A call may be released by either the calling or called terminals or by proxy or network management. Call control is described in clause 6.1.1 of [ITU-T G.7701].

6.2.2 Call admission control

Call admission control is described in clause 6.1.2 of [ITU-T G.7701].

Call admission control is a policy invoked by an originating role in a network and may involve cooperation with the terminating role in the network. Note that a call being allowed to proceed only indicates that the call may proceed to request one or more connections. It does not imply that any of those connection requests will succeed. Call admission control may also be invoked at other network boundaries.

The originating call admission function is responsible for checking that a valid called username and parameters have been provided. The service parameters are checked against a service level specification (a set of parameters and values agreed between a network operator and customer for a particular service indicating the '*scope*' of the service). If necessary, these parameters may need to be renegotiated with the originating user. The scope of this negotiation is determined by policies derived from the original service level specification, which itself is derived from the service level agreement (the service contract between a network operator and a customer that defines global responsibilities between them).

The terminating call admission function is responsible for checking that the called party is entitled to accept the call, based on the calling party and called party service contracts. For example, a caller address may be screened.

6.2.3 Connection control

Connection control is described in clause 6.1.3 of [ITU-T G.7701].

~~Connection control is responsible for the overall control of individual connections. The overall control of a connection is performed by the protocol undertaking the set-up and release procedures associated with a connection and the maintenance of the state of the connection.~~

6.2.4 Connection admission control

Connection admission control is described in clause 6.1.4 of [ITU-T G.7701].

~~Connection admission control is essentially a process that determines if there are sufficient resources to admit a connection (or renegotiates resources during a call). This is usually performed on a link-by-link basis, based on local conditions and policy. For a circuit switched network, this may simply devolve to whether there are free resources available. In contrast, for packet switched networks such as ATM, where there are multiple quality of service parameters, connection admission control needs to ensure that admission of new connections is compatible with existing quality of service agreements for existing connections. Connection admission control may refuse the connection request.~~

6.2.5 Relationship between call state and connection state

See clause 6.1.5 of [ITU-T G.7701].

~~The call state has dependency upon the state of the associated connections. This dependency is related to call type and policy. For example, where there is a single connection and it fails, the call may be immediately released, or alternatively, may be released after a period of time if no alternative connection can be obtained using mechanisms such as protection or restoration. Note that call and connection coincide at domain boundaries.~~

6.3 Interaction between control domains, transport resources and other MC systems

Figure 6-2 illustrates the general relationships between the ASON control domain, other MC systems and transport resources. Each one is autonomous, but some interaction will occur. The following provides further details on the interactions between the various planes.

6.3.1 Other MC systems – Transport interaction

Other MC systems interact with transport resources by operating on a suitable information model, which presents a management view of the underlying resource. The objects of the information model are physically located with the transport resource, and interact with that resource via the management information (MI) interfaces of the layer-specific functional model. These interfaces should be collocated with the managed object (MO) and the control component.

6.3.2 ASON control domain – Transport interaction

Only two architectural components have a strong relationship to a physical transport resource.

At the lower limit of recursion, the connection controller (CC) provides a signalling interface to control a connection function. This component is physically located with the connection function and all further hardware details are hidden. However, given the limited information flow a new protocol may be useful to optimize this communication. The termination and adaptation performer (TAP) is physically located with the equipment that provides adaptation and termination functions, and provides a control view of link connections. The TAP hides the interaction with the hardware.

6.3.3 Other MC systems – ASON control domain interaction

Clause 8.1 of [ITU-T G.7701] states that each component has a set of special interfaces to allow for monitoring of the component operation, and dynamically setting policies and affecting internal behaviour. These interfaces are equivalent to the MI interface of the transport functional model, and

allow the component to present a view to a management system and to be configured by a management system.

The other MC systems interact with control components by operating on a suitable information model, which presents a management view of the underlying component. The objects of the information model are physically located with a control component, and interact with that component via the monitor and configuration interfaces of that component. These interfaces should be collocated with the managed object and the control component.

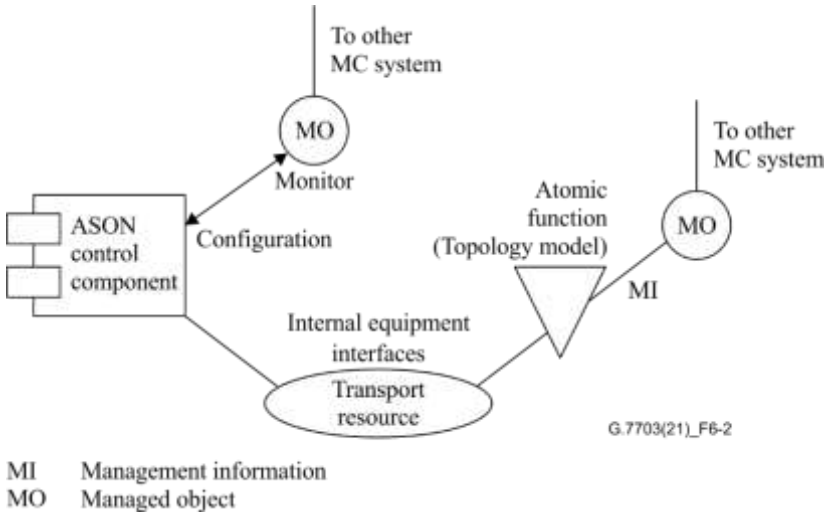


Figure 6-2 – MC system interactions with transport resources

The physical transport resources, which represent the physical reality of the equipment, are described in terms of ITU-T G.805 atomic functions. Managed objects (MO), which represent the external management view of the equipment, interact with the functional model specified in equipment recommendations via the MI reference points, which are also entirely within the equipment. Note that the managed object represents the management view regardless of the management protocol used. The information is independent of the protocol used.

From the MC components view, MC components operate directly on the transport resources, so MC components operation appears autonomous to the other MC components. Likewise, operations performed by other MC systems appear autonomous to an ASON control domain. This is exactly the same situation we have when multiple managers manage equipment. Each manager is unaware of each other's existence, and simply sees autonomous equipment behaviour. Although the information presented to the ASON control domain is similar to that presented to other FCAPS functions, it is not identical to the MI information. Control information overlaps the MI data because the control domain requires some but not all the MI information. For example, restoration is likely to be triggered by the same conditions that normally trigger protection actions.

Component-specific managed objects present a management view of control components via the monitor interfaces on the component. It is critical to realize that this is the view of the manageable aspects of the component, and not a view of the transport resource, which is obtained via the management view.

6.3.4 Resource management

Network resources may be partitioned between those under the authority of FCAPS functions and those under the authority of the ASON control domain. It shall not be possible for an ASON control domain to modify resources that are under the authority of other FCAPS functions. This includes network resources not currently in use, but reserved for future use (e.g., by network planners).

6.4 User architecture

The user side will be referred to as the UNI-C (for "client"), and the network side will be referred to as the UNI-N (for "network").

In this Recommendation the UNI BRI defines one or more globally unique names to each subnetwork point pool (SNPP) link that is part of a UNI. These names are used to identify call destinations. Given that a UNI may contain multiple SNPP links, as in the case of multi-homing, a UNI may therefore have multiple globally unique names for its bearer resources. Note that these names are not usernames.

When there are multiple SNPP links that are part of the same UNI, those addresses can be used to discriminate between which SNPP link to use. Factors such as diversity or cost could be used by callers to select the appropriate SNPP link. SNPP links between a common access group container (AGC) and a network may be in the same UNI if, on the network side, they are within the scope of a common network call controller component.

UNI BRIs can be used to differentiate between UNIs to a user. When there are multiple UNIs, each has distinct UNI BRIs and they do not share a common address.

The following describes the UNI-C architecture:

1) There exists a transport entity called an access group container (AGC) that can terminate multiple SNPP links. This entity can contain a set of [ITU-T G.805] access groups.

2) An AGC is a single layer entity that contains access groups, link resource managers (LRMs), and TAPs. It is similar to [ITU-T G.805] subnetworks except that it is not recursively defined, may or may not be a matrix (it does not have to be specified), and has no defined subnetwork connections. Multiple AGCs from different layers may be coincident in the same equipment.

3) Control functions associated with a UNI-C in an AGC are call control (calling/called party call controller), and resource discovery (LRM). Limited connection control and connection selection is present to interact with the connection controller on the UNI-N side. This is because the connection control on the UNI-N has a routing interface whereas connection control on the UNI-C tracks connection acceptance/release from the UNI-N side.

4) Applications that use one or more trails on an AGC are known as "<application name> connection users". They interact directly with [ITU-T G.805] access points by presenting and receiving adapted information. For each connection user there may be an "<application name> connection requestor". These entities interact with UNI-Cs to request/release connections. A single connection requestor could obtain connections from one or more UNI-Cs for a related connection user.

5) A user is considered to be multi-homed when there are two or more SNPP links connecting the AGC to the network. There is also a service agreement between the user and the network such that the network offers reliability, diversity, or other service characteristic between connections on different multi-homed SNPP links.

7 Transport resources and their representation

As described in clause 7 of [ITU-T G.7701] the transport network is a large, complex network with various components, and an appropriate network model with well-defined, technology agnostic, functional entities is essential for its design, control, and management. The transport network can be described by defining the associations between points in the network. The resultant logical network topology allows the separation between logical connections and the physical routes and resources used.

7.1 Transport functional architecture

The functional architecture of the transport network describes the way that the transport resources are used to perform the basic transport functions in a manner that makes no reference to the control and management of those functions. The functional architecture of the transport network is described in clause 7.1 in [ITU-T G.7701].

The transport resources are organized into routing areas and subnetworks for the purposes of control and management.

7.2 Domains

As introduced in clause 6, the domain notion embodied in the [ITU-T G.805] definition of administrative and management domains has been generalized, along with the notion of Internet administrative regions, to express differing administrative and/or managerial responsibilities, trust relationships, addressing schemes infrastructure capabilities, survivability techniques, distributions of control functionality, etc. A domain thus represents a collection of entities that are grouped for a particular purpose.

As a domain is defined in terms of a purpose, it is evident that domains defined for one purpose need not coincide with domains defined for another purpose. Common description on domains can be referred to clause 7.2 of [ITU-T G.7701].

An example of the relationships between components, domains and reference points is provided in Figure 7-1 which shows a domain, B, and its relationship to domains A, C and D. Each domain is derived from a component of type Z. The internal structure and interactions may be different in each domain, e.g., they may use different federation models.

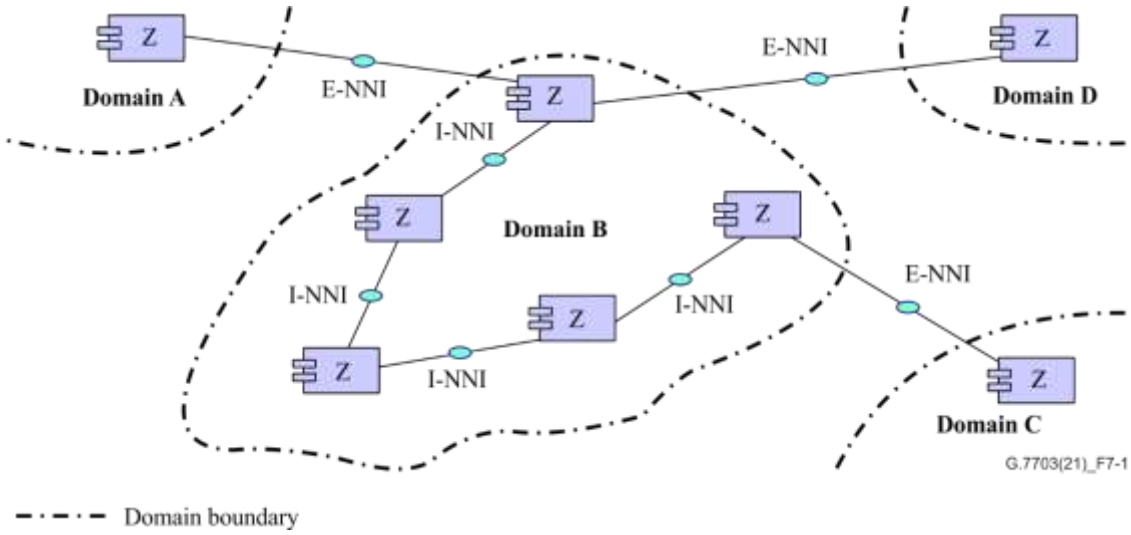


Figure 7-1 – Relationship between domains, protocol controllers and reference points

The same example is shown in Figure 7-2 with the relationships between components, domains and interfaces. The components interact via their protocol controllers, using protocol I on the I-PCs and protocol E on the E-PCs. It is also possible for the protocol used internal to A, for example, to be different from that used in B, and the protocol used between B and C to be different from that between A and B. The I-NNI interfaces are located between protocol controllers within domains whilst E-NNI interfaces are located on protocol controllers between domains.

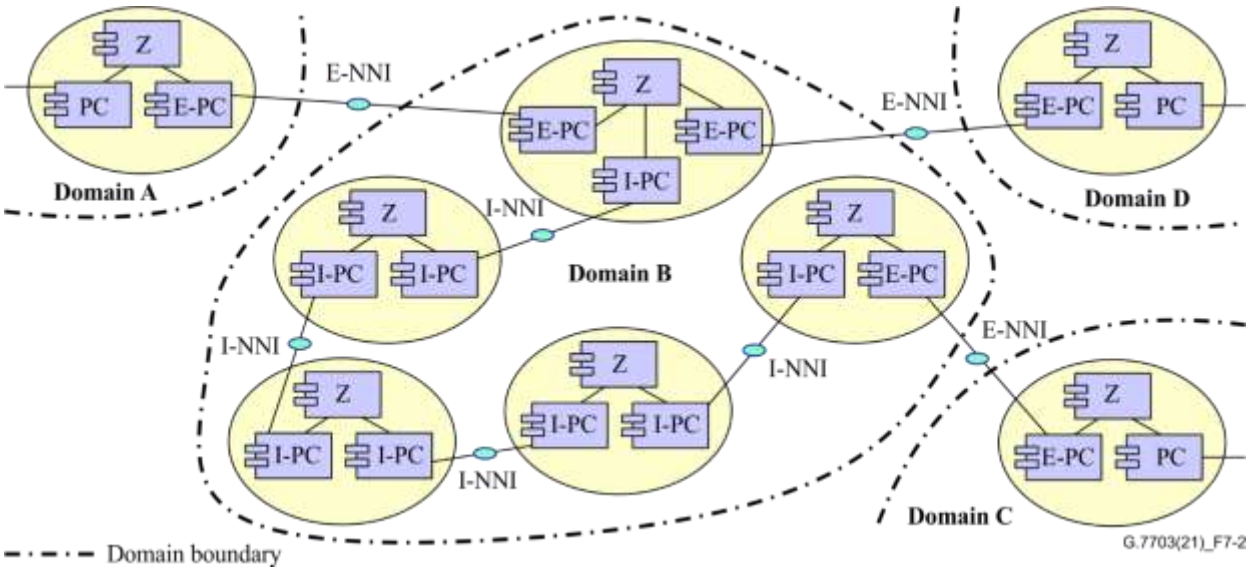


Figure 7-2 – Relationship between domains, protocol controllers and interfaces

7.2.1 Relationship between control domains and control components

The components of a domain may, depending on purpose, reflect the underlying transport network resources. A routing control domain may, for example, contain components that represent one or more routing areas at one or more levels of aggregation, depending upon the routing method/protocol used throughout the domain.

7.2.2 Relationship between control domains, interfaces and reference points

I-NNI and E-NNI interfaces are always between protocol controllers. The protocols running between protocol controllers may or may not use SNPP links in the transport network under control and as such it is incorrect to show I-NNI and E-NNI interfaces on SNPP links.

I-NNI and E-NNI reference points are between components of the same type, where the component type is not a protocol controller, and represents primitive message flows (see clause 7).

In a diagram showing only domains and the relationships between them (and not revealing the internal structure of the domains), the information transfer is assumed to be over a reference point.

7.3 Control view of transport resources for connection management

For the purpose of managing connections within a layer network, the underlying transport resources are represented by a number of entities in the MC components, termed subnetwork point (SNP) and subnetwork point pool (SNPP). The concept of transport entities and their usage in ASON network can be found in clause 7.3 of [ITU-T G.7701].

7.4 Virtualization

A generic description of virtualization is described in clause 7.4 and Figure 7-3 of [ITU-T G.7701]. Routing areas may be summarized and distributed to the other routing areas, see [ITU-T G.7715]. ~~The routing area in ASON system does not support "resource splitting", only the aggregation of routing area is supported.~~ The aggregation of routing areas in ASON system is not directly providing resources/services to the client, it is for routing purposes.

Variable adaptation functions

A number of transport systems support variable adaptation, whereby a single server layer trail may dynamically support different clients. For example, different generic framing procedure (GFP) mappings for packet clients or different multiplexing structures for SDH/OTN. From the perspective

of client layer, the server layer details are not fully visible and is usually considered to be virtual, and vice versa. The description below illustrates the application.

This situation is modelled by assigning SNPs for each framing procedure (FP) in the various structures, and placing those SNPs in their respective layer subnetworks. When a particular subnetwork point (SNP) instance is allocated, this causes the relevant client specific process in the adaptation function to be activated and creates the associated FP. SNPs in other layer networks that use the same resources become unavailable.

7.5 Multilayer aspects

Multilayer aspects are described in clause 7.5 of [ITU-T G.7701].

7.5.1 Representation as SNP link connections

The representation as SNP link connections is described in clause 7.5.1 of [ITU-T G.7701].

7.5.2 Representation as a set of SNPP links and subnetworks

The representation as a set of SNPP links and subnetworks is described in clause 7.5.2 of [ITU-T G.7701].

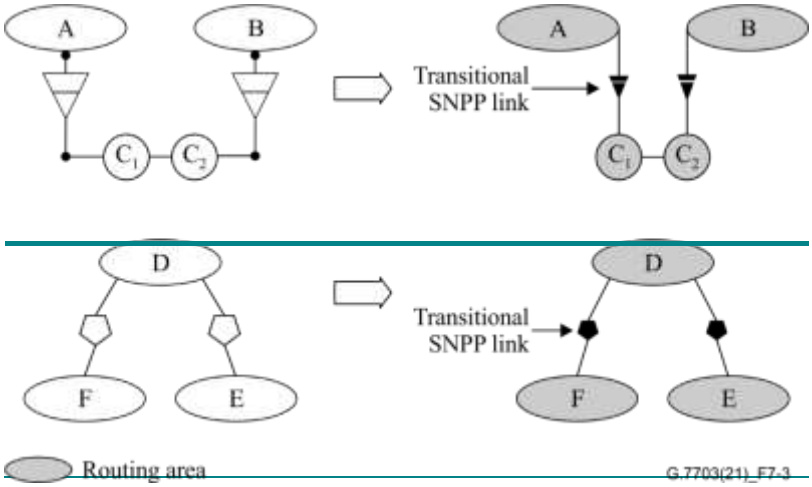
7.5.3 Multi-layer routing topology

Multi-layer topologies are described in clause 7.5.3 of [ITU-T G.7701].

~~In addition to the single layer topology representation in clause 7.5.2, a routing topology representative of a multi-layer network may be constructed using transitional SNPP links that connect routing areas in different layers. Paths may be determined that traverse link and subnetwork connections in more than one layer and/or sublayer. Control components that are involved in configuring the resulting connection configure link connections, subnetwork connections, and transitional link connections.~~

~~The use of a transitional SNPP link implies that a sequence of adaptations between layers, or sequence of layer processors within a layer, is used. The transitional SNPP link enables a connected graph to be constructed that represents a multi-layer network for the purpose of routing.~~

~~Figure 7-3 illustrates the [ITU-T G.800] representation and the corresponding multi-layer routing topologies.~~



~~Figure 7-3 – Multi-layer routing topology representations~~

~~A multi-layer topology is illustrated in Figure 7-4 on the left.~~

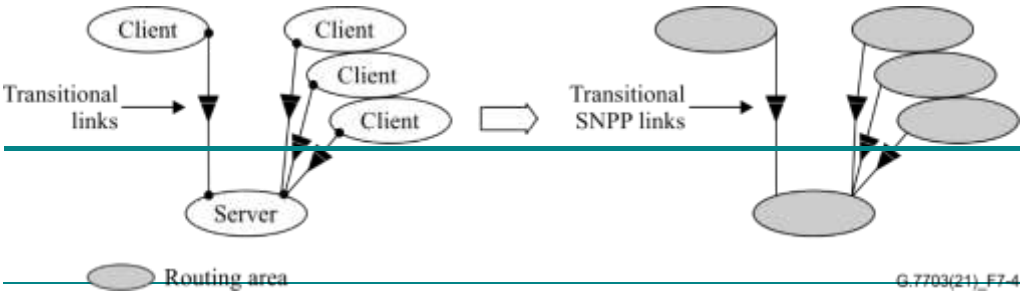


Figure 7-4 – Multi-layer routing topology representation

A path computation begins by selecting two (sets of) points at the boundary of a routing area. The path computation may be accomplished using a set of subordinate path computations involving contained routing areas. The routing areas relevant to the path computation are selected by the path computation algorithm, and these routing areas may belong to one or more layers (i.e., they may be part of a multi-layer topology).

In a multi-layer routing topology, when a transitional SNPP link is used between a routing area in a first layer to another routing area in an adjacent second layer and the first layer is traversed further in a route, it is expected that a corresponding transitional link is used to eventually return to the first layer in computed paths.

For segments of a path computed between sublayers, traversal of a single transitional SNPP link is allowed.

An additional example of a multi-layer topology is shown in Appendix IV.

NOTE – Use of transitional links for a 1:1 adaptation is described in this clause.

7.6 Interlayer client support

The support of a client in the inter-layer case is described in Clause 7.6 of [ITU-T G.7701]. The NCC interactions in an inter-layer scenario is described as follow:

7.6.1 NCC to NCC calls in ASON networks

In ASON networks, NCC-NCC relationships are described in the clause 8.3.1 of [ITU-T G.7701] as peering or hierarchical. A call may exist between a pair of NCCs in the absence of calling party call controllers (CCCs). This call is within the same layer. To allow such calls to be requested, a BRI is assigned to a set of SNPs that reference resources that may be used to support a call. This is analogous to the UNI BRI associated with transport resources at the UNI.

When used in an interlayer call, a client NCC is used to initiate the call between another pair of NCCs in a server layer. As the client layer NCC invokes the server layer call, a domain boundary is crossed. This domain boundary is instituted to provide a policy control point, as well as provide separation of the SNPP as well as BRIs used in the client and server layers. The BRIs used for the call request are in the server layer. This is illustrated in Figure 7-53.

A server layer connection from an NCC to NCC call used to support a mapped client CI, has an association with an adaptation. Such a call/connection may exist before the adaptation is actually used.

7.6.2 Name space interactions

UNI BRIs are defined to be globally unique in ASON networks. The additional BRIs defined are not required to be part of the UNI BRI space. BRIs associated with NCC to NCC calls may be from separate identifier spaces.

Within a single layer network, there may be independent SNPP identifiers spaces. Connections can be created across these different SNPP identifier spaces due to the fact that RCs understand the mapping of SNPP identifiers between routing levels.

When two SNPP identifier spaces are not mapped via routing, it is allowed to map one SNPP identifier space into the BRIs associated with the second SNPP identifier space. One purpose for this would be for a business boundary that has no routing exchange. An interlayer boundary is such a case.

In Figure 7-~~5~~3 below, SNPP-X in the client layer is mapped to BRI B and SNPP-Y is mapped to BRI C. An interlayer call can be invoked using the returned BRI.

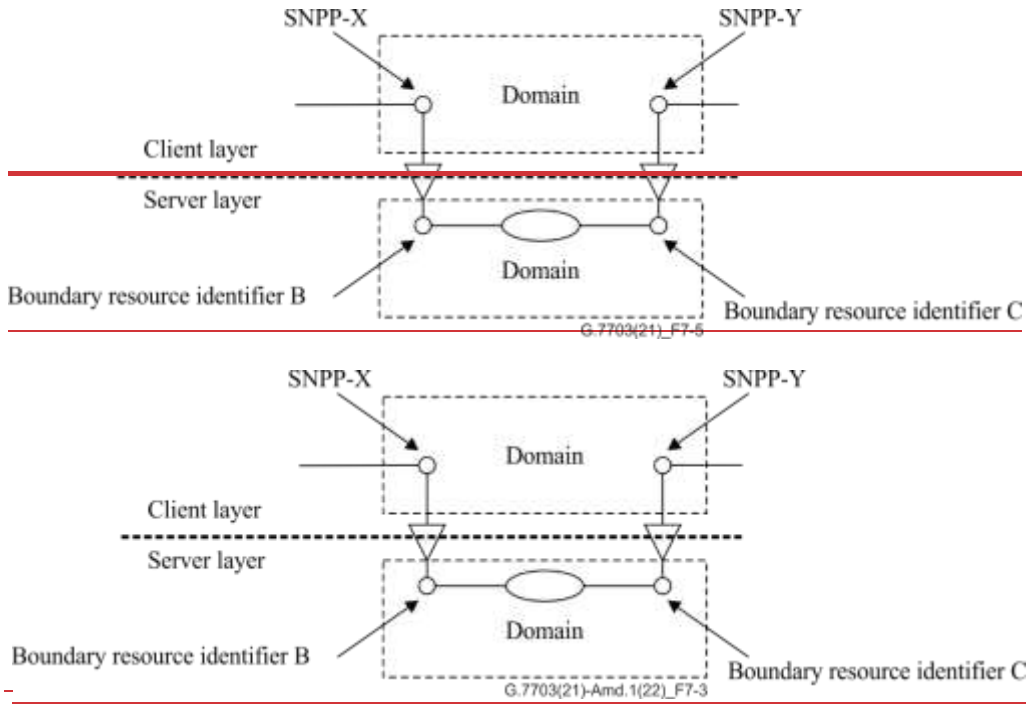


Figure 7-~~5~~3 – Name space interactions

7.7 Calls supported by calls at same layer

The calls supported by calls at the same layer can be found in clause 7.7 in [ITU-T G.7701].

7.8 Mapped server interlayer relationships

The mapped server interlayer relationships can be found in clause 7.8 in [ITU-T G.7701].

8 Control components

General description of control components can be found in clause 8 in [ITU-T G.7701].

8.1 Notation

The generalized notation can be found in clause 8.1 of [ITU-T G.7701].

8.2 Policy and federations

8.2.1 Policy in ASON networks

ASON networks use the policy model described in clause 8.2 of [ITU-T G.7701].

8.2.2 Federation in ASON networks

The creation, maintenance, and deletion of connections across multiple domains is required. This is achieved by cooperation between controllers in different domains. For the purposes of this Recommendation, a federation is considered a community of domains that cooperate for the purposes of connection management, and is illustrated using the cooperation between connection controllers. (Connection controllers are described in clause 8.3.2.)

There are two types of federation:

– joint federation model

– cooperative model.

In the joint federation case one connection controller, the parent connection controller, has authority over connection controllers that reside in different domains. Where a connection is required that crosses multiple domains the highest-level connection controller (the parent) acts as the coordinator. This connection controller has knowledge of the highest-level connection controllers in each domain. The parent connection controller divides the responsibility for the network connection between the next level connection controllers, with each responsible for its part of the connection. This is illustrated in Figure 8-1. This model is recursive with a parent connection controller at one level being a child to a parent at a higher level.

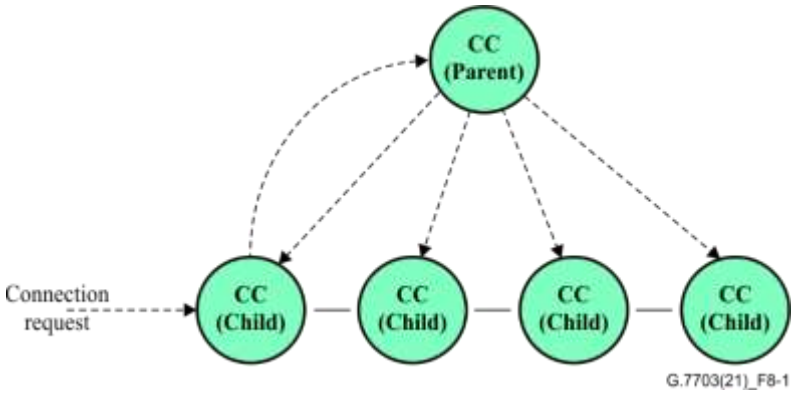


Figure 8-1 – Joint federation model

In the cooperative model, there is no concept of a parent connection controller. Instead, when a connection request is made, the originating connection controller contacts each of the connection controllers associated with domains of its own volition and there is no overall coordination. The simplest method of achieving this is for the originating connection controller to contact the next connection controller in the chain. This is illustrated in Figure 8-2, where each connection controller calculates what part of the connection it can provide and what the next connection controller will be. This continues until the connection is provided.

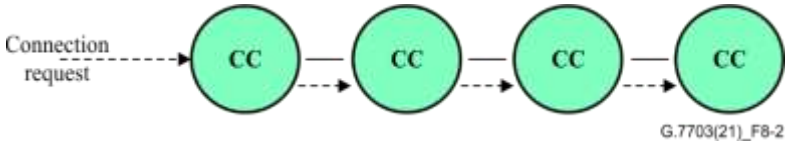


Figure 8-2 – Cooperative federation model

Federation between administrative domains is by means of the cooperative model. In this case, all administrative domains are expected to have the capability to federate with other administrative domains. Parent connection controllers within an administrative domain may federate with other parent connection controllers in other administrative domains by means of the cooperative model.

An administrative domain may also be subdivided, and the choice of federation model employed between domains within an administrative domain can be independent of what happens in another administrative domain. It is therefore possible to combine both federation models to construct large networks as illustrated in Figure 8-3. The principle described above can also be applied to federations of call controllers.

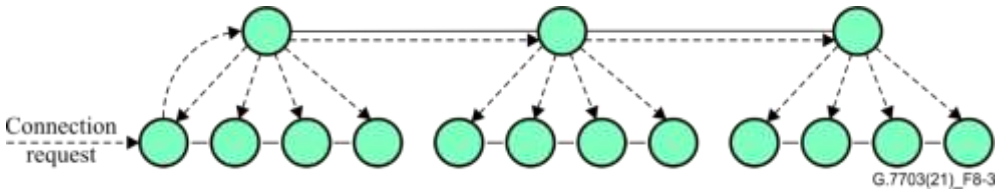


Figure 8-3 – Combined federation model

8.3 Architectural components

The components of the control architecture are described in this clause. Components can be combined in different ways, depending upon the required functionality. Appendix III illustrates examples of interactions of these components for use in connection set-up. Each component is described by a brief description of its primary function in this reference architecture. Component interfaces are provided next, and a more detailed description of operation is then given.

The connection controller, routing controller, calling/called party call controller, and network call controller are MC components. These components are either public, in which case they use public SNPPs only, or private, in which case they use the SNPPs associated to a particular VPN. The VPN context of a MC component is provided by the protocol controller associated with that component.

8.3.1 Call controller components

Call controller components are described in clause 8.3.1 of [ITU-T G.7701].

8.3.1.1 Calling/called party call controller

The calling/called party call controller interacts with the network call controller and is described in clause 8.3.1.1 of [ITU-T G.7701].

~~The role of this component is:~~

- ~~generation of outgoing call requests~~
- ~~acceptance or rejection of incoming call requests~~
- ~~generation of call termination requests~~
- ~~processing of incoming call termination requests~~
- ~~call state management.~~

~~This component has the interfaces provided in Table 1. The calling/called party call controller component is illustrated in Figure 8-4.~~

~~Table 1 – Calling/called party call controller component interfaces~~

~~Input interface~~	~~Basic input parameters~~	~~Basic return parameters~~
~~Call accept~~	~~Transport resource identifier, VPN transport resource identifier or call name~~	~~Confirmation or rejection of call request~~
~~Call release in~~	~~Transport resource identifier or VPN transport resource identifier~~	~~Confirmation of call release~~

Call modification accept	Call name, parameters to change	Confirmation or rejection of call modification

Output interface	Basic output parameters	Basic return parameters
Call request	Transport resource identifier or VPN transport resource identifier; route (optional, for VPN only)	Confirmation or rejection of call request
Call release out	Transport resource identifier or VPN transport resource identifier	Confirmation of call release
Call modification request	Call name, parameters to change	Confirmation or rejection of call modification

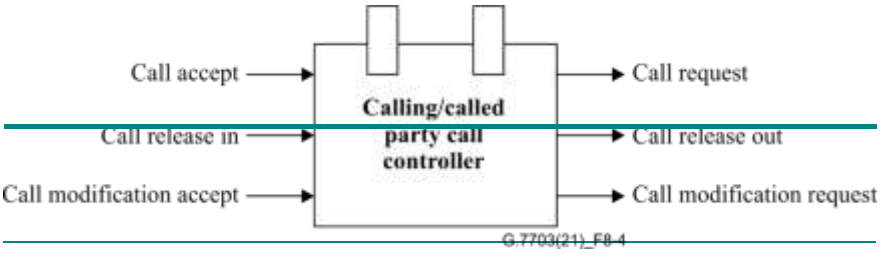


Figure 8-4 – Calling/called party call controller component

Call request: This interface is used to place requests for set-up, maintenance and release of a call. This interface also accepts a confirmation or rejection of a call request.

Call accept: This interface is used to accept incoming call requests. It also confirms or rejects the incoming call request.

Call release: This interface is used to place, receive and confirm release requests.

Call modification request: This interface is used to place requests to modify an existing call. It also receives the confirmation or rejection of the request.

Call modification accept: This interface is used to accept incoming requests to modify an existing call. It also confirms or rejects the request.

Note that the same calling/called party call controller may play the role of originator or terminator in different transactions.

8.3.1.2 Network call controller

Network call controller is described in clause 8.3.1.2 of [ITU-T G.7701].

8.3.1.3 Call controller interactions

The interaction between call controller components is dependent upon both the type of call and the type of connection, as described below.

Switched connections: The calling party call controller (associated with an end terminal) interacts with the network call controller to form an incoming call and the network call controller interacts with the called party call controller (associated with an end terminal) to form an outgoing call. The network call controller interacts with the connection controllers to provide the call. An example of this interaction is illustrated in Figure 8-54. It should be noted that the calling/called party call controllers have no direct interaction with the connection controller associated with the corresponding network call controller.

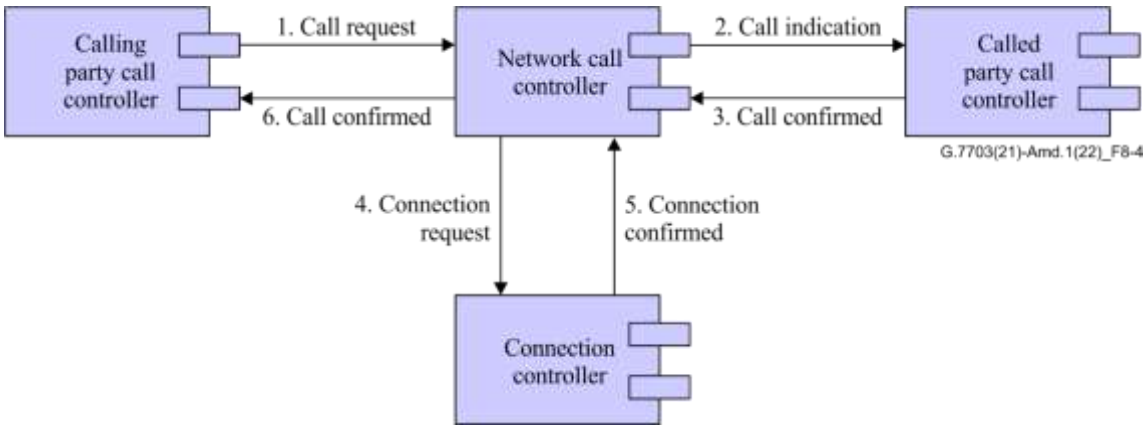


Figure 8-~~5~~ 4 – Called/calling party call controller interaction for switched connections: Example 1

Figure 8-~~5~~ 4 shows the situation whereby the called party call controller accepts the call, prior to the ingress network call controller requesting the connection. It is also valid to define the interaction such that the connection set-up follows the call, as is illustrated in Figure 8-~~6~~5.

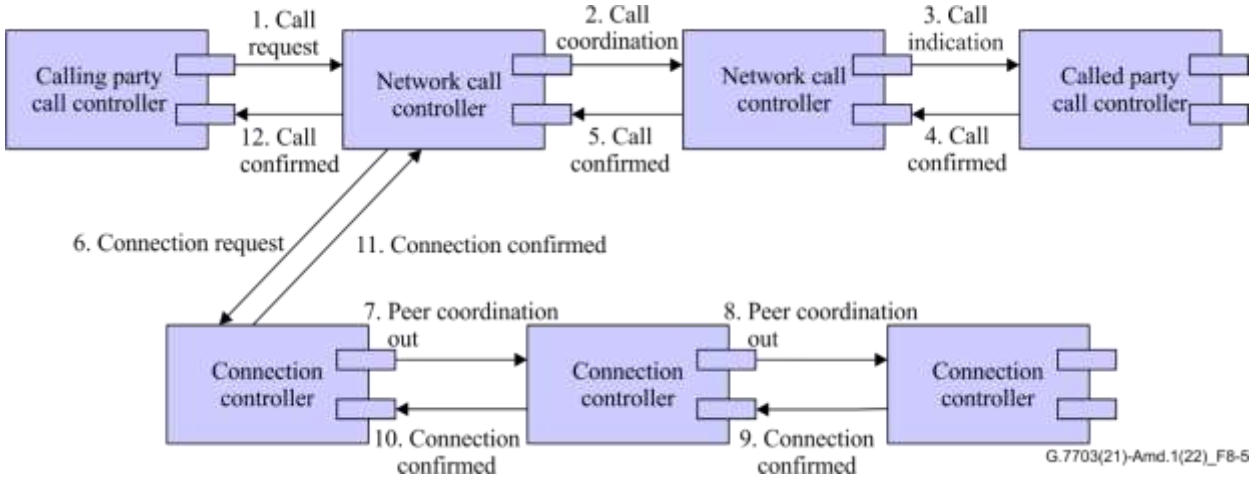


Figure 8-~~6~~ 5 – Called/calling party call controller interaction for switched connections: Example 2

Soft permanent connections: The calling/called party controllers are supported by another MC system. The MC system issues a command to configure the calling party call controller that initiates the network call controllers in the ASON control domain when the call configuration commands are sent to the ASON control domain. The response to a call configuration command from the ASON control domain is considered as a call set-up confirmation by the MC system that initiated the call. This represents a null call with no service. The protocols between the MC system and the ASON control domain are a command and command response interface. Figure 8-~~7~~ 6 illustrates the call controller interactions for soft permanent connections.

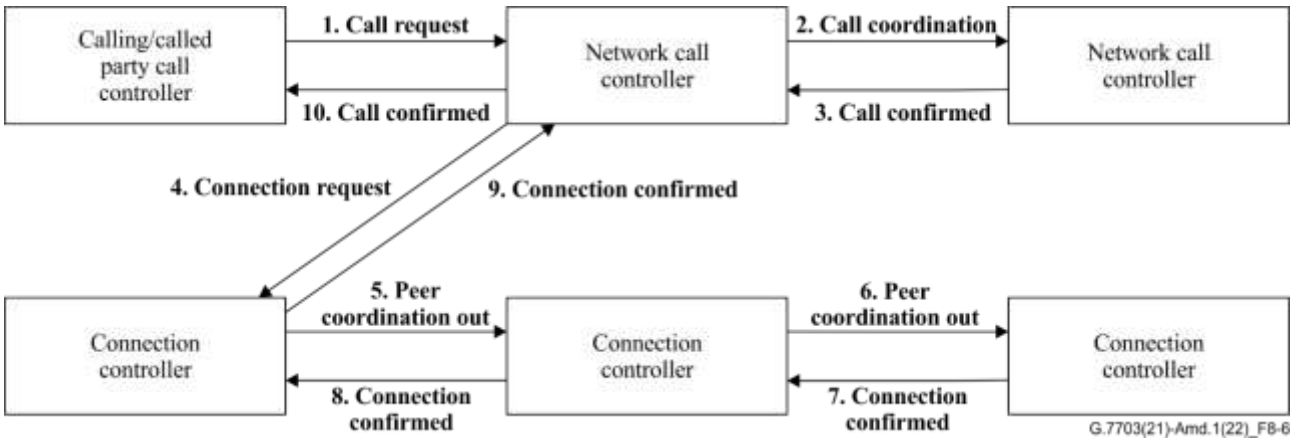


Figure 8-~~7~~ 6 – Call controller interactions for soft permanent connections

Proxy call: The calling/called party call controller interacts with the network call controller by means of a call protocol, but is not coincident with the user.

Figure 8-~~8~~ 7 indicates an example of the interactions necessary to support call admission control policy between network call controllers.

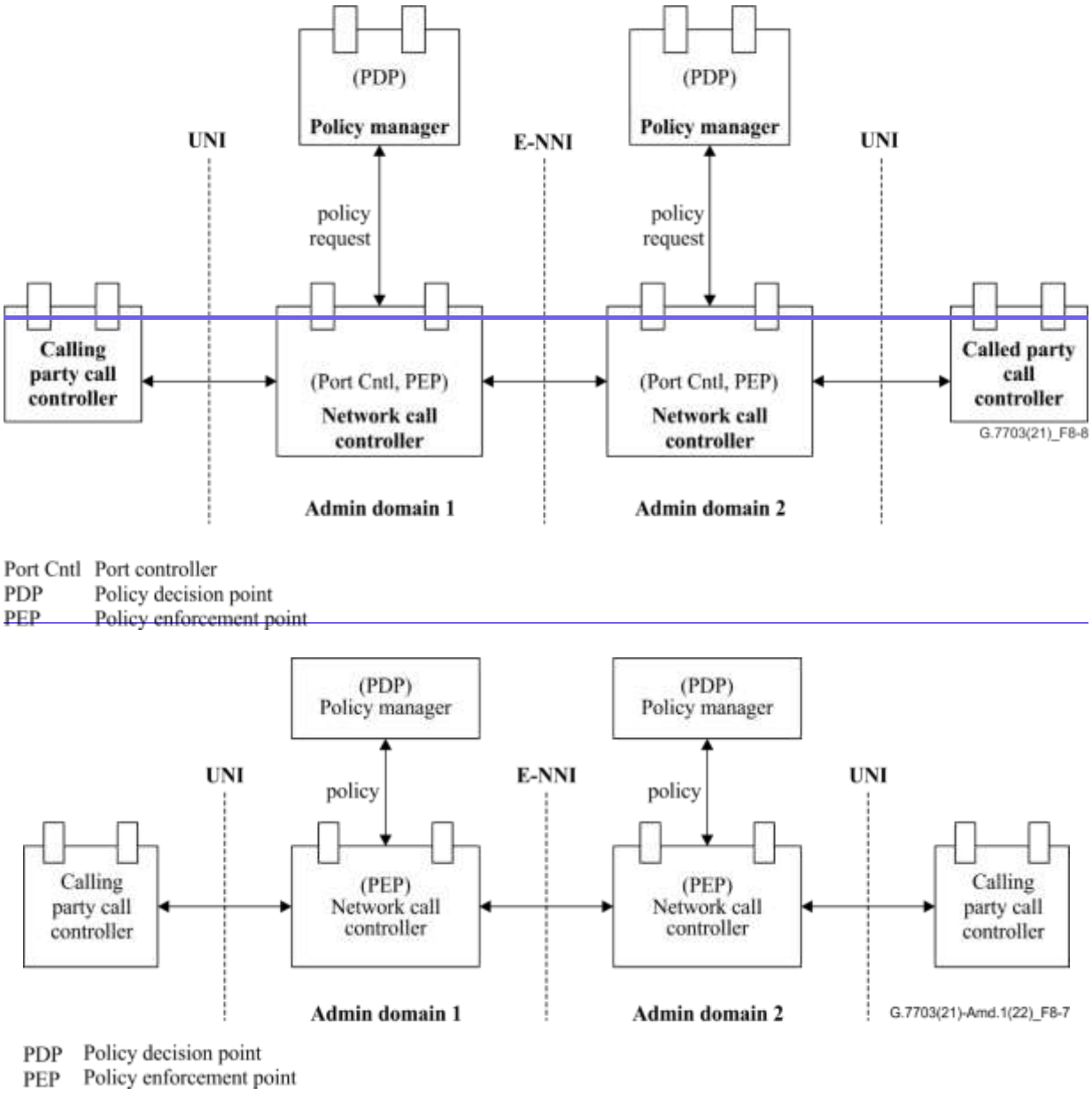


Figure 8-~~8~~ 7 – Example of call admission control policy interactions

Layered calls: Two NCCs in different layers may cooperate to allow support of client CI in a server layer. Use of these interlayer interfaces is governed by operator policy. This may be initiated either to or from a server layer depending on what layer the operation is initiated from. From an NCC, the request to a server layer NCC returns the same result as the "Connection Request Out" interface. The difference is that an association with a server NCC is made. This action either results in the use or creation of, a server layer call segment that will support the client NCC. If the server layer needs to create a call as a result of the use of the "Server NCC Coordination Out" or "Client NCC Coordination In" interfaces, the source and destination identifiers are used as call parameters. An identical action to the "Call Request Accept" interface behaviour is then performed if connection establishment at that server layer is determined to be the correct action. The server layer NCC could alternately use its "Server NCC Coordination Out" interface to make a (layer recursive) request for an SNP pair from another layer NCC that is a server to it.

An NCC could also initiate an action to a client layer whereby it presents a pair of SNPs that can be used by the client layer for transferring client CI. The "Client NCC Coordination Out" or "Server NCC Coordination In" interfaces are used for this purpose. When this interface is used, the SNP pair

presented is able to transfer client CI and no call action at the server layer is initiated. This is used for an operation where a server layer has already established a call, and this is presented to the client layer at a later point in time. The client layer may accept or reject the use of the offered SNP pair.

8.3.1.4 Call modification

The service provided by a call can be modified by actions initiated by a CCC or network management application acting on an NCC at the ~~UNI~~reference point. The degree of modification is set by operator policy and the policy may or may not be shared with the end user (e.g., informing the user of what bandwidth increments are allowed). The extent to which a call can be modified is subject to the following rules:

- The CI associated with the call at the reference point~~UNI~~ is not modifiable.
- The link connection end-points associated with the call at the reference point~~UNI-N~~ are not modifiable. They may be added/removed however when connections are added/removed from a call.

Actions can either be modification of a call segment where the NCCs remain fixed, or the creation/deletion of call segments within an overall call where NCCs are created/deleted.

Examples of what may be modified at the reference point ~~UNI~~ include bandwidth (e.g., rate of Ethernet call) and number of CCCs involved (e.g., multi-party call).

Examples of what may occur within the network as a result of reference point ~~UNI~~ call modification requests include:

- changing the number of server layer connections associated with a VCAT call that supports an Ethernet call.
- In response to a request to increase the availability of a call, adding an additional connection to create a 1+1 configuration.

In the ASON architecture, the reference points above are the UNI and ENNI.

8.3.1.5 Call failure handling

For a new call request, if the network is unable to establish all the connections required to satisfy the call request, any connections or partial connections that have been established will be torn down (deleted) and the call request will be rejected.

For call modifications, if the network is unable to add the connections requested, then the call modification is considered to have failed. Any connections or partial connections will be removed, and no changes will be made to the existing call.

8.3.2 Connection controller (CC) component

The connection controller (CC) component is described in clause 8.3.2 of [ITU-T G.7701].

8.3.3 Routing controller (RC) component

The routing controller (RC) component is described in clause 8.3.3 of [ITU-T G.7701].

8.3.4 Link resource manager (LRM)

The generic link resource manager (LRM) component is described in clause 8.3.4 of [ITU-T G.7701].

In ASON, two LRM components are used – the LRMA and LRMZ. An SNPP link is managed by a pair of LRMA and LRMZ components, one managing each end of the link. Requests to assign SNP link connections are only directed to the LRMA. If the required link resources are not available, then the LRM must either request additional capacity from the TAP or reject the connection request.

The two cases for SNPP link are illustrated in Figure 8-~~9~~8.

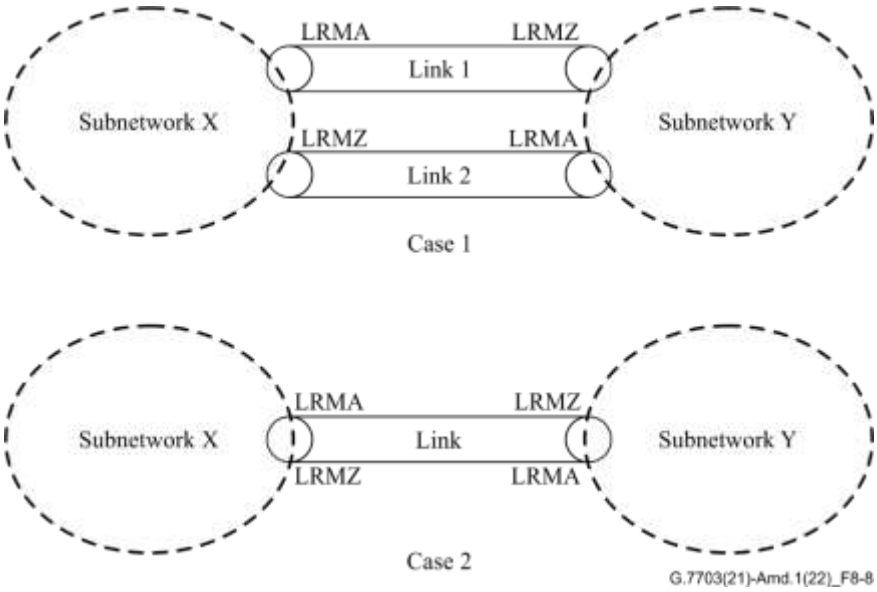


Figure 8-~~9~~ 8 – SNPP link cases

In case 1, link 1 is dedicated to connection set-up requests originating from subnetwork X. Requests for SNP link connections from subnetwork X are directed to the adjacent LRMA for link 1, which can process the request without negotiation with the far end of the link. This LRMA can assign the SNP identifier and capacity (and hence the link connection) without negotiation with the LRMZ for link 1. Similarly, link 2 is dedicated to connection set-up requests originating from subnetwork Y. Requests for SNP link connections from subnetwork Y are directed to the adjacent LRMA for link 2. This LRMA can assign the SNP identifier without negotiation with the LRMZ for link 2. In this case, the same SNP identifier is used for both directions of transmission in a bidirectional connection. For a packet-switched network, the bandwidth assigned to a bidirectional connection may be asymmetric and must be tracked, by LRMA, independently for each direction. Also, for packet-switched networks, the LRMA and LRMZ, in addition to assigning the SNP identifier, must communicate with the TAP to configure the policing and shaping functions.

In case 2, the link is shared between subnetworks X and Y for connection set-up. Requests for SNP link connections from subnetwork X are directed to the adjacent LRMA, since an LRMA component at the far end of the link can independently allocate SNP identifiers and link resources, the LRMA may need to negotiate an SNP identifier and capacity assignment with the LRMA at the far end (via the LRMZ at the far end). A similar process is required for request from subnetwork Y to its adjacent LRMA. Case 2 can be broken down into three sub-cases:

a) The same SNP identifier is used for both directions of a bidirectional connection.

b) The SNP identifiers are assigned independently for each direction at the source end of the link.

c) The SNP identifiers are assigned independently for each direction at the sink end of the link.

8.3.4.1 LRMA

The LRMA is responsible for the management of the A end of the SNPP link as described below.

The LRMA component interfaces are provided in Table 2 and illustrated in Figure 8-~~10~~9.

Table 2 – LRMA component interfaces

Input interface	Basic input parameters	Basic return parameters
Connection request	Request id SNP ~~id~~ ID (optional) CIR and EIR (packet switched only)	Request id SNP ~~id~~ ID pair or denied
Connection deletion	SNP Id; or request id	Confirm or denied
Configuration	Link information	–
Translation	Local id	Interface id
Connection modification	SNP Id; or request id New CIR and EIR (packet switched only)	Confirm or deny
SNP binding state	Busy, potential, allocated, shutting down	Resource released (in response to the shutting down state)
SNP operational state	Enabled, disabled	
Add SNP	List of SNP identifiers	confirm
Withdraw SNP	List of SNP identifiers	confirm
Output interface	**Basic output parameters**	**Basic return parameters**
Assign SNP (Case 1 only)	SNP id	confirm
SNP negotiation (Case 2 only)	Request id List of SNP ids CIR and EIR (packet switched only)	Request id SNP id
SNP release (unassign)	List of SNP id	Confirm
Topology	Link information	–
SNP ~~id~~ ID assignment/ unassignment (packet switched only)	SNP id CIR and EIR (to TAP)	Confirm or deny (TAP must bind the SNP to the resource label and configure the policing functions)
Capacity change request (packet switched only)	SNP id CIR and EIR	Link configuration

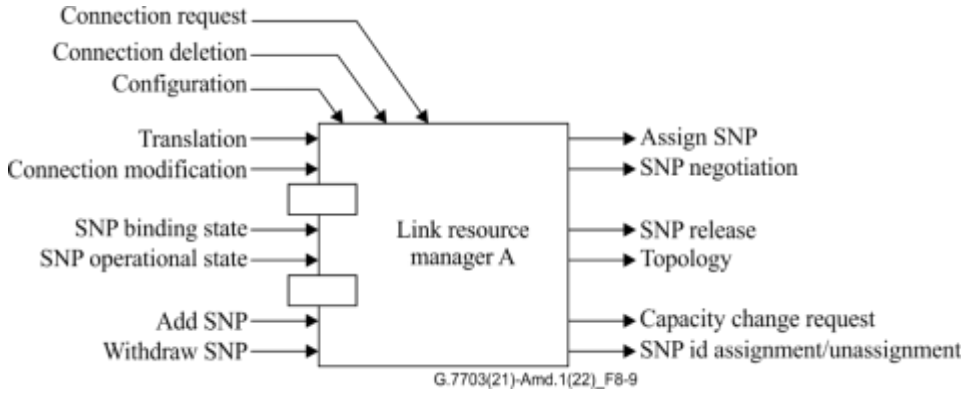


Figure 8-~~10~~ 9 – Link Resource Manager A component

- **Functions**

– *Assignment of a link connection to a connection*

When a connection request is received, connection admission is invoked to decide if there is sufficient free resource to allow a new connection. Connection admission can also be decided based on prioritization or on other policy decisions. Connection admission policies are outside the scope of standardization.

For the purposes of the description below, the designations x and y are added to the LRM components to clarify the location and role.

If there are insufficient local resources, the request is rejected or the local LRMAx may request the TAP to convert an SNP with a binding state of potential and the associated potential capacity to allocated (if the policy allows such requests).

If sufficient local resources are available, the connection request is allowed to process as described in the cases below. Note that, for circuit-switched networks, the local LRMAx can determine if sufficient capacity is available since the bandwidth is always symmetric and is implicit in the availability of an SNP. For packet-switched networks, since the bandwidth may be asymmetric and must be tracked explicitly, it is possible that there are insufficient far-end resources available to support a connection, in which case the connection request will be rejected.

– Case 1 circuit-switched layer network: Since the SNP identifiers (and the corresponding link connections) are only assigned from one end of the SNPP link, the LRMAx can select the SNP identifier without interaction with the LRMZy at the far end of the link. The LRMAx passes the SNP identifier to the connection controller.

– Case 1 packet-switched layer networks: Since the link resources are only assigned from one end of the link, the LRMAx can perform the admission control and bandwidth reservation process without interaction with LRMZy at the far end of the link. LRMAx selects the SNP identifier, configures the policing and shaping functions and informs LRMZy. LRMZy must assign the SNP ~~id~~ ID and communicate with the TAP to configure the policing and shaping functions. LRMAx tracks the capacity assigned to connections for both directions of transmission on the link. The LRMAx passes the SNP identifier to the connection controller.

– Case 2a circuit-switched layer networks: Since the SNP identifiers (and the corresponding link connections) may be used by the LRMA at either end of the SNPP link, the LRMAx passes a list of usable SNP ids to the LRMZy. The LRMZy (in cooperation with its local LRMAy) selects one of the SNPs and returns the ~~id~~ ID to the originating LRMAx. The originating LRMAx passes the SNP identifier to the connection controller.

– Case 2a packet-switched layer networks: LRMAx adds the requested capacity for the A to Z direction of transmission on the link to its local copy of the link capacity assignment. Since the resources are assigned independently by the LRMA at either end of the SNPP link and SNPs are assigned from a common pool, the LRMAx passes a list of the useable SNP ids and the bandwidth parameters for the Z to A direction of transmission to the LRMZy at the far end of the link. The LRMZy passes this information to the local LRMAy which confirms that the link capacity is available, adds this to its local copy of the assigned link capacity and selects and assigns an SNP identifier and communicates with the TAP to configure the policing and shaping functions. If the available resources are insufficient to support the connection, the request is rejected; or the LRMAy may request additional resources from the TAP. This information is returned to the originating LRMAx. If the request has been accepted by the remote LRMAy, the local LRMAx assigns the SNP, communicates with the TAP to configure the policing and shaping functions. The LRMAx passes the SNP identifier to the connection controller. If the request is denied by the remote LRMAy, the local LRMAx rejects the connection request and removes any local reservations.

– Case 2b packet-switched layer networks: LRMAx adds the requested capacity for the A to Z direction of transmission on the link to its local copy of the link capacity assignment, and selects an SNP identifier for the A to Z direction of transmission. Since the resources are assigned independently by the LRMA at either end of the SNPP link, the LRMAx passes the selected SNP (for the A to Z) and the bandwidth parameters for the Z to A direction of transmission to the LRMZy at the far end of the link. The LRMZy passes the bandwidth requirements to the local LRMAy which confirms that the link capacity is available, adds this to its local copy of the assigned link capacity, and selects and assigns an SNP identifier (from its local pool for the A to Z direction of transmission) and communicates with the TAP to configure the policing and shaping functions. If the available resources are insufficient to support the connection, the request is rejected; or the LRMAy may request additional resources from the TAP. This information is returned to the local LRMZy which then assigns the SNP provided by the remote LRMAx and passes the information to the remote (originating) LRMAx. If the request has been accepted by the remote LRMAy, the local LRMAx assigns the SNP (for the A to Z direction of transmission), provides the Z to A SNP to the local LRMZx and communicates with the TAP to configure the policing and shaping functions. The LRMAx passes the SNP identifiers to the connection controller. If the request is denied by the remote LRMAy, the local LRMAx rejects the connection request and removes any local reservations.

– Case 2c packet-switched layer networks: LRMAx adds the requested capacity for the A to Z direction of transmission on the link to its local copy of the link capacity assignment and selects an SNP identifier for the Z to A direction of transmission. Since the resources are assigned independently by the LRMA at either end of the SNPP link, the LRMAx passes the selected SNP and the bandwidth parameters for the Z to A direction of transmission to the LRMZy at the far end of the link. The LRMZy passes the bandwidth requirements and SNP to the local LRMAy which confirms that the link capacity is available, adds this to its local copy of the assigned link capacity and selects an SNP identifier (from its local pool for the Z to A direction of transmission), assigns the SNP identifier provided by the remote LRMAx and communicates with the TAP to configure the policing and shaping functions. If the available resources are insufficient to support the connection, the request is rejected; or the LRMAy may request additional resources from the TAP. This information is returned to the local LRMZy which then assigns the SNP provided by the local LRMAy and passes the information to the remote (originating) LRMAx. If the request has been accepted by the remote LRMAy, the local LRMAx assigns the SNP (for the A to Z direction of transmission), provides the Z to A SNP to the local LRMZx, communicates with the TAP to configure the policing and shaping functions. The LRMAx passes the SNP identifiers to the connection controller. If the request is denied by the remote LRMAy, the local LRMAx rejects the connection request and removes any local reservations.

– *Deletion of a connection*

  Case 1: When a request to delete a connection is received, the corresponding SNP is marked as unassigned, and the corresponding resources are removed from the assigned link capacity. The associated LRMZy is informed so that it can release the SNP identifier.

  Case 2: When a request to delete a connection is received, LRMAx marks the corresponding SNP identifier as unassigned, and the corresponding resources are removed from the assigned link capacity. Both the local LRMZx and the LRMZy at the far end of the link are informed. The LRMZ releases the SNP identifier. The remote LRMZy passes the request to its local LRMAy which marks the SNP identifier as unassigned and removes the resource reservation.

– *Interface to local ~~id~~ ID translation*

If required, the LRM provides the translation of an interface ~~id~~ ID to a local id. This is used, for example, if the ends of the SNPP link are in different routing areas.

- **Topology**

This function provides the link topology using the interface SNPP ids; the allocated SNP ids; assigned SNP ids; allocated capacity (packet switched only); assigned capacity (packet switched only).

It also provides link characteristics, e.g., link cost, diversity and quality. Some characteristics, for example link cost, may vary with link utilization. The process used to modify link characteristics is controlled by a local policy.

8.3.4.2 LRMZ

The LRMZ is responsible for the management of the Z end of the SNPP link as described below.

The LRMZ component interfaces are provided in Table 3 and illustrated in Figure 8-1~~1~~0.

Table 3 – LRMZ component interfaces

Input interface	Basic input parameters	Basic return parameters
SNP assignment	SNP id CIR and EIR (packet switched only)	Confirmation
SNP negotiation In (Case 2 only)	Request id List of SNP ids CIR and EIR (packet switched only)	Request id SNP ~~id~~ ID or denied
SNP unassignment	SNP id	Confirmation
Configuration	Link information	–
Translation	Local id	Interface id
Output interface	**Basic output parameters**	**Basic return parameters**
Topology	Link information	–
SNP ~~id~~ ID assignment/unassignment	(Packet switched only) SNP id	

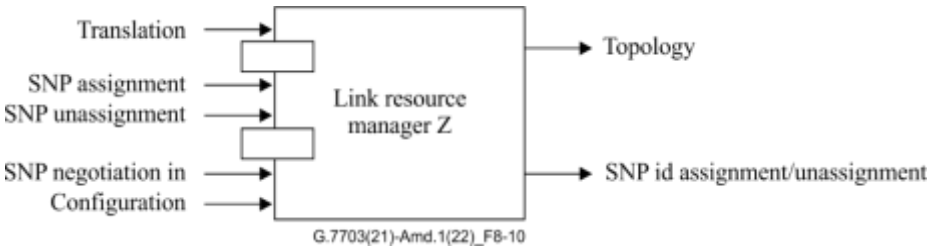


Figure 8-~~11~~ 10 – Link resource manager Z component

- **Functions**

– *Assignment of SNP identifiers (case 1 only)*

When the remote LRMAx requests LRMZy to assign an SNP, LRMZy implements the request and, in the case of packet-switched networks, it also informs the TAP which configures the shaping and policing function (if required).

– *Negotiation and assignment of SNP (only used for case 2)*

- Case 2a circuit-switched networks: When a list of usable SNP ids is received from the remote LRMAx, one is selected (by the local LRMAy) and returned.
- Case 2a packet-switched networks: When a list of usable SNP ids and connection bandwidth parameters are received, the local LRMAy is informed. If sufficient capacity is available, the local LRMAy selects and returns an SNP identifier to the LRMZy. The

LRMZy assigns this SNP identifier and informs the originating LRMAx. If the local LRMAy determines that the available link capacity is not sufficient, the request is denied.

- Case 2b packet-switched networks: When an SNP ~~id~~ ID and connection bandwidth parameters are received, the local LRMAy is informed. If sufficient capacity is available, the local LRMAy selects and returns an SNP identifier to the LRMZy. The LRMZy assigns the SNP identifier provided by the remote LRMAx and returns the SNP ~~id~~ ID provided by the local LRMAy to the remote (originating) LRMAx. The originating LRMAx provides this SNP ~~id~~ ID to its local LRMZx so that it can be assigned. If the local LRMAy determines that the available link capacity is not sufficient, the request is denied.
- Case 2c packet-switched networks: When an SNP ~~id~~ ID and connection bandwidth parameters are received, the local LRMAy is informed. If sufficient capacity is available, the local LRMAy selects and returns an SNP identifier to the LRMZy. The LRMZy assigns this SNP identifier and returns it to the remote (originating) LRMAy. The originating LRMA then provides the SNP ~~id~~ ID to the local LRMZx so that it can be assigned. If the local LRMAy determines that the available link capacity is not sufficient, the request is denied.

– *Unassignment of SNP identifiers in case 1*

When the associated LRMAx indicates that an SNP has been unassigned, the corresponding SNP identifier in LRMZy is marked as available.

– *Unassignment of SNP identifier (only used for case 2)*

When the associated LRMAx indicates that an SNP has been unassigned, the SNP is marked as available. The local LRMAy is also informed.

– *Interface to local ~~id~~ ID translation (case 1 only)*

If required, the LRM provides the translation of an interface ~~id~~ ID to a local id. This is used, for example, if the ends of the SNPP link are in different routing areas.

Topology (case 1 only)

This function provides the link topology using the interface SNPP ids; allocated SNP ids; assigned SNP ids; allocated capacity (packet switched only); assigned capacity (packet switched only).

8.3.5 Discovery agent (DA) and link discovery process

The common description for the discovery agent (DA) component is provided in clause 8.3.5 of [ITU-T G.7701].~~The federation of discovery agents (DAs) operates in the [ITU-T G.800] FP name space, and provides for separation between that space and the control names. The federation has knowledge of connection points (CPs) and termination connection points (TCPs) in the network, while a local DA has knowledge of only those points assigned to it. Discovery coordination involves accepting potential hints about pre-existing CPs and link connections. The DA holds the CP-CP link connections to enable SNP-SNP link connections to be bound to them later. The resolution interfaces assist in discovery by providing name translation from global TCP handles to the address of the DA responsible for the point, together with the local name of the TCP. Note that hints come from cooperation with other components, or from external provisioning systems.~~

~~Discovery agents have no private equipment interfaces, and can be located on any suitable platform. Detailed input and output parameters can be found in Table 4.~~

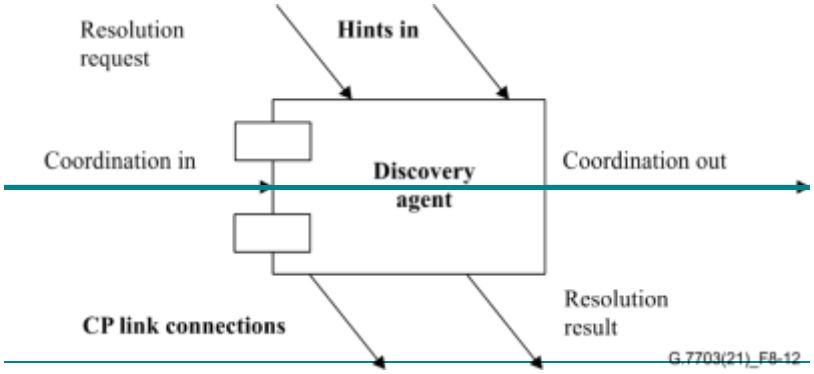
~~Figure 8-12 illustrates the components of the discovery agent.~~

~~Table 4 – Discovery agent (DA) component interface~~

~~Input interface~~	~~Basic input parameters~~	~~Basic return parameters~~

~~Coordination in~~		
~~Hints in~~	~~CP pairs~~	
~~Resolution request~~	~~TCP name~~	

~~Output interface~~	~~Basic output parameters~~	~~Basic return parameters~~
~~Coordination out~~		
~~CP link connection~~	~~CP pair~~	
~~Resolution result~~		~~DA DCN address, TCP index~~



~~**Figure 8-12 – Discovery agent component**~~

Link discovery process

The generic process of discovery is split into two separate and distinct times and name spaces. The first part takes place entirely in the FP name space (CPs and CTPs), as described in Figure 8-~~13~~11.

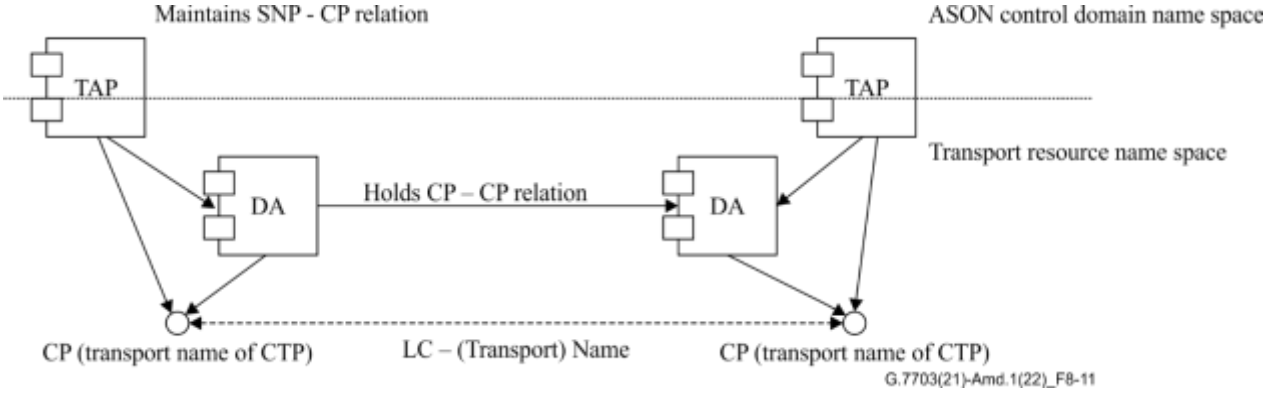


Figure 8-~~13~~ 11 – Discovery of transport link connections (LC)

The DA operates entirely within the transport resource name space and is responsible for holding the transport name of the link connection (associated with each CP). This information may be obtained by using transport mechanisms invisible in the control name space, by holding previously obtained relation information or by provisioning. The DA assists in an underlying automatic discovery process by cooperatively resolving transport CP names among all the DAs in the network, thus enabling the DAs (or other components) responsible for each end of the transport link connection to communicate about that link connection.

A CP can be assigned to a set of VPNs, including the empty set and the singleton set. This set of VPNs can be represented by an ownership tag. The DA verifies that ownership tag attached to each CP of a link connection is the same.

The second part takes place entirely within the control name space (SNPs), as depicted in Figure 8-~~14~~12.

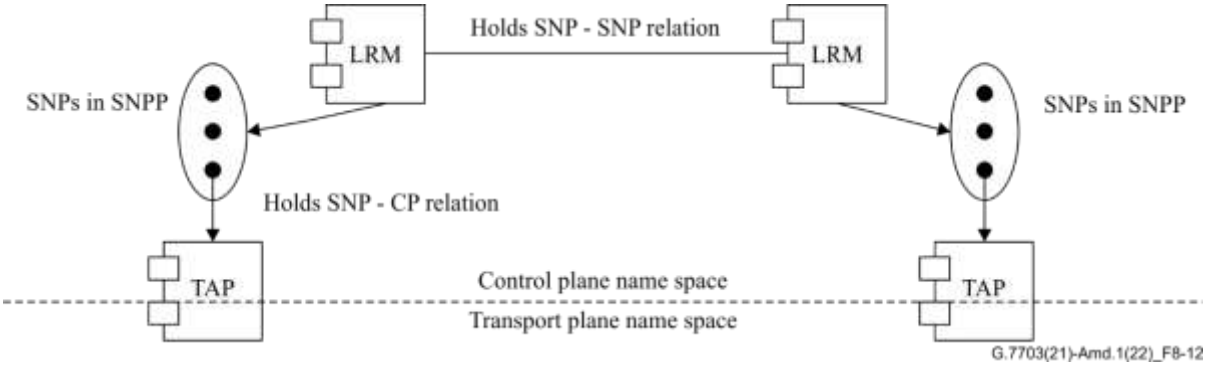


Figure 8-~~14~~12 – Population of link connections in the control name space

The link resource manager (LRM) holds the SNP-SNP binding information necessary for the control name of the link connection, while the TAP holds the relation between the control name (SNP) and the FP name of a resource. This separation allows control names to be completely separate from transport network plane names, and completely independent of the method used to populate the DAs with those transport names.

In order to assign an SNP-SNP link connection to an SNPP link, it is only necessary for the transport name for the link connection to exist. Thus, it is possible to assign link connections to the control domain without the link connection being physically connected. This assignment procedure may be verified by the LRMs exchanging the transport link connection name (i.e., CP-CP name or TCP-TCP name) that corresponds to the SNP.

The fully qualified SNPP link name is a control name reflecting the structure of transport network resources.

8.3.6 Termination and adaptation performers (TAP)

The termination and adaptation performers (TAP) component is described in clause 8.3.6 of [ITU-T G.7701].

8.3.7 Directory service (DS)

The directory service (DS) component is described in clause 8.3.7 of [ITU-T G.7701].

BRIs may be associated with UNI and ENNI reference points.

~~The directory service component is responsible for identifier resolution and coordination among peer directory service components. The role of this component is to provide mappings between identifier spaces for other components~~Two additions to the DS component description in clause 8.3.7 of [ITU-T G.7701] are the "peer coordination in" and "peer coordination out" interfaces. These are shown in Table 5, which extends Table 8-13 of [ITU-T G.7701]. The Boundary Resource identifiers (BRIs) in this Recommendation are associated with UNI and E-NNI references points (see clauses 3.2.16 and 3.2.17), but are not limited to just those reference points.

NOTE – All interfaces of Table 5 below are not intended to be used in one instance of this component. Only the directory request interface might be required for basic usage, but distributed implementations might use more interfaces.

Directory service functions can be implemented in both distributed and centralized applications. In a centralized application, peer coordination interfaces of the DS component might be unused. The in and out parameters of DS components can be found in Figure 8-~~15~~13.

Table 5 – Directory service component interfaces

Input interface	Basic input parameters	Basic return parameters
~~Directory request in~~	~~1) UNI/E-NNI BRI; or~~ ~~2) UNI/E-NNI BRI alias; or~~ ~~3) SNPP identifier; or~~ ~~4) SNPP alias;~~	~~1) SNPP identifier; or~~ ~~2) UNI/E-NNI BRI; or~~ ~~3) UNI/E-NNI BRI; or~~ ~~4) SNPP identifier.~~
Peer coordination in	1) <~~UNI/E-NNI~~ BRI, SNPP identifier> 2) <~~UNI/E-NNI~~ BRI alias, ~~UNI/E-NNI~~ BRI> 3) <SNPP identifier, ~~UNI/E-NNI~~ BRI> 4) <SNPP alias, SNPP identifier> 5) <SNPP identifier, SNPP alias>	
~~Directory information in~~	~~1) <UNI/E-NNI BRI, SNPP identifier>~~ ~~2) <UNI/E-NNI BRI alias, UNI/E-NNI BRI>~~ ~~3) <SNPP identifier, UNI/E-NNI BRI>~~ ~~4) <SNPP alias, SNPP identifier >~~ ~~5) <SNPP identifier, SNPP alias>~~ ~~6) list of BRIs~~	

Table 5 – Directory service component interfaces

Output interface	Basic output parameters	Basic return parameters
~~Directory request out~~	~~1) UNI/E-NNI BRI; or~~ ~~2) UNI/E-NNI BRI alias; or~~ ~~3) SNPP identifier; or~~ ~~4) SNPP alias;~~	~~1) SNPP identifier; or~~ ~~2) UNI/E-NNI BRI; or~~ ~~3) UNI/E-NNI BRI; or~~ ~~4) SNPP identifier.~~
Peer coordination out	1) <~~UNI/E-NNI~~ BRI, SNPP identifier> 2) <~~UNI/E-NNI~~ BRI alias, ~~UNI/E-NNI~~ BRI> 3) <SNPP identifier, ~~UNI/E-NNI~~ BRI> 4) <SNPP alias, SNPP identifier> 5) <SNPP identifier, SNPP alias>	
~~Directory information out~~	~~1) <UNI/E-NNI BRI, SNPP identifier>~~ ~~2) <UNI/E-NNI BRI alias, UNI/E-NNI BRI>~~ ~~3) <SNPP identifier, UNI/E-NNI BRI>~~ ~~4) <SNPP alias, SNPP identifier>~~ ~~5) <SNPP identifier, SNPP alias>~~ ~~6) list of BRIs~~	

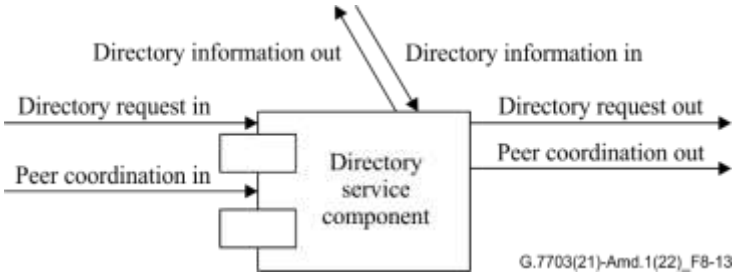


Figure 8-~~15~~ 13 – Directory service component

Directory request in/out

This interface is used to get an SNPP identifier from a ~~UNI/E-NNI~~ BRI or alias. And this interface is also used to get ~~UNI/E-NNI~~ BRI from a UNI BRI alias or SNPP identifier. Directory request should be bidirectional. CC could initialize a directory request and send to/receive from a DS component when it needs to decode DS.

Peer coordination in

This interface is used to get directory information from a peer directory service component.

Peer coordination out

This interface is used to transmit directory information to a peer directory service component.

Directory information in/out

This interface is used to receive/send directory information from other control components which could include other MC applications, and equipment management functions (EMFs) on subnetworks (e.g., NEs). The list of ~~TRIs~~ BRIs may be used by the DS component to create mappings to SNPPs and return in response to requests.

8.3.8 Notification Components

The notification component is described in clause 8.3.8 of [ITU-T G.7701].

8.3.9 Protocol controller (PC) components

The protocol controller (PC) component is described in clause 8.3.9 of [ITU-T G.7701].

8.3.10 Traffic policing (TP)

The traffic policing (TP) component is described in clause 8.3.10 of [ITU-T G.7701].

9 Common control communications

The common control communication can be found in clause 9 of [ITU-T G.7701].

9.1 Control communications network

With the multiple layers capabilities described in clauses 7.5 to 7.8, the existence of multiple signalling control networks (SCNs) (DCNs) creates an additional control component communication scenario.

It is possible that multiple disjoint SCNs exist between two connections or routing controllers that are adjacent in their layer. This is illustrated in Figure 9-1. This might happen if two service providers are using a third party in the middle of the network to provide the server layer service. In Figure 9-1, A_{client} and D_{client} are on separate SCNs and do not have any direct connectivity. Messages have to go through an intermediate SCN in order to reach D_{client} from A_{client}. The SCNs may form an even more complex topology, each SCN containing a set of policies regarding which messages are allowed to be carried over its SCN.

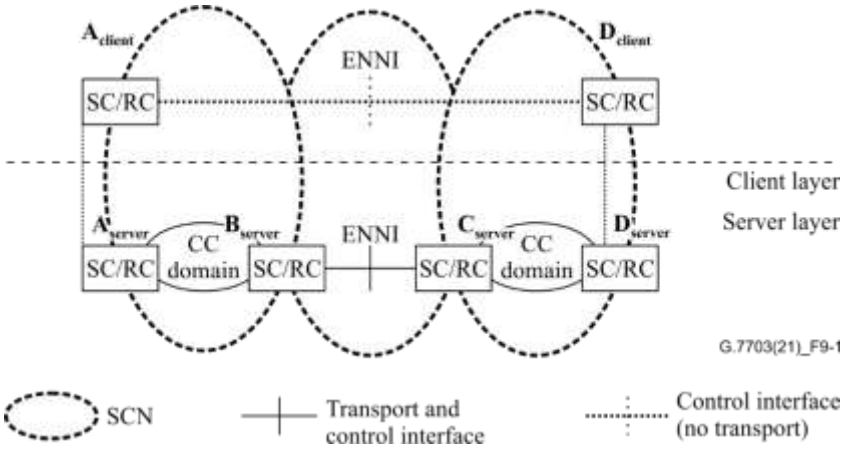


Figure 9-1 – Example of disjoint SCNs in the client layer

10 Common management aspects of common control components

The common management aspects of common control components can be found in clause 10 of [ITU-T G.7701].

11 Identifiers

[ITU-T G.7701] defines the following distinct and independent sets of name spaces exist, from which identifiers are drawn, for:

– resources in the transport network;

– control view of transport resources;

– control components;
– control artefacts;
– reference points;
– control communications network.

11.1 Resources in the transport network

The resources used in the transport network are described in clause 11.1 of [ITU-T G.7701].

11.2 Control view of transport resources

The control view of transport resources is described in clause 11.2 of [ITU-T G.7701].

11.2.1 Name spaces for routing and connection control

The name spaces used for routing and connection control are described in clause 11.2.1 of [ITU-T G.7701].

11.2.2 Name space recursion

Name space recursion is described in clause 11.2.2 of [ITU-T G.7701].

11.3 Control Components

The name spaces used by control components are described in clause 11.3 of [ITU-T G.7701].

11.4 Control artefacts

The control artefacts are described in clause 11.4 of [ITU-T G.7701].

11.5 Reference points

A generalized description of reference points is provided in clause 11.5 of [ITU-T G.7701]. This Recommendation defines the specific reference points that are used in ASON where signalling/routing information is exchanged. Reference points may be supported by multiple interfaces. These reference points are the UNI, the I-NNI and the E-NNI. It is important to recognize that there will be multiple domains within the ASON and that the UNI and E-NNI are used for inter-domain control signalling. The following clauses describe the specific functionalities that need to be carried across the various reference points (UNI, I-NNI and E-NNI) and how they differ.

Policy may be applied at the interfaces that support a reference point. The policies applied are dependent on the reference point and functions supported. For example, at the UNI, I-NNI and E-NNI reference points, policy may be applied to call and connection control. In addition, for the I-NNI and E-NNI reference points, policy may be applied to routing.

A reference point represents a collection of services, provided via interfaces on one or more pairs of components. The component interface is independent of the reference point; hence the same interface may be involved with more than one reference point. From the viewpoint of the reference point the components supporting the interface are not visible; hence the interface specification can be treated independently of the component.

The information flows that carry services across the reference point are terminated (or sourced) by components, and multiple flows need not be terminated at the same physical location. These may traverse different sequences of reference points as illustrated in Figure 11-1.

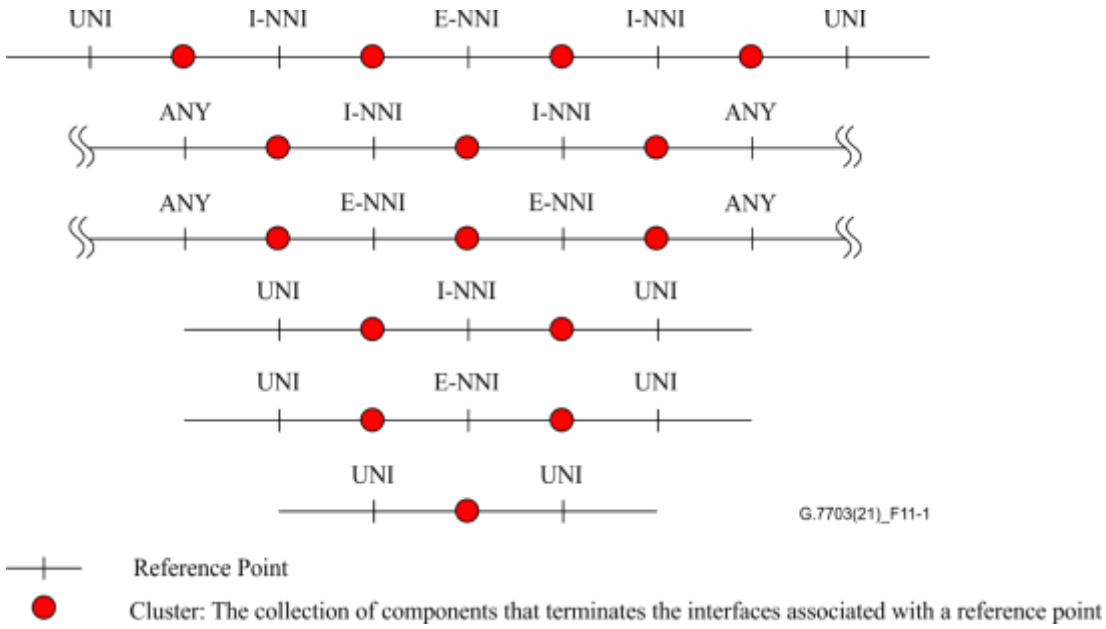


Figure 11-1 – Reference points

11.5.1 UNI

Information flows expected across the UNI reference point support the following functions:

– call control
– resource discovery
– connection control
– connection selection.

Note, there is no routing function associated with the UNI reference point.

Additional functions such as security and authentication of calls, or enhanced directory services, may be added to this basic set of functions.

11.5.2 I-NNI

Information flows expected across the I-NNI reference point support the following functions:

– resource discovery
– connection control
– connection selection
– connection routing.

11.5.3 E-NNI

Information flows expected across the E-NNI reference point support the following functions:

– call control
– resource discovery
– connection control
– connection selection
– connection routing.

Additional functions such as security and authentication of calls, or enhanced directory services may be added to this basic set of functions.

When the E-NNI reference point exists between a VPN customer domain and a VPN in a service provider domain, supplementary services may be supported (see [ITU-T Y.1312]). Examples are:

– VPN user authentication and authorization

– VPN user policy management, including connectivity restrictions

– transparent transfer of control information between VPN users

– VPN participation in the customer routing control domain.

Support for such services is outside the scope of this Recommendation.

12 Resilience

As described in clause 12 of [ITU-T G.7701], resilience refers to the ability of the MC components to continue operating under failure conditions. Operation of the MC components depends upon elements of the control communication network (CCN), the transport network and the internal components of the MCS itself.

12.1 Principles of MC component and transport network interaction

The interaction between the MC components and the transport network are described in clause 12.1 of [ITU-T G.7701].

12.2 Principles of protocol controller communication

Protocol controller communications are described in clause 12.2 of [ITU-T G.7701].

13 Connection availability enhancement techniques

This clause describes the strategies that can be used to maintain the integrity of an existing call in the event of failures within the transport network through the use of protection and/or restoration mechanisms. The description and application of these mechanisms is provided in clause 13 of [ITU-T G.7701].

13.1 Protection

The description of protection is provided in clause 13.1 of [ITU-T G.7701].

13.2 Restoration

The description of restoration is provided in clause 13.2 of [ITU-T G.7701].

The edge components of each ASON re-routing domain negotiate the activation of the re-routing services across the re-routing domain for each call. During the negotiation of the re-routing services, the edge components of a re-routing domain exchange their re-routing capabilities and the request for a re-routing service can only be supported if the service is available in both the source and destination at the edge of the re-routing domain.

This is illustrated in Figure 13-1.

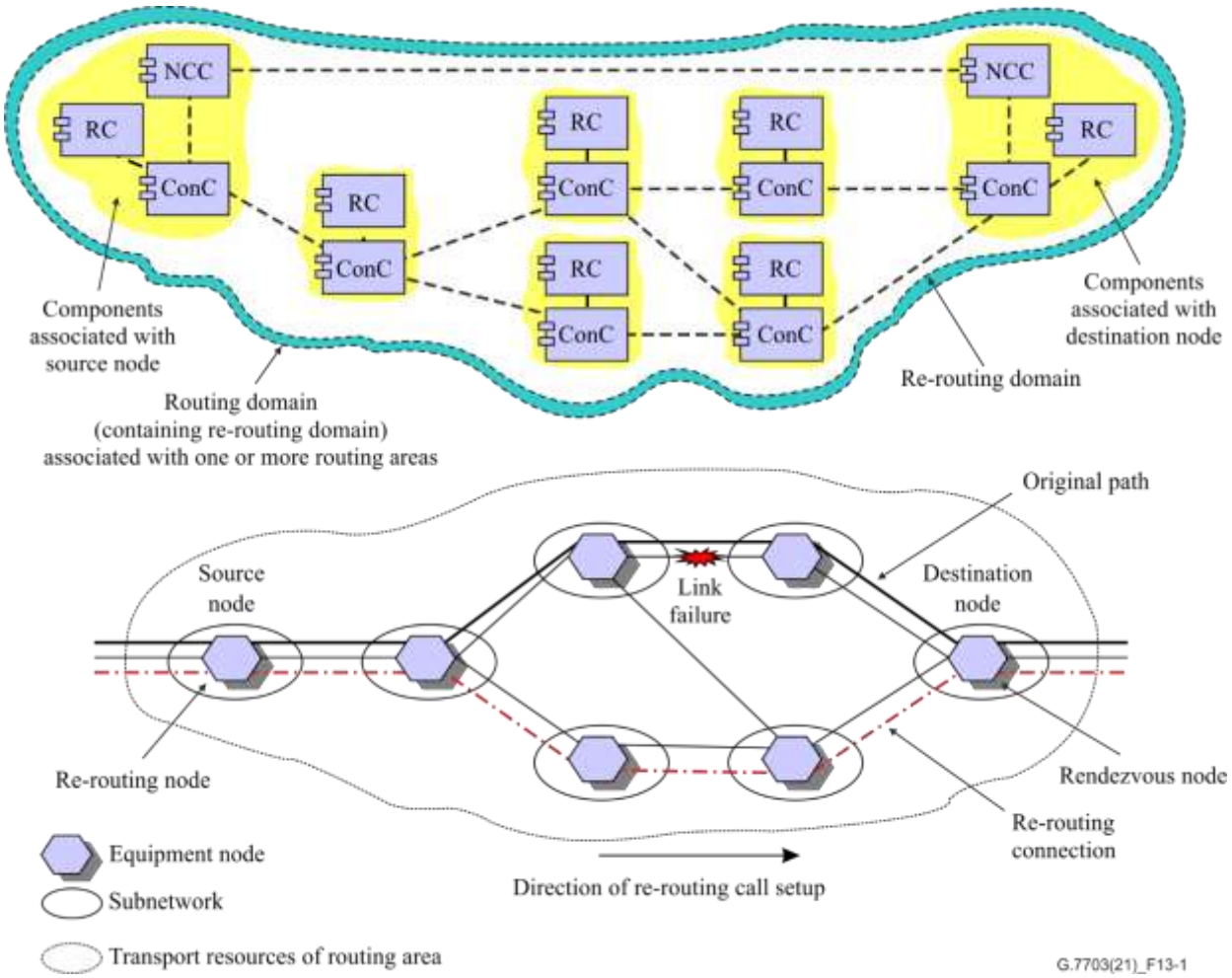


Figure 13-1 – Example of re-routing

13.2.1 Re-routing in response to failure

13.2.1.1 Intra-domain failures

The description of intra-domain failures is provided in clause 13.2.1.1 of [ITU-T G.7701].

13.2.1.2 Inter-domain failures

The description of inter-domain failures is provided in clause 13.2.1.2 of [ITU-T G.7701].

13.2.1.3 Link failure between adjacent gateway network elements

The description of recovery from a link failure between adjacent gateway network elements is provided in clause 13.2.1.2.1 of [ITU-T G.7701].

13.2.1.4 Gateway network element failure

The description of recovery from the failure a gateway network element is provided in clause 13.2.1.2.2 of [ITU-T G.7701].

13.3 Nested routing domains

The use of nested routing domains is described in clause 13.3 of [ITU-T G.7701].

14 Topology and discovery

14.1 SNPP links

The routing function understands topology in terms of SNPP links. Before SNPP links can be created, the underlying transport topology, i.e., the trail relationship between the access points, must be established. These relationships may be discovered (or confirmed against a network plan) using a number of different techniques; for example, use of a test signal or derived from a trail trace in the server layer. They may also be provided by a management system based on a network plan. The capability of the transport equipment to support flexible adaptation functions (and thus link connections for multiple client layer networks) may also be discovered or reported.

Link connections that are equivalent for routing purposes are then grouped into links. This grouping is based on parameters, such as link cost, delay, quality or diversity. Some of these parameters may be derived from the server layer but in general they will be provisioned by other MC systems.

Separate links may be created (i.e., link connections that are equivalent for routing purposes may be placed in different links) to allow the division of resources between different ASON networks (e.g., different VPNs) or between resources controlled by ASON and other MC systems.

The link information (e.g., the constituent link connections or resource label range with the available link bandwidth) is then used to configure the LRM instances (as described in clause 8.3.4) associated with the SNPP link. Additional characteristics of the link, based on parameters of the (potential) link connections, may also be provided. The LRMs at each end of the link must establish a control domain adjacency that corresponds to the SNPP link. The interface SNPP ids may be negotiated during adjacency discovery or may be provided as part of the LRM configuration. The link connections and CP names or resource labels (and link connections) are then mapped to interface SNP ids (and SNP link connection names). In the case where both ends of the link are within the same routing area the local and interface SNPP ~~id~~ ID and the local and interface SNP ids may be identical. Otherwise, at each end of the link the interface SNPP ~~id~~ ID is mapped to a local SNPP ~~id~~ ID and the interface SNP ids are mapped to local SNP ids. This is shown in Figure 14-1.

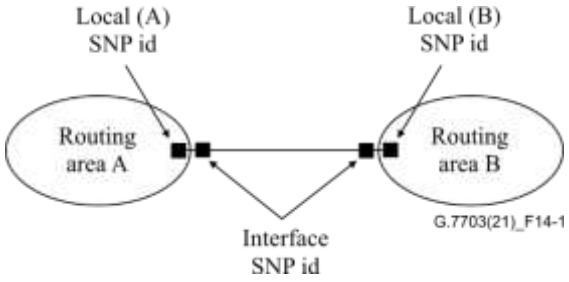


Figure 14-1 – Relationship between local and interface ids

The resulting SNP link connections may then be validated by a discovery process. The degree of validation required at this stage is dependent on the integrity of the link connection relationships initially provided by the transport network or other MC systems and the integrity of the process used to map CPs to SNPs.

Validation may be derived from a trail trace in the server layer or by using a test signal and test connections. If test connections are used, the discovery process may set up and release these connections using either another MC system or ASON. If ASON is used, the link must be made temporarily available to routing and connection control, for test connections only.

Once the SNPP link validation is completed, the LRMs inform the RC component (see clause 8.3.3) of the SNPP link adjacency and the link characteristics, e.g., cost, performance, quality diversity, and bandwidth.

14.2 Routing areas

Within the context of this Recommendation, a routing area exists within a single layer network. 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 one subnetwork.

Note that the SNPP links fully contained within the routing area may be transitional links, interconnecting child RAs operating on different CI.

Routing areas and subnetworks are very closely related as both provide an identical function in partitioning a network. The critical distinction is that at the boundary, the link ends are visible from inside a routing area, whereas inside a subnetwork only connection points can be seen. Seen from the outside, subnetworks and RAs are identical, and the terms subnetwork and RA can be used almost synonymously. The distinction between the two is usually obvious from the context, though the term node is often used to denote either a subnetwork or RA. Also note that from the outside of both subnetworks and routing areas, it is not possible to see any internal details, and both subnetworks and routing areas appear as points in the network topology graph.

Where an SNPP link crosses the boundary of a routing area, all the routing areas sharing that common boundary have contained coincident SNPP links. This is illustrated in Figure 14-2.

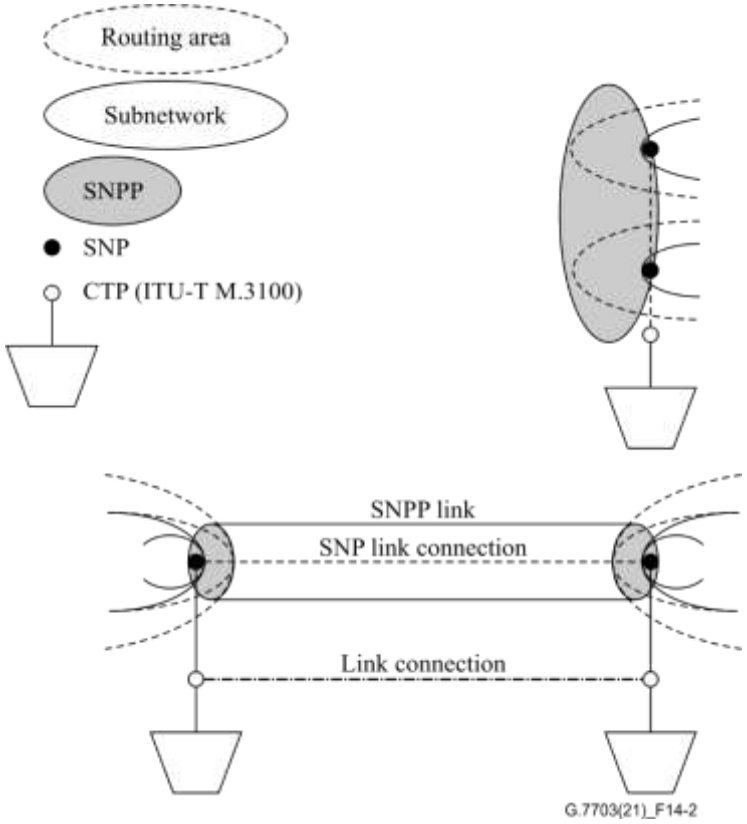


Figure 14-2 – Relationship between routing areas, subnetworks, SNPs and SNPP

14.2.1 Aggregation of links and routing areas

Figure 14-3 illustrates the relationships between routing areas, SNPPs, and SNPP links. Routing areas and SNPP links may be related hierarchically. In the example, routing area A is partitioned to create a lower level of routing areas, B, C, D, E, F, G and interconnecting SNPP links. This recursion can continue as many times as necessary. For example, routing area E is further partitioned to reveal routing areas H and I. In the example given, there is a single top level routing area. In creating a hierarchical routing area structure based upon "containment" (in which the lower level routing areas

are completely contained within a single higher level routing area), only a subset of lower level routing areas, and a subset of their SNPP links are on the boundary of the higher level routing area. The internal structure of the lower level is visible to the higher level when viewed from inside of A, but not from outside of A. Consequently, only the SNPP links at the boundary between a higher and lower level are visible to the higher level when viewed from outside of A. Hence the outermost SNPP links of B and C and F and G are visible from outside of A but not the internal SNPP links associated with D and E or those between B and D, C and D, C and E or between E and F or E and G. The same visibility applies between E and its subordinates H and I. This visibility of the boundary between levels is recursive. SNPP link hierarchies are therefore only created at the points where higher level routing areas are bounded by SNPP links in lower level routing areas.

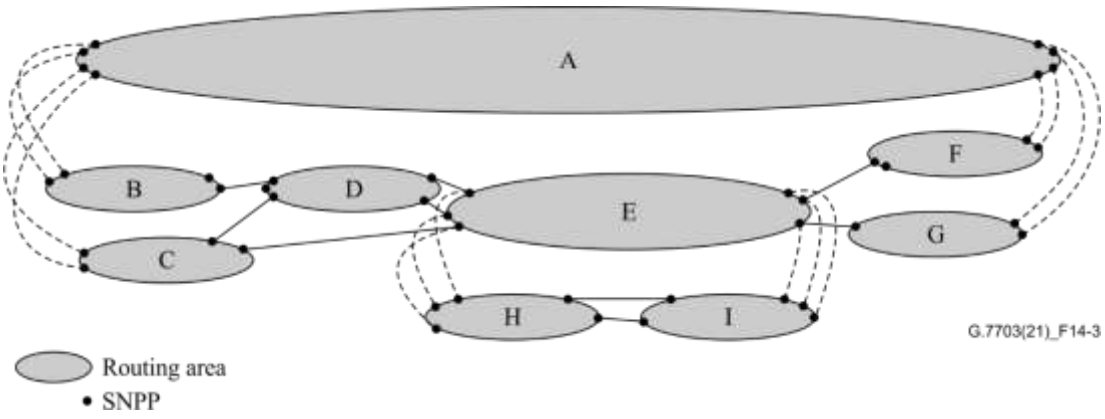


Figure 14-3 – Example of a routing area hierarchy and SNPP link relationships

Subnetwork points are allocated to an SNPP link at the lowest level of the routing hierarchy and can only be allocated to a single subnetwork point pool at that level. At the routing area hierarchy boundaries, the SNPP link pool at a lower level is fully contained by an SNPP link at a higher level. A higher level SNPP link pool may contain one or more lower level SNPP links. In any level of this hierarchy, an SNPP link is associated with only one routing area. As such routing areas do not overlap at any level of the hierarchy. SNPP links within a level of the routing area hierarchy that are not at the boundary of a higher level may be at the boundary with a lower level thereby creating an SNPP link hierarchy from that point (e.g., routing area E). This provides for the creation of a containment hierarchy for SNPP links.

A routing area may have an SNPP name space that is independent from those used in other routing areas. Note, an SNPP name is routable in the RA whose SNPP name space it belongs to.

14.2.2 Relationship to links and link aggregation

A number of SNP link connections within a routing area can be assigned to the same SNPP link if, and only if, they go between the same two subnetworks. This is illustrated in Figure 14-4. Four subnetworks, SNa, SNb, SNc and SNd and SNPP links 1, 2 and 3 are within a single routing area. SNP link connections A and B are in the SNPP link 1. SNP link connections B and C cannot be in the same SNPP link because they do not connect the same two subnetworks. Similar behaviour also applies to the grouping of SNPs between routing areas.

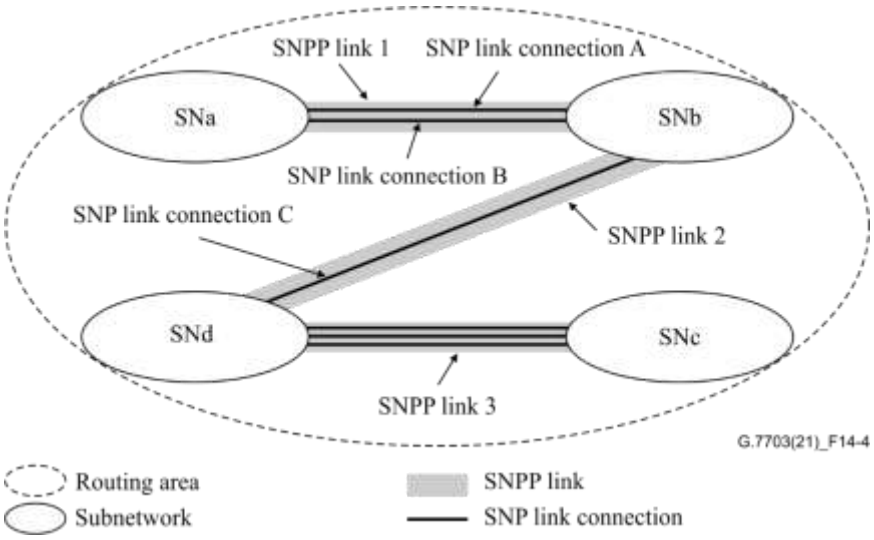


Figure 14-4 – SNPP link relationship to subnetworks

Figure 14-5 shows three routing areas, RA-1, RA-2 and RA-3 and SNPP links 1 and 2. SNP link connections A, B, and C cannot be in the same SNPP link because more than two routing areas are found in their endpoints. SNP link connections A & B are not equivalent to SNP link connection C for routing from Routing Area 3 (RA-3).

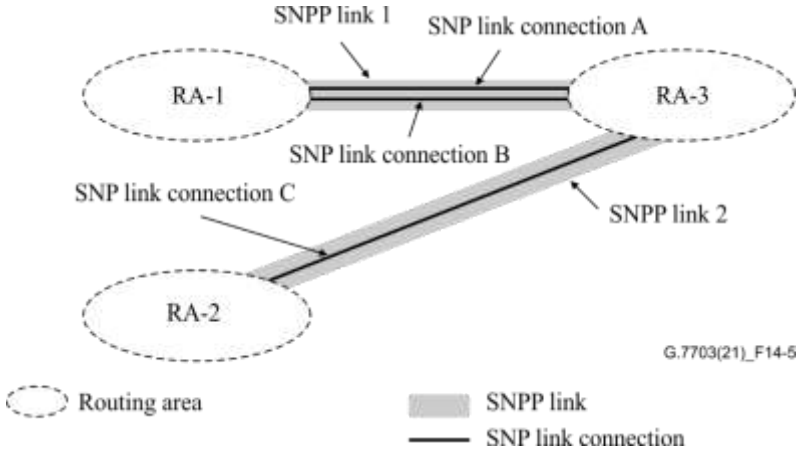


Figure 14-5 – SNPP link relationships to routing areas

SNP link connections between two routing areas, or subnetworks, can be grouped into one or more SNPP links. Grouping into multiple SNPP links may be required:

– if they are not equivalent for routing purposes with respect to the routing areas they are attached to, or to the containing routing area;

– if smaller groupings are required for administrative purposes.

There may be more than one routing scope to consider when organizing SNP link connections into SNPP links. In Figure 14-6, there are two SNP link connections between routing areas 1 and 3. If those two routing areas are at the top of the routing hierarchy (there is therefore no single top level routing area), then the routing scope of RA-1 and RA-3 is used to determine if the SNP link connections are equivalent for the purpose of routing.

The situation may however be as shown in Figure 14-6. Here RA-0 is a containing routing area. From RA-0's point of view, SNP link connections A & B could be in one (case a) or two (case b) SNPP links. An example of when one SNPP link suffices is if the routing paradigm for RA-0 is step-by-step.

Path computation sees no distinction between SNP link connections A and B as a next step to get from say RA-1 to RA-3.

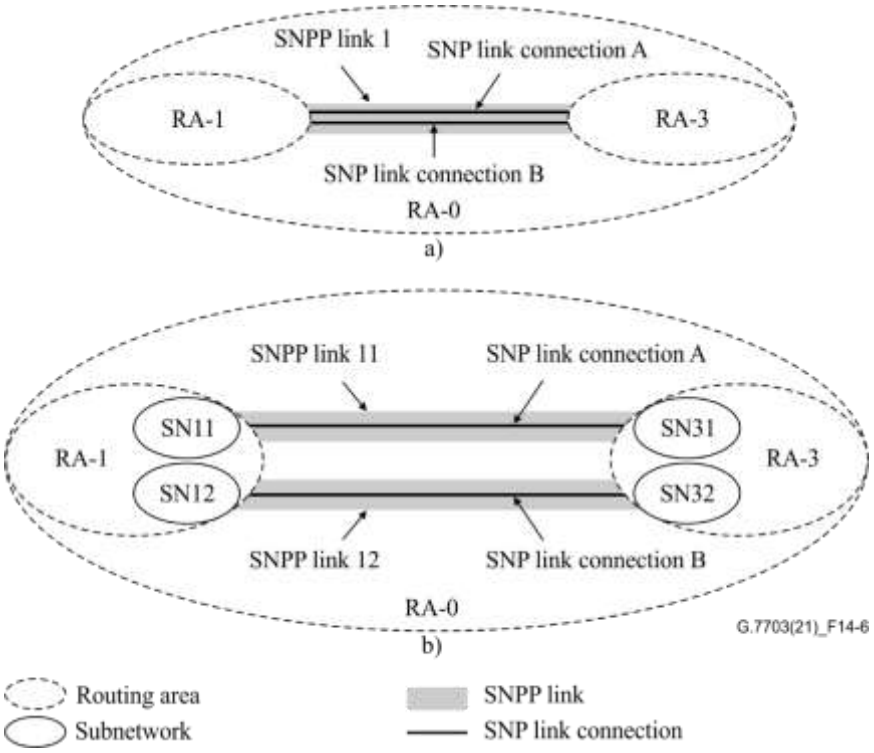


Figure 14-6 – Routing scope

From RA-1 and RA-3's point of view though, the SNP link connections may be quite distinct from a routing point of view as choosing SNP link connection A may be more desirable than SNP link connection B for cost, protection or another reason. In this case, placing each SNP link connection into its own SNPP link meets the requirement of "equivalent for the purpose of routing". Note that in Figure 14-6, SNPP link 11, link 12 and link 1 can all coexist.

Another reason for choosing SNPP link 11 (Figure 14-6 b) over SNPP link 12 could be because the cost of crossing RA-3 is different from SNPP link 11 than from SNPP link 12. This suggests that a mechanism to determine the relative cost of crossing RA-3 from link 11 and from link 12 would be useful. Such a mechanism could be used recursively to determine the relative cost of crossing RA-0. Note that this does not imply exposing the internal topology of any routing area outside of its scope. A query function could be invoked to return the cost of a particular route choice. The costs returned by such a query would be determined by policy applied to each routing area. A common policy should be used in all the routing areas, resulting in comparable costs. Such a query could also be generalized to apply routing constraints before calculating the cost.

Routing areas in different layers may be connected by transitional SNPP links. This enables multi-layer routing topology construction. Routing areas in different sublayers of the same layer may also be connected by transitional SNPP links.

Annex A

Connection services

(This annex forms an integral part of this Recommendation.)

The control of connectivity is essential to the operation of a transport network. The transport network itself can be described as a set of layer networks, each acting as a connecting function whereby associations are created and removed between the inputs and outputs of the function. These associations are referred to as connections. Three types of connection establishment are defined:

1) Permanent connection: This type of connection is established within an ASON control domain by an external MC system. This type of connection is referred to as a permanent connection. The MC components within the ASON control domain cannot create, delete or modify a permanent connection. See Figure A.1.

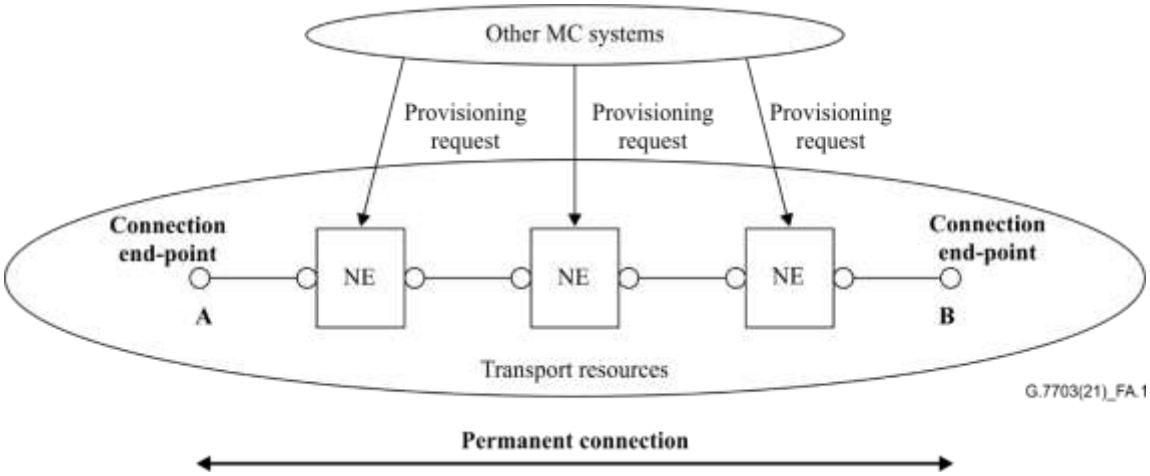


Figure A.1 – Example of end-to-end transport connection set-up using provisioning via another MC system

2) **Switched connection (SC)**: This type of connection is established on demand by the communicating end-points within an ASON control domain using a dynamic protocol message exchange in the form of signalling messages. These messages flow across either the UNI, I-NNI or E-NNI within the control domain. This type of connection is referred to as a switched connection. Such connections require network naming and addressing schemes and control protocols. See Figure A.2.

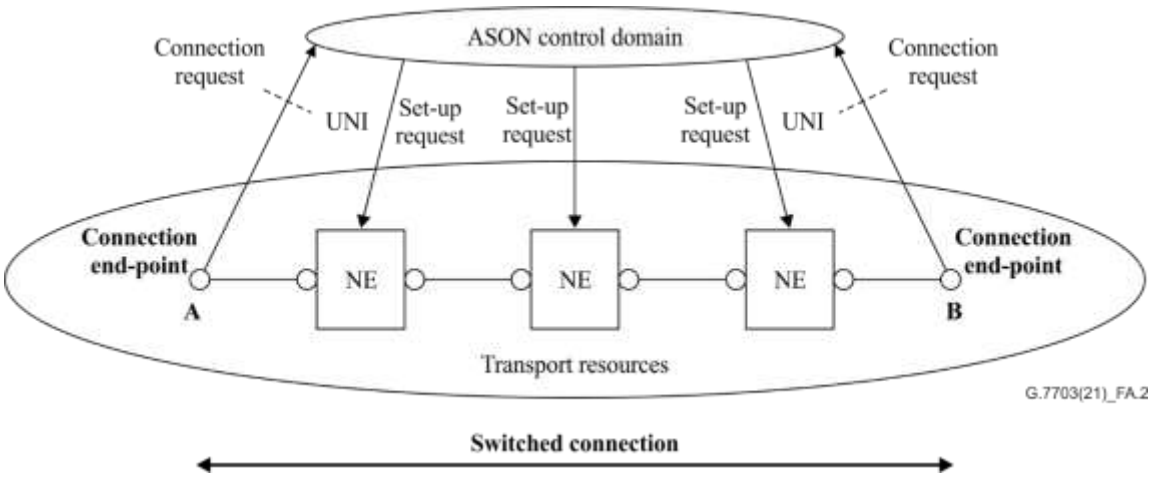


Figure A.2 – Example of end-to-end transport connection set-up using control signalling (a switched connection from A to B)

3) **Soft permanent connection (SPC)**: This type of connection consists of the concatenation of a permanent connections at the edge of the network with a switched connection within the ASON control domain to provide an end-to-end connection. This type of network connection is known as a soft permanent connection (SPC). The UNI reference point is not used with this type of connection. From the perspective of the end points a soft permanent connection has the same characteristics as a permanent connection. See Figure A.3.

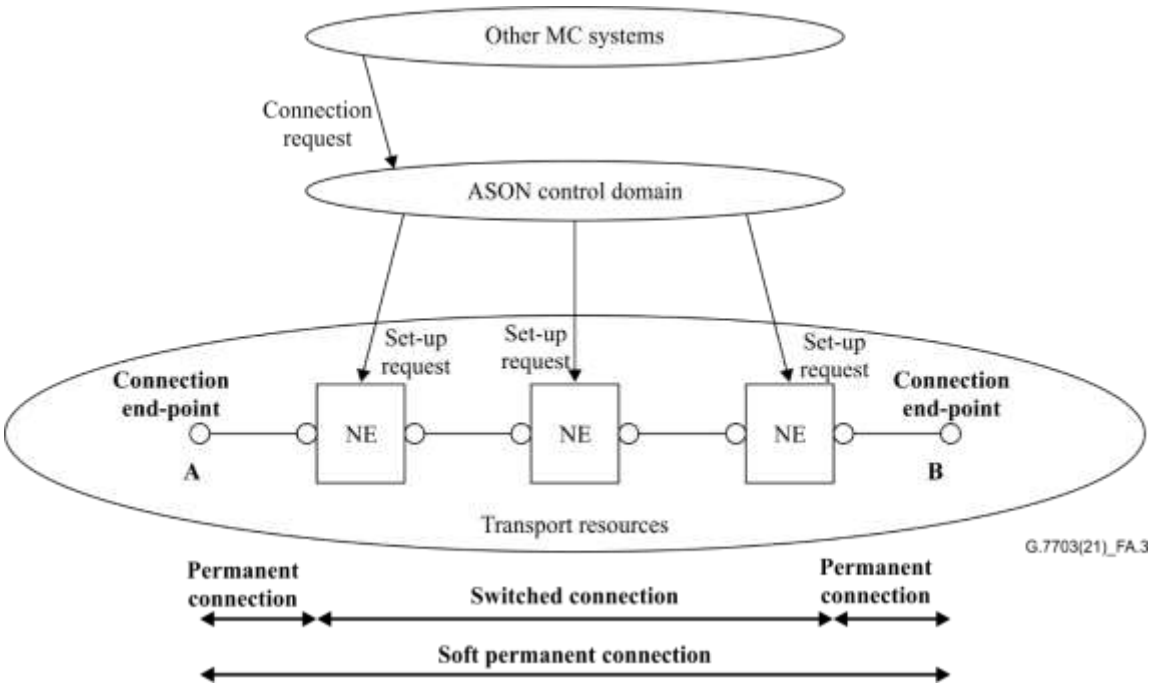


Figure A.3 – Example of end-to-end transport connection set-up as soft permanent connection (SPC)

The most significant difference between the connection types described above is the party that sets up the connection. In the case of permanent connection or SPC set-up is the responsibility of the network operator, whilst in the case of the SC, connection set-up controlled by the end user using the UNI reference point. Additionally, third party signalling should be supported across a UNI.

NOTE 1 – The type of connection may have impact on future billing systems.

ASON shall support either a switched connection (SC) or soft permanent connection (SPC) of the basic connection capability in the transport network. These connection capability types are defined below:

- unidirectional point-to-point connection
- bidirectional point-to-point connection
- unidirectional point-to-multipoint connection.

NOTE 2 – A further connection type can be considered, namely an asymmetric connection. This may be constructed either as two unidirectional point-to-point connections, having different properties in each direction, or as a special case of bidirectional connection.

The function of a UNI is to pass signalling messages directly to the ASON control domain. Alternatively, where a network operator already has extensive management systems in place that provide planning assignment and auto-configuration, these signalling messages may be passed directly to service management and network management system agents to effect connection set-up. Such an application will allow near real time automated service provision from the existing management platforms.

Appendix I

Resilience relationships

(This appendix does not form an integral part of this Recommendation.)

Resilience refers to the ability of ASON to continue operating under failure conditions. Operation of ASON depends upon elements of the data communication network (DCN), the transport resources, other MC systems and the internal components of the ASON control domain itself (refer to Figure 6-1). The following clauses identify the dependencies on those areas. The desired degree of control domain resiliency can then be engineered by providing appropriate redundancy for the dependent functions.

I.1 ASON control domain – DCN relationships

An ASON control domain relies on the DCN for the transfer of signalling messages over some or all of the following interfaces (refer to Figure I.1): UNI, NNI, NMI. The impact of a signalling channel failure on the operation of the ASON control domain will be examined for each of the protocol controllers associated with each interface.

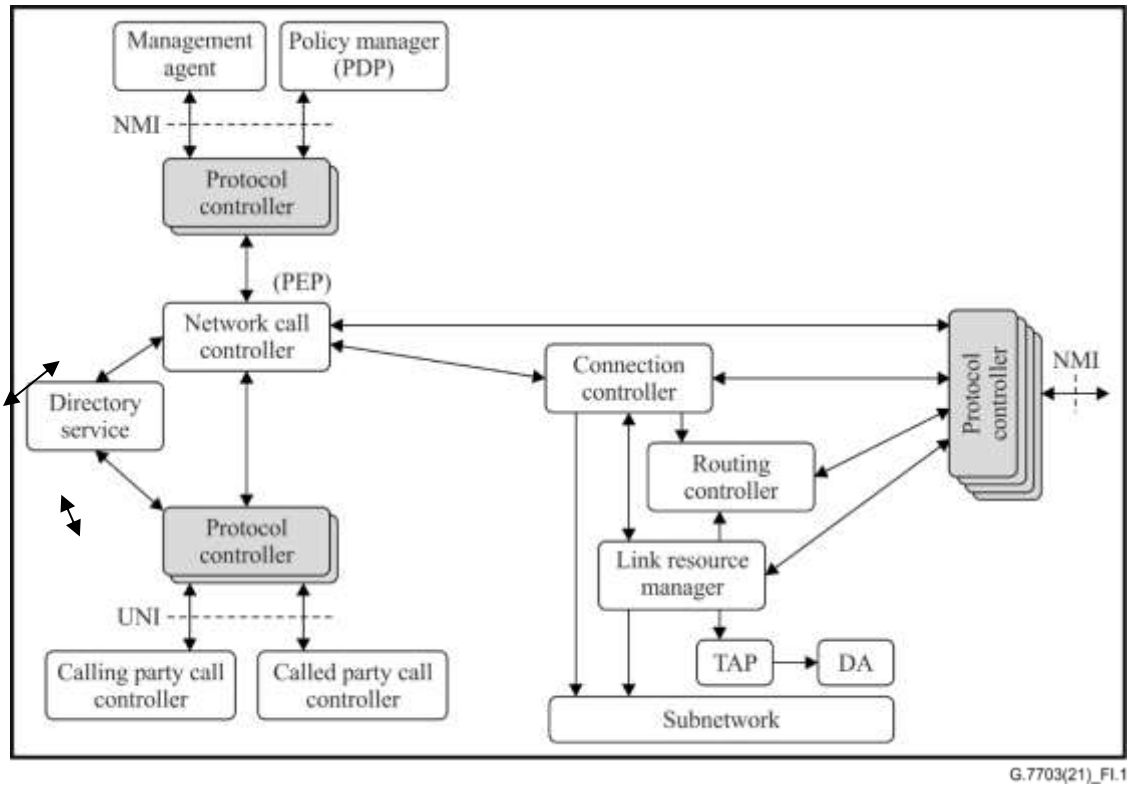


Figure I.1 – ASON control domain components (an interpretation)

I.1.1 UNI

There are potentially two separate protocol controllers handling the signalling sessions over the UNI: one for the calling party call controller link and one for the called party call controller link.

I.1.1.1 Failure case

A failure of the signalling session supporting the UNI for the calling party call controller link will result in the loss of the call request/call release control flows.

A failure of the signalling session supporting the UNI for the called party call controller link will result in the loss of the call request/call indication control flows.

A failure of either of the UNI-related signalling session impacts the network call controller function.

In all cases above, existing calls and their connections are not altered. Other MC systems may be notified if the failure persists and requires operator intervention (for example, to release a call).

I.1.1.2 Recovery case

When the signalling channel recovers, state re-synchronization between the client call controllers and the network call controller, and the connection controllers over the UNI, should be performed.

I.1.2 NNI

There are potentially four separate protocol controllers handling the signalling sessions over the NNI: one for the network call controller link, one for the connection controller link, one for the routing controller link and one for the link resource manager link.

I.1.2.1 Failure case

A failure of the signalling session supporting the NNI for the network call controller link will result in the loss of the network call controller coordination control flows. Call set-up or release will not be possible, but there is no impact on connection set-up or release.

A failure of the signalling session supporting the NNI for the connection controller link will result in the loss of the connection controller coordination and connection request/call release control flows. Connection set-up or release will not be possible. Further, if call control is piggybacked on connection control, no call set-up or release will be possible either.

A failure of the signalling session supporting the NNI for the routing controller link will result in the loss of the network/local topology control flows.

A failure of the signalling session supporting the NNI for the link resource manager link will result in the loss of the SNP negotiation/release control flows.

A failure of the link resource manager signalling session impacts the routing controller function and the connection controller function. A failure of the routing controller signalling session impacts the connection controller function. A failure of the connection controller signalling session impacts the network call controller function.

In all cases above, existing calls and their connections are not altered. Other MC systems may be notified if the failure persists and requires operator intervention (for example, to release a call).

Note that a failure of the DCN may affect one or more or all of the above signalling sessions simultaneously. The protocol controller associated with each signalling channel must detect and alarm a signalling channel failure.

I.1.2.2 Recovery case

Upon restoral of a previously failed signalling channel, the corresponding protocol controller must ensure all messaging resumes in sequence. Components are responsible for re-establishing state information after protocol controller recovery.

I.2 ASON control domain – Transport resource relationships

This clause considers only those transport resource failures that affect the ability of the control domain to perform its functions, for example when an LRM cannot be informed. Transport resource failures, such as port failures, are not within the scope of this Recommendation as it is expected that the control domain is informed of this situation. Information consistency between the control domain and the transport network is described in clause 12.1 of [ITU-T G.7701].

I.2.1 Transport resource information – Query

The control components will query the transport resources under the following scenarios:

– when a connection controller signalling session activates, or re-activates (for example, following the recovery of a data link or transport NE);

– control component queries about the transport resources;

– as part of transport resource information synchronization (for example, when the control domain recovers following a failure).

I.2.2 Transport resource information – Event driven

The transport resources will inform the control components on an event basis under the following scenarios:

– failure of a transport resource

– addition/removal of a transport resource.

I.2.2.1 Transport network protection

Transport network protection actions, which are successful, are largely transparent to the control domain. The transport network is only required to notify the control domain of changes in the availability of transport resources.

Transport network protection attempts, which are unsuccessful, appear to the control domain as connection failures, and may trigger control domain restoration actions, if such functionality is provided. Given that the control domain supports restoration functionality, the following relationships exist.

The routing controller must be informed of the failure of a transport network link or node and update the network/local topology database accordingly. The routing controller may inform the local connection controller of the faults.

I.2.3 Transport network dependency on the ASON control domain

If the control domain fails, new connection requests that require the use of the failed control components cannot be processed. Note, however that other MC systems could be used as a fallback to respond to new connection requests. Established connections must not be affected by a control domain failure.

I.3 Control domain – MC system relationships

The control domain may obtain directory and policy information from other MC systems during the call admission control validation process. Failure of the directory or policy servers could result in the failure of connection set-up requests.

Examples of this are:

– At the network call controller (at the calling or called party end), call requests may need to be validated by policy checking.

– When connection controllers request a path from the routing controller, a policy server may need to be consulted.

Call release actions can take place in the control domain if the other MC system is not available. A record of these actions must be maintained by the control domain so that when the MC system becomes available, a log can be sent or the control domain can be queried for this information.

I.3.1 NMI

All control components have monitor, policy and configuration ports which provide the management view of the control components (see clause 7.2.1).

There are potentially two separate protocol controllers/signalling sessions involving management information flows: one for the policy manager session and one for a transport management session. Other protocol controllers may be introduced in the future for other management functions.

I.3.1.1 Failure case

A failure of the signalling session supporting the policy manager link will result in the loss of the policy out control flows.

A failure of the transport management signalling session will result in the loss of FCAPS (fault, configuration, accounting, performance, security) information exchange.

A failure of the policy session impacts the network call controller function. For example, the potential failure of new connection set-up requests during the call admission control validation process requires policy manager access.

I.3.1.2 Recovery case

When management communication is recovered, information stored in the control domain that should have been sent (e.g., call records) is delivered to the MC system. Information pending from the MC system to the control domain should be sent (e.g., revised policy or configuration).

I.4 Intra-control domain relationships

The impact of control component failures on the operation of the control domain overall will be examined per the component relationship illustrated in Figure I.1. To achieve continuous operation of the control domain under a component failure, the ability to detect a component failure and switch to a redundant component, without loss of messages and state information, is required.

If control components are not redundant, then when a failed component recovers, it must re-establish a sufficient view of the transport network resources in order to be operational.

It is assumed that the communications between components other than protocol controllers (i.e., non-PC communications) is highly reliable. Such communications is likely internal to a control node and is implementation specific, thus it is outside the scope of this Recommendation.

I.4.1 Network call controller

The failure of a network call controller will result in the loss of new call set-up requests and existing call release requests.

I.4.2 Connection controller

The failure of a connection controller will result in the loss of new connection set-up requests and existing connection release requests. As call control signalling is often implemented via the connection controller and its protocol controller, a failure of the connection controller may impact the network call controller function (e.g., may not be able to release existing calls).

I.4.3 Routing controller

The failure of a routing controller will result in the loss of new connection set-up requests and loss of topology database synchronization. As the connection controller depends on the routing controller for path selection, a failure of the routing controller impacts the connection controller. MC system queries for routing information will also be impacted by a routing controller failure.

I.4.4 Link resource manager

The failure of a link resource manager will result in the loss of new connection set-up requests and existing connection release requests, and loss of SNP database synchronization. As the routing controller depends on the link resource manager for transport resource information, the routing controller function is impacted by a link resource manager failure.

I.4.5 Protocol controllers

The failure of any of the protocol controllers has the same effect as the failure of the corresponding DCN signalling sessions as identified above. The failure of an entire control node must be detected by the neighbouring nodes NNI protocol controllers.

I.4.6 Intra-control domain information consistency

As discussed in clause 12.1 of [ITU-T G.7701], at a given node, control component resource and subnetwork connection (SNC) state information consistency with the local transport NE resource and state information must be established first. Then control components must ensure SNC state information consistency with its adjacent control components. Any connection differences must be resolved such that no connection fragments remain, or misconnections occur. Following the control component information consistency cross-check, the control components are permitted to participate in control domain connection set-up or release requests.

Appendix II

Example of layered call control

(This appendix does not form an integral part of this Recommendation.)

Figure II.1 illustrates the mapped server case with the inter-layer call model for two Ethernet clients. They attach to a common VC-3 network that does not support Ethernet switching. Suppose that a 40 Mbit/s call is requested over a Gigabit Ethernet UNI. To carry Ethernet CI, a VC-3 connection is created. The decision by the NCC_{MAC} to make a call to the corresponding NCC_{VC-3} is driven by operator policy. Both layers are shown with only the VC-3 layer having a network connection. Once the VC-3 connection is established, the ETH FPP link connection between the two NCC_{MAC} comes into existence.

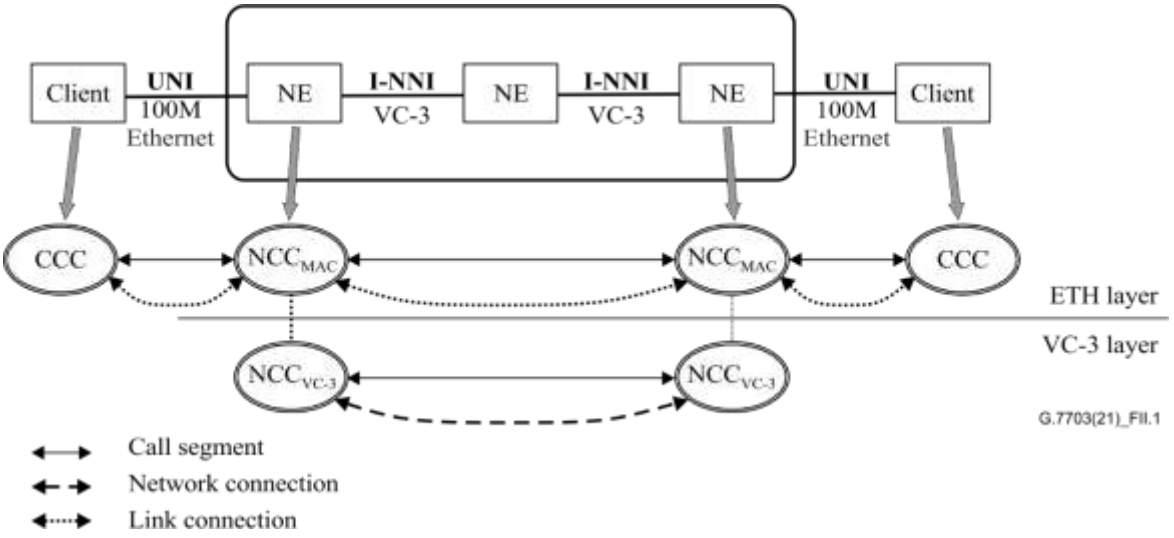


Figure II.1 – Ethernet over VC-3 example

In the sequence of events, the establishment of calls at different server layers may be independent in time. For example, the incoming Ethernet call could trigger the VC-3. Alternately the VC-3 connection may already exist and then be associated to an incoming MAC call. The association of the VC-3 connection to the requested Ethernet call is also driven by operator policy.

There are numerous other examples of interlayer calls such as fibre channel over SDH/OTN.

Appendix III

Component interactions for connection set-up

(This appendix does not form an integral part of this Recommendation.)

Clause 8.1 of [ITU-T G.7701] states that controller components are abstract entities that may be implemented as a single entity or as a distributed set of entities making up a cooperative federation. However, for clarity of illustration, the examples in this appendix show potential implementation approaches in which the components shown are not abstract entities but rather specific instances of implementation code. Specifically:

– network call controllers are shown as a distributed cooperative federation

– routing controllers are shown in a distributed cooperative federation

– connection controllers are shown as a single entity for a matrix

– LRM are shown as a single entity handling all link ends for a matrix.

In some examples, a shaded box is used to show the boundaries of the distributed cooperative federation that make up an abstract entity.

In order to control a connection, it is necessary for a number of components to interact.

Three basic forms of algorithm for dynamic path control can be distinguished: hierarchical, source routing and step-by-step routing as shown in the following figures. The different forms of path control result in a different distribution of components between nodes and relationships between these connection controllers. In case an RC does not have sufficient routing information to provide a route for a connection request, it may communicate with other RCs to resolve the route using the route query interface as described in clause 8.3.3 of [ITU-T G.7701].

III.1 Hierarchical routing

In the case of hierarchical routing, as illustrated in Figure III.1, a node contains a routing controller, connection controllers and link resource managers for a single level in a routing area hierarchy. The decomposition of routing areas follows the decomposition of a layer network into a hierarchy of subnetworks (in line with the concepts described in [ITU-T G.805]. Connection controllers are related to one another in a hierarchical manner. Each routing area has its own dynamic connection control that has knowledge of the topology of its routing area but has no knowledge of the topology of routing areas above or below itself in the hierarchy, or other routing areas at the same level in the hierarchy.

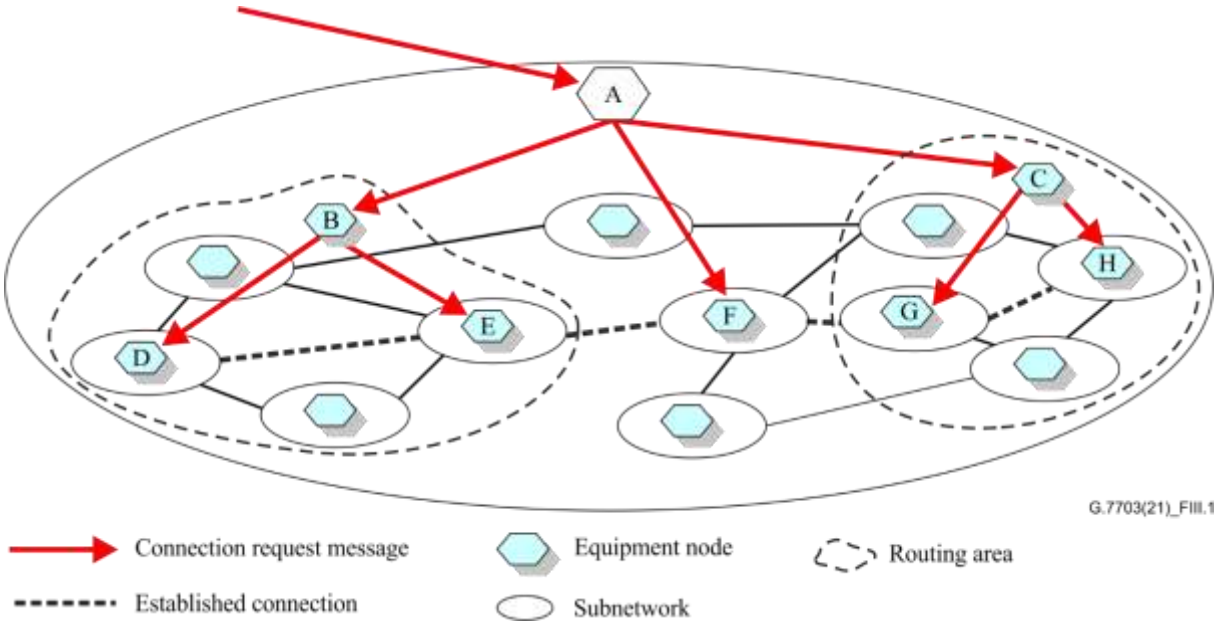


Figure III.1 – Hierarchical signalling flow

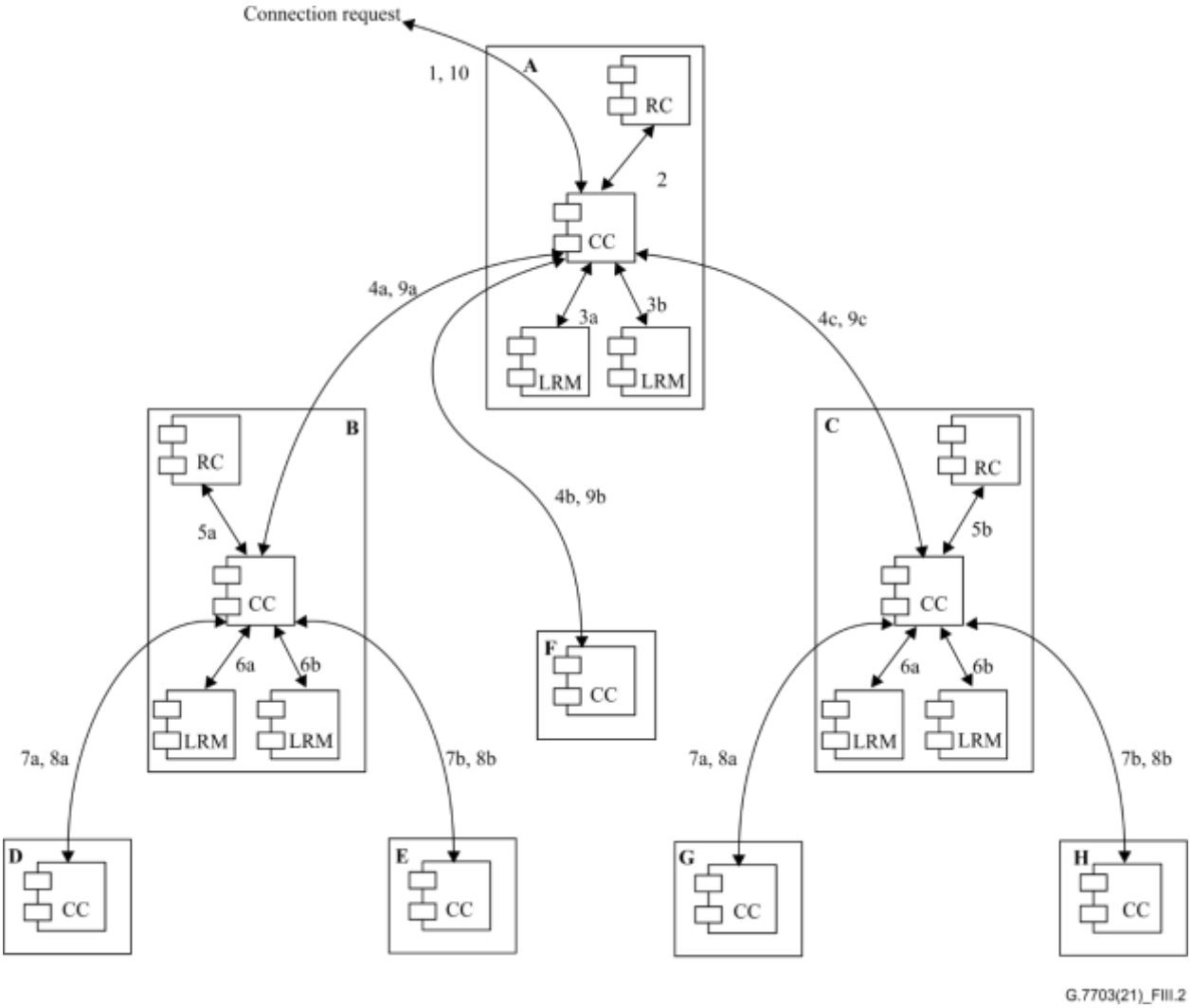


Figure III.2 – Hierarchical routing interactions

In Figure III.2, the detailed sequence of operations involved in setting up a connection using hierarchic routing is described. The steps involved are listed below:

1) A connection request arrives at the connection controller (CC) from the connection request in interface, specified as a pair of SNPs at the edge of the top level routing area.

2) The routing controller (RC) is queried (using the Z end SNP over the route table query interface) and returns the set of links and subnetworks involved.

3) Link connections are obtained (in any order, i.e., 3a, or 3b in Figure III.2) from the link resource managers (LRM) over the link connection request interface.

4) Having obtained link connections (specified as SNP pairs), subnetwork connections can be requested from the child routing areas, by passing a pair of SNPs over the connection request in interface and confirming subnetwork connections to the CC via the connection request out interface. Again, the order of these operations is not fixed, the only requirement being that link connections are obtained before subnetwork connections can be created. The initial process now repeats recursively.

5) The child routing controllers now resolve a route between the SNPs specified.

6) Link connections are obtained (in any order) from the link resource managers (LRM) over the link connection request interface.

7) As a final step, the lowest level switches, which do not contain any routing or link allocation components at all, provide the necessary subnetwork connections.

8) The remaining steps indicate the flow of confirmations that the connection has been set up, culminating in step 10, where the confirmation is returned to the original user.

III.2 Source and step-by-step routing

While similar to hierarchical routing, for source routing, the connection control process is now implemented by a federation of distributed connection and routing controllers. The significant difference is that connection controllers invoke a different sequence of path computation functions between routing levels for hierarchical vs source routing. The signal flow for source (and step-by-step) routing is illustrated in Figure III.3.

In order to reduce the amount of network topology, each controller only needs to have available that portion of the topology that applies to its own routing area.

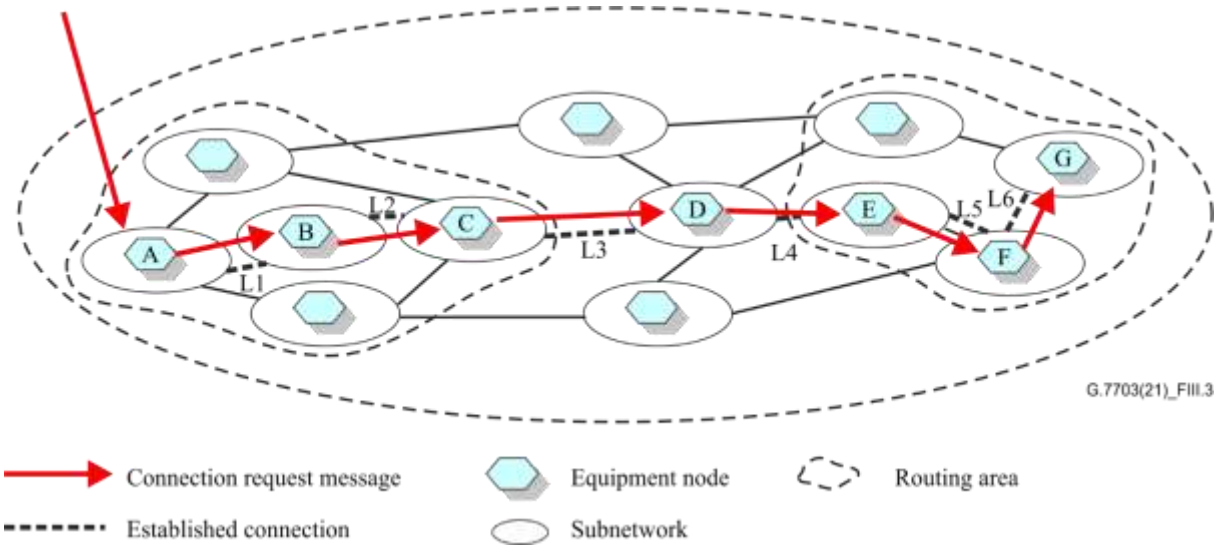


Figure III.3 – Source and step-by-step signalling flow

III.2.1 Source routing

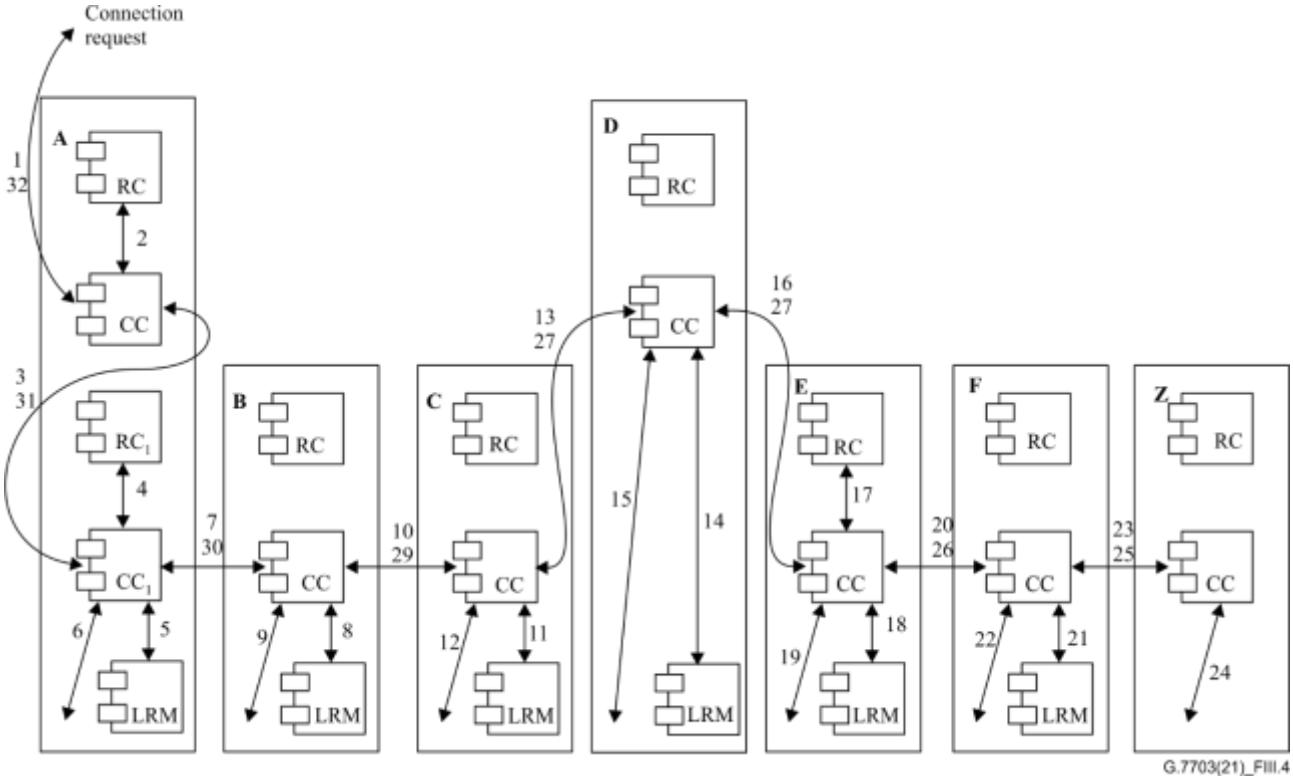


Figure III.4 – Source routing interactions

In the following steps, we describe the sequence of interactions shown in Figure III.4. The following notation is used: X_A represents the component at the highest level in node A, X_{An} represents the component that is at the next nth highest level in node A.

1) A connection request arrives at the connection controller (CC_A) from the connection request in interface, specified as a pair of names (A and Z) at the edge of the subnetwork.

2) The routing controller (RC_A) is queried (using the Z end SNP over the route table query interface) and returns route (A, L3, L4, Z).

3) As CC_A does not have access to the necessary link resource manager (LRM_C), the request (A, L3, L4, Z) is passed on to a peer CC_{A1} (over the connection request out/in interface), which controls routing through this routing area.

4) CC_{A1} queries RC_{A1} (over the route query interface) for L3 and obtains a list of additional links, L1 and L2.

5) Link L1 is local to this node, and a link connection for L1 is obtained from LRM_A over the link connection request interface.

6) The SNC is made across the local switch (controller not shown).

7) The request, now containing the remainder of the route (L2, L3, L4 and Z), is forwarded to the next peer CC_B (over the peer coordination out/in interface).

8) LRM_B controls L2, so a link connection is obtained from this link over the link connection request interface.

9) The SNC is made across the local switch (controller not shown).

10) The request, now containing the remainder of the route (L3, L4 and Z), is forwarded to the next peer CC_C (over the peer coordination out/in interface).

11) LRM_C controls L3, so a link connection is obtained from this link over the link connection request interface.

12) The SNC is made across the local switch (controller not shown).

13) The request, now containing the remainder of the route (L4, Z), is forwarded to the next peer CC_D (over the peer coordination out/in interface).

14) LRM_D controls L4, so a link connection is obtained from this link over the link connection request interface.

15) The SNC is made across the local switch (controller not shown).

16) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_E (over the peer coordination out/in interface).

17) CC_E queries RC_E (over the route table query interface) for Z and obtains links L5 and L6.

The process of connecting across the next routing area (i.e., steps 18 to 24 in Figure III.4) is identical to that already described. Events 25 to 32 describe the flow of confirmation signals to the connection originator.

III.2.2 Step-by-step routing

In this form of routing there is further reduction of routing information in the nodes, and this places restrictions upon the way in which routing is determined across the subnetwork. Figure III.5 applies to the network diagram of Figure III.3.

The process of step-by-step routing is identical to that described for source routing, with the following variation: routing controller RC_{A1} can only supply link L1, and does not supply link L2 as well. CC_B must then query RC_B (via the route table query interface) for L2 in order to obtain L2. A similar process of obtaining one link at a time is followed when connecting across the second routing area.

1) A connection request arrives at the connection controller (CC_A) from the connection request in interface, specified as a pair of names (A and Z) at the edge of the subnetwork.

2) The routing controller (RC_A) is queried (using the Z end SNP over the route table query interface) and returns the egress link, L3.

3) As CC_A does not have access to the necessary link resource manager (LRM_C), the request (A, L3, Z) is passed on to a peer CC_{A1} (over the connection request out/in interface), which controls routing through this routing area.

4) CC_{A1} queries RC_{A1} (over the route query interface) for L3 and obtains L1.

5) Link L1 is local to this node, and a link connection for L1 is obtained from LRM_A over the link connection request interface.

6) The SNC is made across the local switch (controller not shown).

7) The request, now containing the route (L3 and Z), is forwarded to the next peer CC_B (over the peer coordination out/in interface).

8) CC_{B1} queries RC_{B1} (over the route query interface) for L3 and obtains L2.

9) LRM_B controls L2, so a link connection is obtained from this link over the link connection request interface.

10) The SNC is made across the local switch (controller not shown).

11) The request, now containing the remainder of the route (L3 and Z), is forwarded to the next peer CCC (over the peer coordination out/in interface).

12) LRM_C controls L3, so a link connection is obtained from this link over the link connection request interface.

13) The SNC is made across the local switch (controller not shown).

14) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_D (over the peer coordination out/in interface).

15) CC_D queries RC_D (over the route query interface) for Z and obtains link L4.

16) LRM_D controls L4, so a link connection is obtained from this link over the link connection request interface.

17) The SNC is made across the local switch (controller not shown).

18) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_E (over the peer coordination out/in interface).

19) CC_E queries RC_E (over the route query interface) for Z and obtains link L5.

20) LRM_E controls L5, so a link connection is obtained from this link over the link connection request interface.

21) The SNC is made across the local switch (controller not shown).

22) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_F (over the peer coordination out/in interface).

23) CC_F queries RC_F (over the route query interface) for Z and obtains link L6.

24) LRM_F controls L6, so a link connection is obtained from this link over the link connection request interface.

25) The SNC is made across the local switch (controller not shown).

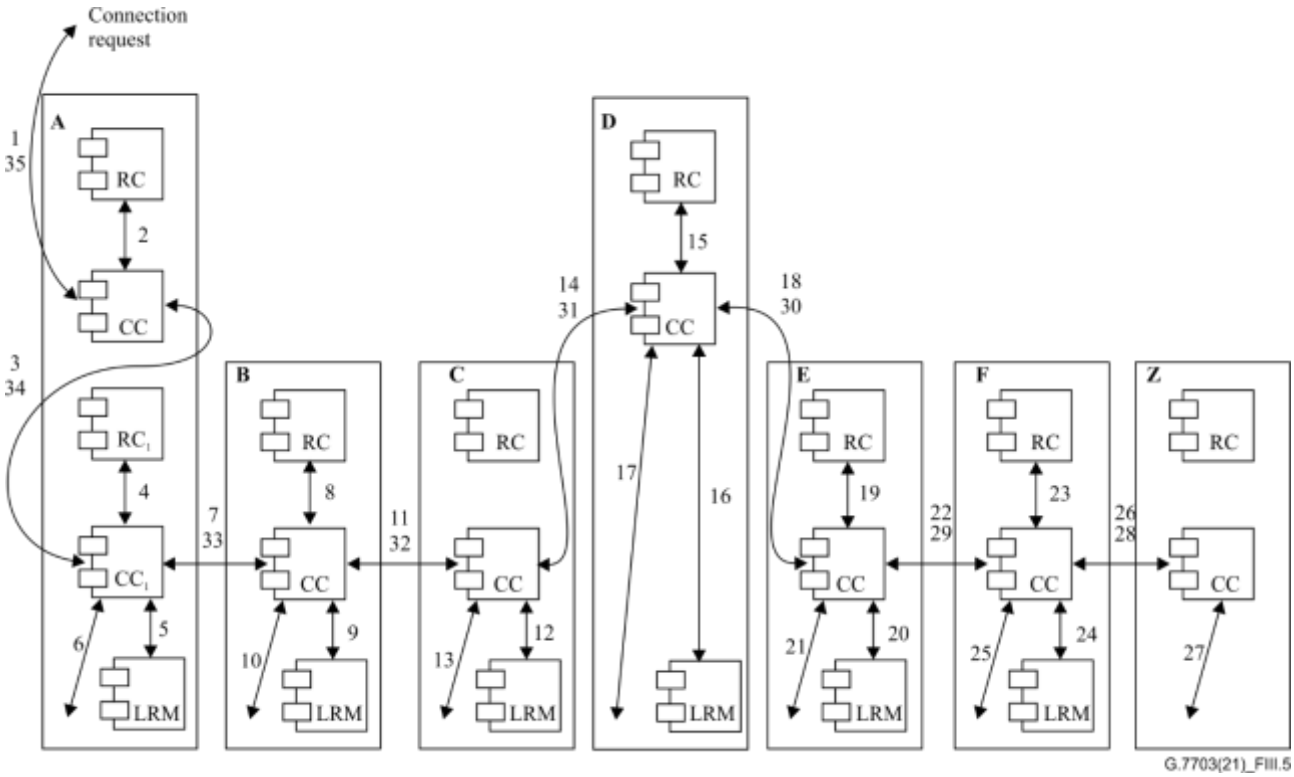


Figure III.5 – Step-by-step routing

III.2.3 Combination of source and step-by-step routing

Figure III.6 illustrates an example where source and step-by-step routing can be used, but at different routing levels. In this example, the low level routing is source routing, while the high level routing is step-by-step.

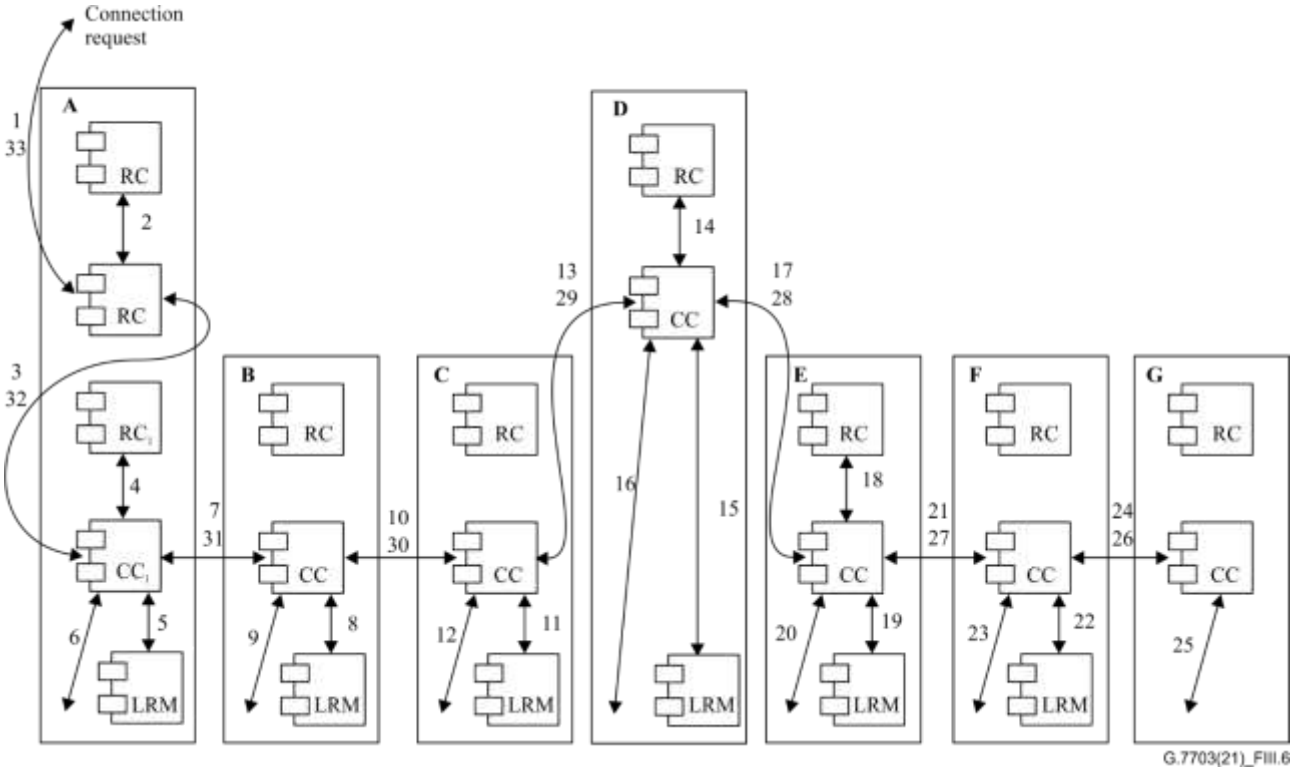


Figure III.6 – Combined source and step-by-step routing

1) A connection request arrives at the connection controller (CC_A) from the connection request in interface, specified as a pair of names (A and Z) at the edge of the subnetwork.

2) The routing controller (RC_A) is queried (using the Z end SNP over the route table query interface) and returns the egress link, L3.

3) As CC_A does not have access to the necessary link resource manager (LRM_C), the request (A, L3, Z) is passed on to a peer CC_{A1} (over the connection request out/in interface), which controls routing through this routing area.

4) CC_{A1} queries RC_{A1} (over the route table query interface) for L3 and obtains a list of additional links, L1 and L2.

5) Link L1 is local to this node, and a link connection for L1 is obtained from LRM_A over the link connection request interface.

6) The SNC is made across the local switch (controller not shown).

7) The request, now containing the remainder of the route (L2, L3 and Z), is forwarded to the next peer CC_B (over the peer coordination out/in interface).

8) LRM_B controls L2, so a link connection is obtained from this link over the link connection request interface.

9) The SNC is made across the local switch (controller not shown).

10) The request, now containing the remainder of the route (L3 and Z), is forwarded to the next peer CC_C (over the peer coordination out/in interface).

11) LRM_C controls L3, so a link connection is obtained from this link over the link connection request interface.

12) The SNC is made across the local switch (controller not shown).

13) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_D (over the peer coordination out/in interface).

14) CC_D queries RC_D (over the route table query interface) for Z and obtains link L4.

15) LRM_D controls L4, so a link connection is obtained from this link over the link connection request interface.

16) The SNC is made across the local switch (controller not shown).

17) The request, now containing the remainder of the route (Z), is forwarded to the next peer CC_E (over the peer coordination out/in interface).

18) CC_E queries RC_E (over the route table query interface) for Z and obtains links L5 and L6.

III.3 Connection protection

When the control domain is used to provide protection, a protection connection is set up to protect the working connection before the happening of a failure. After a working connection failure is detected, only the source and destination connection controllers are involved to complete the protection switching operation from the original working connection to protection connection.

Figure III.7 shows an example of connection protection using source based routing and distributed signalling. Here the protection signalling flow is shown after a link failure is detected. The relationship of the working and protection is assumed to be 1:1. That is, CI is not transferred on both working and protection at the same time. Instead, when the working path is interrupted by a link failure, the control domain is used to switch user CI to the protection path.

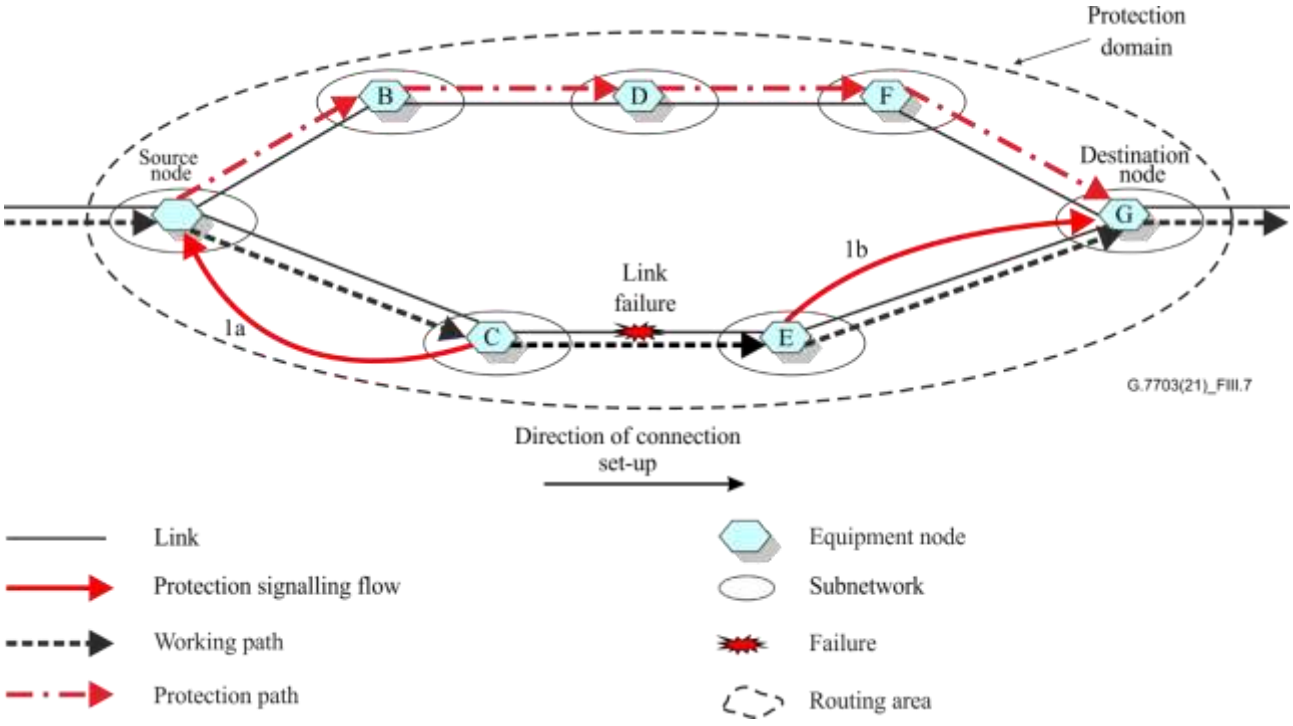


Figure III.7 – Protection signalling flow

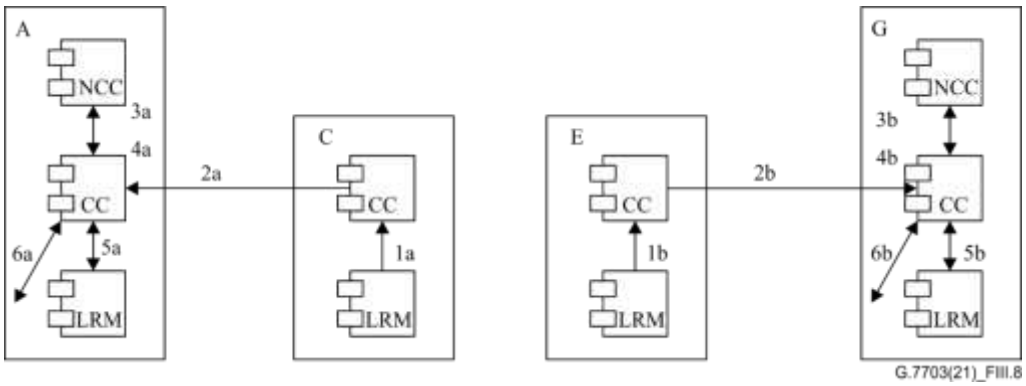


Figure III.8 – Protection interactions

In Figure III.8, the detailed sequence of operations involved in protection is described. The steps involved are listed below:

1) A bidirectional link failure notification generated by the link resource managers (LRM) arrives at the connection controller (CC) containing the failure link information. This occurs in node E and node C.

2) The link failure notification is forwarded to CC_A from CC_C and to CC_G from CC_E.

3) At both CC_A and CC_G, the NCCs are alerted to the failure of the working path.

4) The NCCs initiate the protection switching request to their CCs that cause the SNC to be made across the local switch from working connection to protection connection.

III.4 Restoration – Hard re-routing – Intra-domain – Hierarchical method

In hard re-routing, which is known as break-before-make, the original connection segment is released prior to the creation of an alternative connection segment.

Figure III.9 shows the signalling flow of a hard re-routing scenario with hierarchical connection control after an intra-domain link failure is detected. In the step of re-routing connection creation, hierarchical algorithm is adopted.

In Figure III.10, the detailed sequence of operations involved in Figure III.9 is described. The steps involved are listed below:

1) An intra-domain link failure notification generated by the link resource managers (LRM) arrives at the connection controller (CC), containing the crankback routing information which specifies the failure link. This may occur in node J or node H or both according to which node detects the link failure.

2) The intra-domain link failure notification is forwarded to CC_B.

3) Link connections are released (in any order, i.e., 3a, or 3b in Figure III.10) by LRM.

4) The SNCs are released by the lowest level switches.

5) The connection release confirmations are returned to CC_B.

6) The routing controller (RC_B) is queried with crankback routing information and returns the set of links excluding the failure link and subnetworks involved.

7-9) Steps 7 to 9 describe the flow of connection set-up using hierarchical algorithm which is identical to that described in clause III.1, Hierarchical routing.

10) If failed to set up the connection in re-routing domain A, the crankback routing information is forwarded to upper level re-routing domain C.

11) The remaining link connections are released by the LRM.

12) The SNCs are released by the lowest level switches. This requires release at nodes G and J via CC_B and then CC_G and CC_J.

13) The connection release confirmations are returned to CC_A. This includes release from CC_B.

14) RC_A is queried with crankback routing information and returns the set of links excluding the failure link and subnetworks involved.

15-21) Steps 15 to 21 describe the flow of connection set-up using hierarchical algorithm which is identical to that described in clause III.1, Hierarchical routing.

22) If failed to set up the connection in re-routing domain C, the crankback routing information is forwarded to upper level re-routing domain.

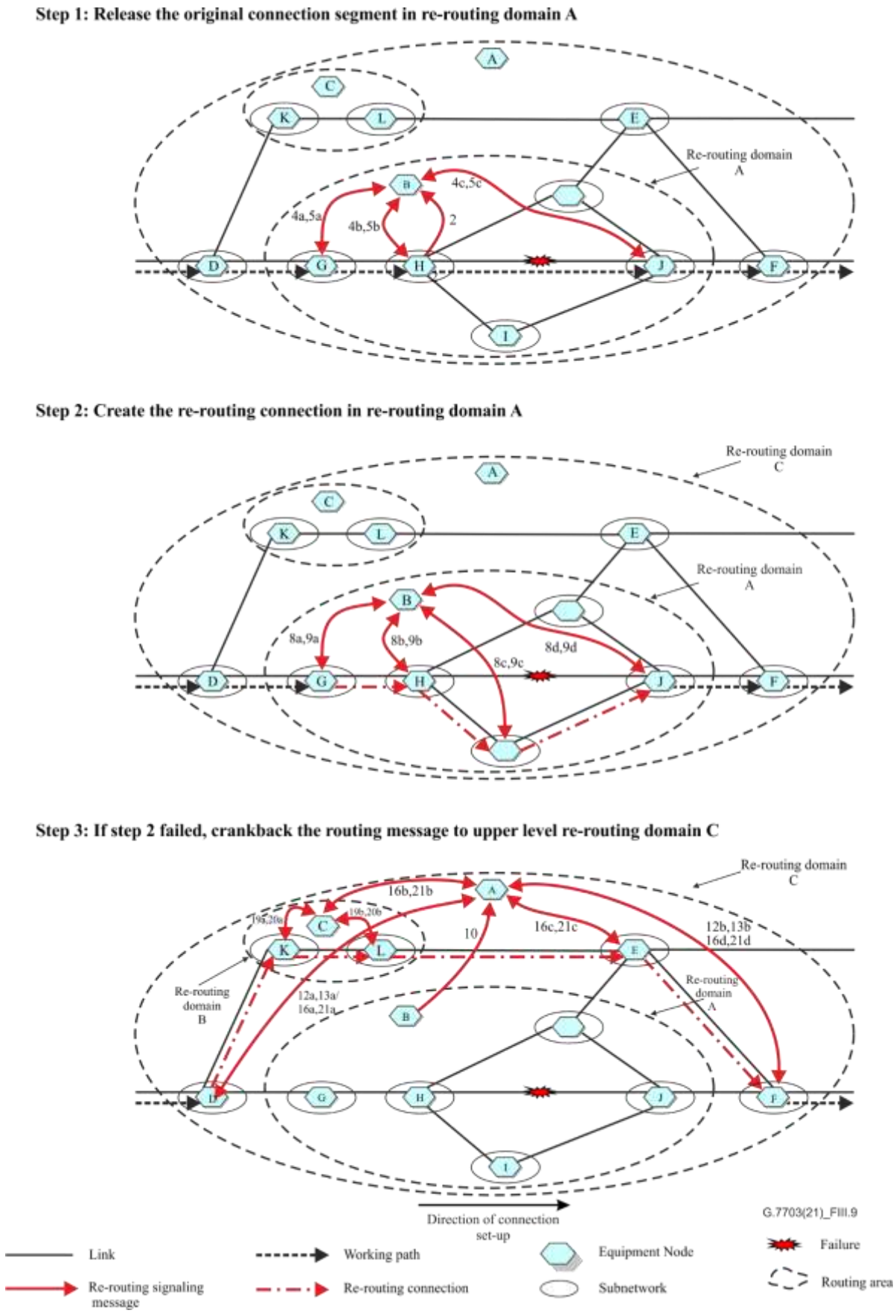


Figure III.9 – Signalling flow of hard re-routing using a hierarchical algorithm after an intra-domain link failure

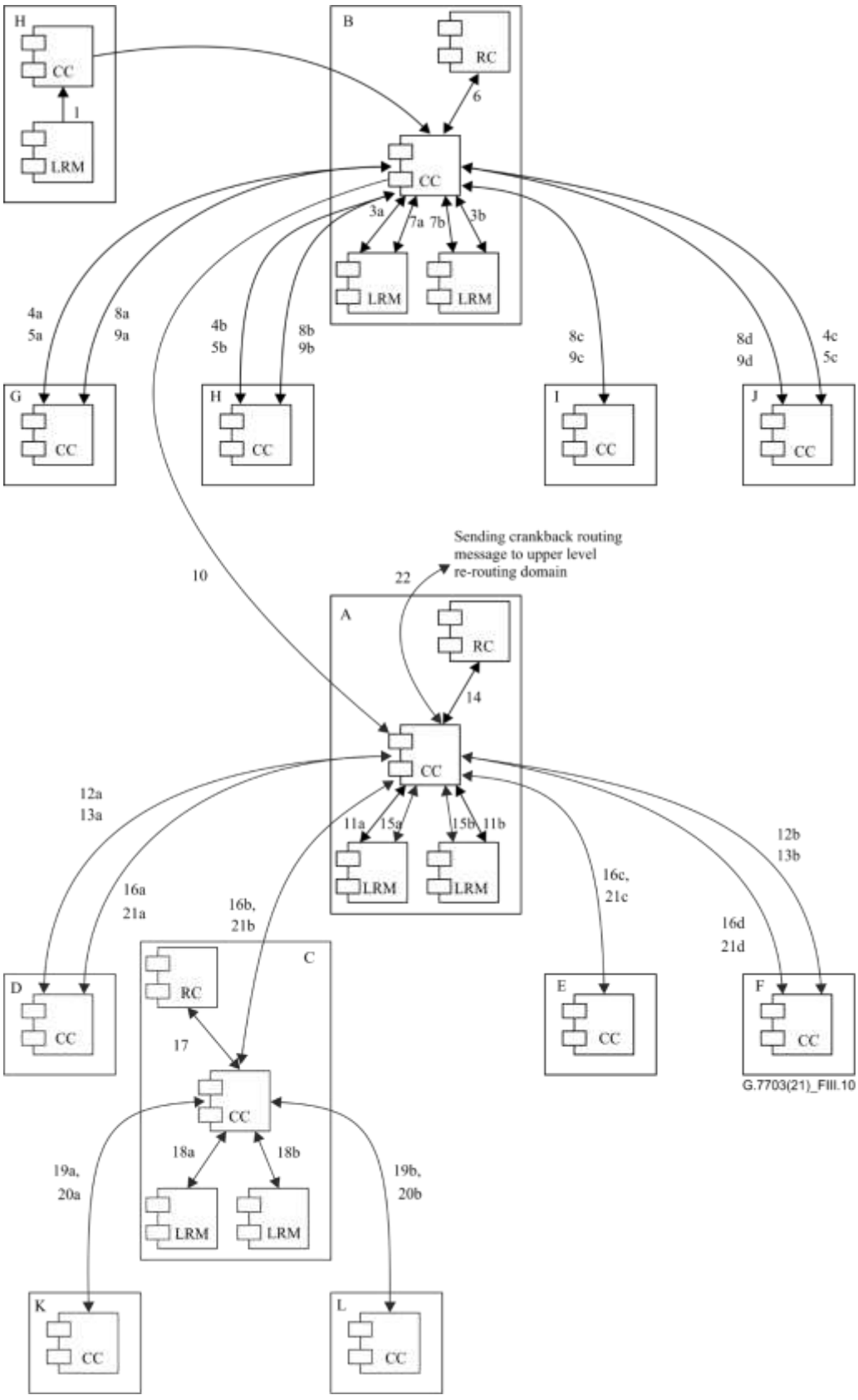


Figure III.10 – Component interactions of hard re-routing using a hierarchical algorithm after an intra-domain link failure

III.5 Restoration – Soft re-routing – Intra-domain – Source method

Soft re-routing service is a mechanism for the re-routing of a call for administrative purposes. When a re-routing operation is triggered (generally via a request from the MC system) and sent to the location of the re-routing components, the re-routing components establish a re-routing connection that traverses (or does not traverse) the appointed set of components according to the administrative purposes. In soft re-routing, which is known as make-before-break, the initial connection is deleted after the creation of a re-routing connection.

Figure III.11 shows the signalling flow of a soft re-routing scenario with source (or step-by-step) routing connection control after receiving a request from a MC system to re-route a connection excluding a certain intra-domain link.

In Figure III.12, the detailed sequence of operations using source routing involved in Figure III.11 is described. The steps involved are listed below:

1) A MC system request arrives at the connection controller (CC_G), containing constraints that the re-routing connection must comply with. For example, an explicit route of the re-routing connection. In this example, there is an exclusion constraint that specifies that link L1 is not to be used in the re-routing connection.

2a) Routing controller (RC_G) receives a re-routing connection set-up request initiated by CC_G containing the pair of SNPs at the edge of the re-routing domain A and the exclusion constraint.

2b) RC_G returns the set of links excluding the link L1.

3-15) Steps 3 to 15 describe the flow of connection set-up using a source routing algorithm which is identical to that described in clause III.2.3, Source and step-by-step routing. The new connection is joined to the original one coming into domain A at G and J.

If the connection is set up successfully in re-routing domain A, then step 16a is followed, otherwise step 16b is followed.

16a) The link connection of the original path is released by LRM_G in step 16a which consists of steps 16a1 and 16a2.

17) The SNC is released across the local switch.

18) The connection release request, containing the original connection information, is forwarded to CC_H.

19) The link connection of the original path is released by LRM_H.

20) The SNC is released across the local switch.

21) The connection release request, containing the original connection information, is forwarded to CC_J.

22) The SNC is released across the local switch.

23) The connection release confirmation is returned to the source CC_G and the re-routing process completes.

16b) The crankback routing information is forwarded to CC_D in upper level re-routing domain C.

17a) RC_D receives a re-routing connection set-up request initiated by CC_D containing the pair of SNPs at the edge of the re-routing domain C and the exclusion constraint to avoid domain A.

17b) RC_D returns the set of links excluding domain A.

18-39) Steps 18 to 39 describe the flow of connection set-up using a source routing algorithm which is identical to that described in clause III.2.3, Source and step-by-step routing.

40) The link connection of the original path is released by LRM_D in step 40 which consists of steps 40a and 40b.

41) The SNC is released across the local switch.

42) The connection release request, containing the original connection information, is forwarded to CC_G.

43) The link connection of the original path is released by LRM_G.

44) The SNC is released across the local switch.

45) The connection release request, containing the original connection information, is forwarded to CC_H.

46) The link connection of the original path is released by LRM_H.

47) The SNC is released across the local switch.

48) The connection release request, containing the original connection information, is forwarded to CC_J.

49) The link connection of the original path is released by LRM_J.

50) The SNC is released across the local switch.

51) The connection release request, containing the original connection information, is forwarded to CC_F.

52) The SNC is released across the local switch.

53) The connection release confirmation is returned to the source CC_D.

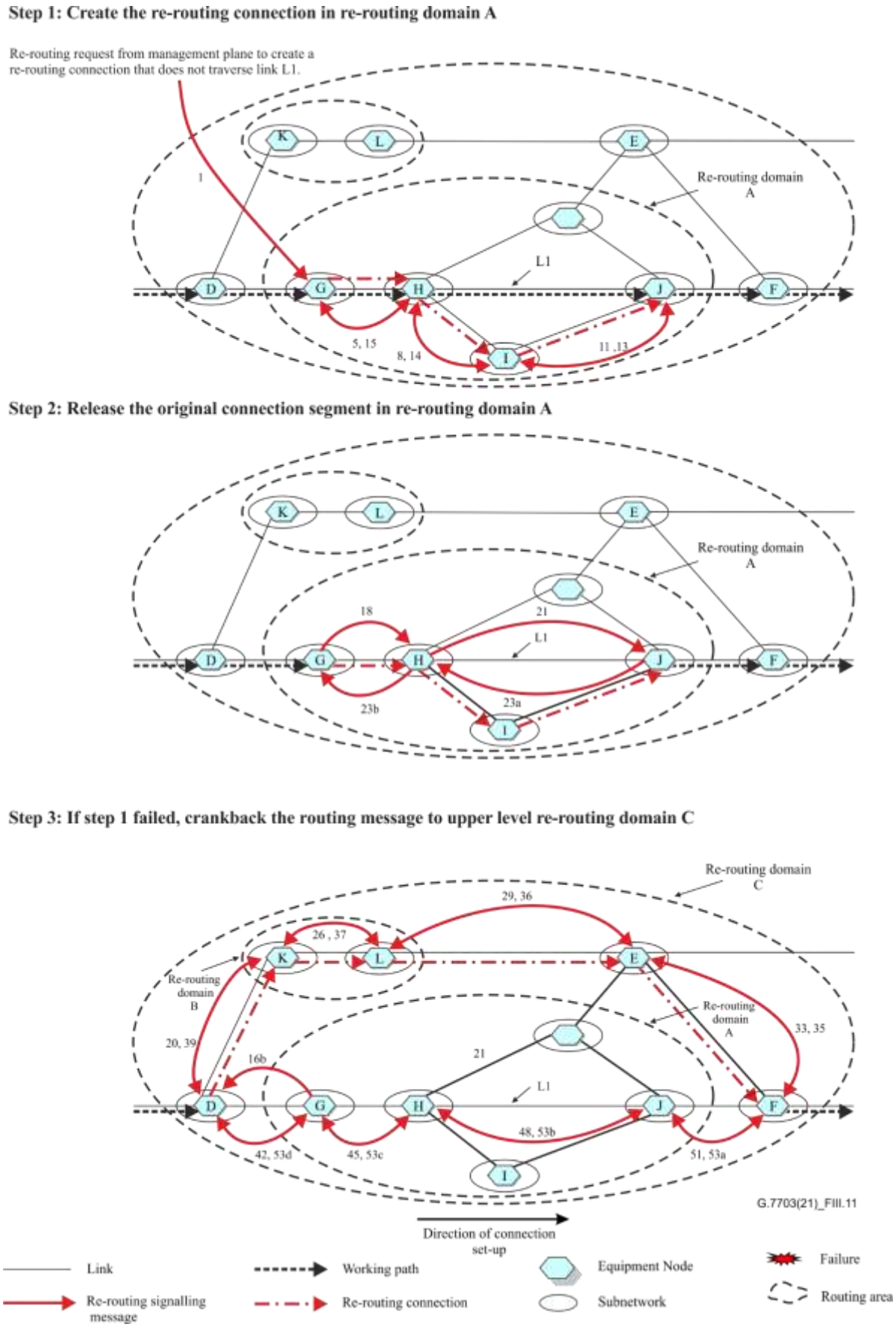


Figure III.11 – Signalling flow of soft re-routing using a source (or step-by-step) routing algorithm excluding an intra-domain link

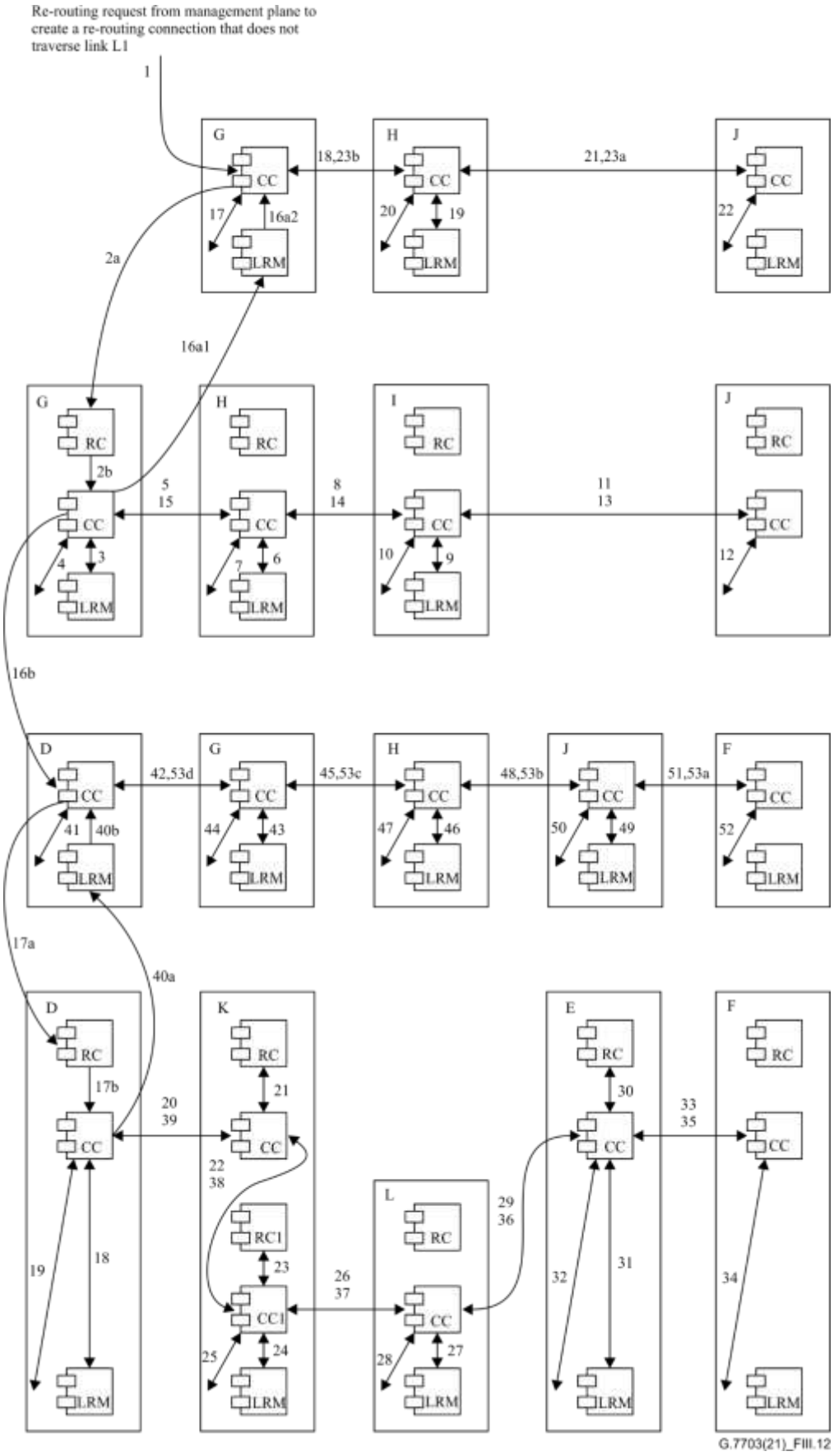


Figure III.12 – Component interactions of soft re-routing using a source routing algorithm excluding an intra-domain link

III.6 Restoration – Revertive re-routing – Intra-domain – Source method

In revertive behaviour re-routing, the original connection must not be released and is monitored by the network call controllers. When the failure is repaired, the call is restored to the original connection.

Figure III.13 shows the signalling flow of a revertive behaviour re-routing scenario with source (or step-by-step) routing connection control after an intra-domain link failure is detected.

In Figure III.14, the detailed sequence of operations using source routing involved in Figure III.13 is described. The steps involved are listed below:

1) An intra-domain link failure notification generated by the link resource managers (LRM) arrives at the connection controller (CC), containing the crankback routing information which specifies the failure link. This may occur in node J or node H or both according to which node detects the link failure.

2) The intra-domain link failure notification is forwarded to CC_G. No SCN changes are made.

3) The routing controller (RC_G) is queried with crankback routing information and returns the set of links excluding the failure link and subnetworks involved.

4-16) Steps 4 to 16 describe the flow of connection set-up using a source routing algorithm which is identical to that described in clause III.2.3, Source and step-by-step routing.

17) If failed to set up the connection in re-routing domain A, the crankback routing information is forwarded to upper level re-routing domain C.

18) RC_D is queried with crankback routing information and returns the set of links excluding domain A.

19-40) Steps 19 to 40 describe the flow of connection set-up using a source routing algorithm which is identical to that described in clause III.2.3, Source and step-by-step routing.

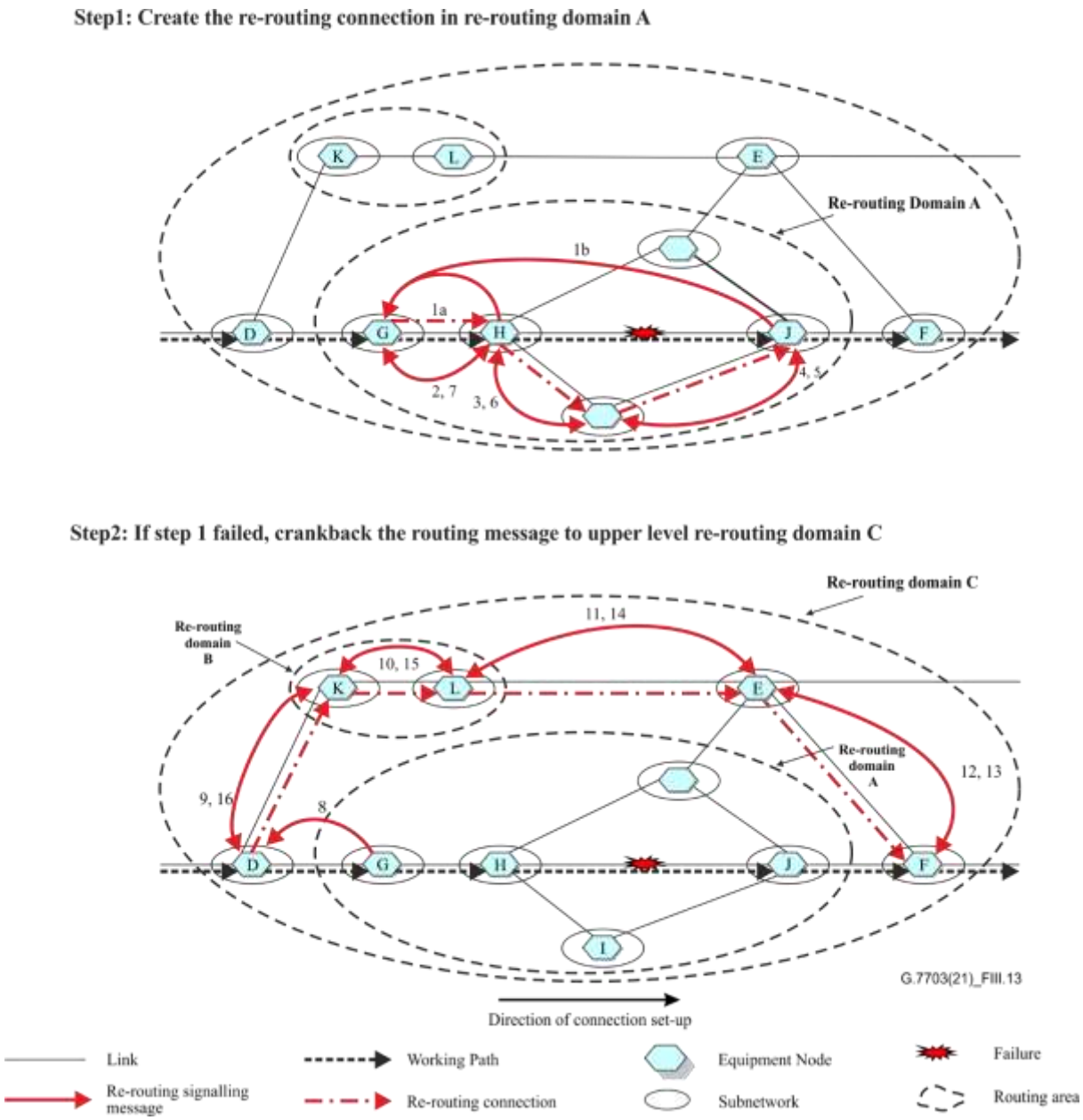


Figure III.13 – Signalling flow of revertive behaviour re-routing using a source (or step-by-step) routing algorithm after an intra-domain link failure

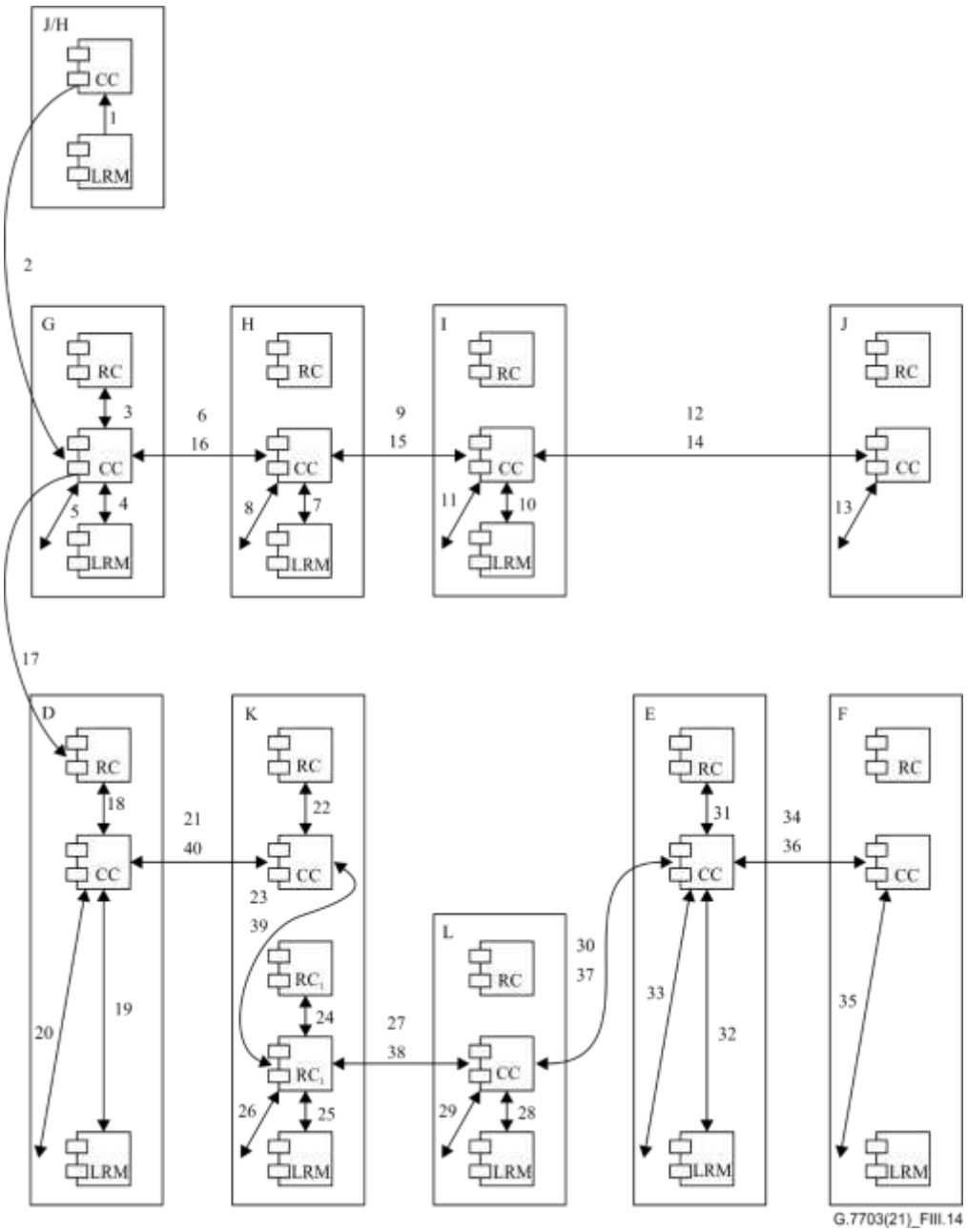


Figure III.14 – Component interactions of revertive behaviour re-routing using a source routing algorithm after an intra-domain link failure

III.7 Source routing using a routing query interface

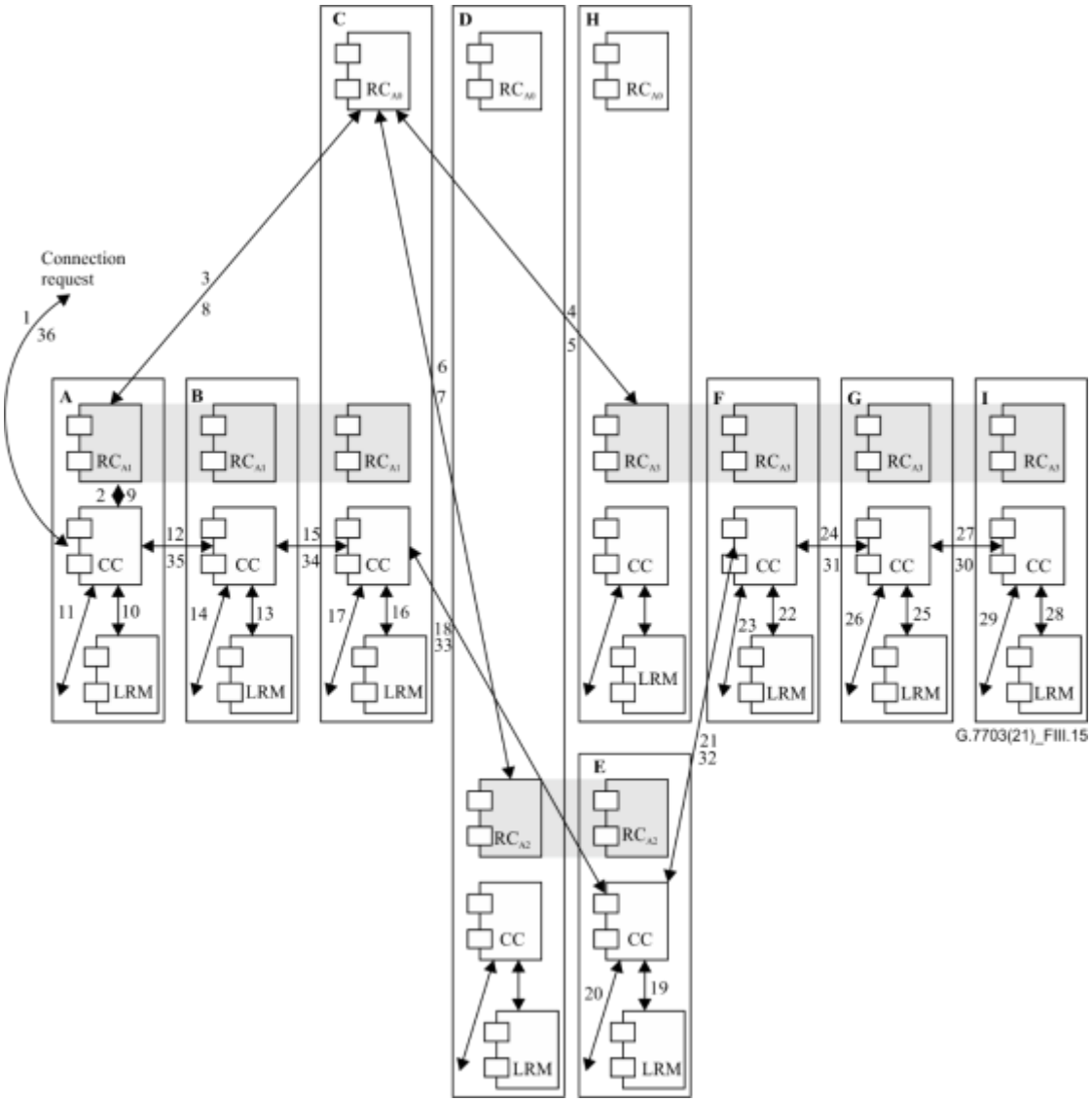


Figure III.15 – Component interactions for source routing using a routing query interface

Figure III.15 illustrates the detailed sequence of operations involved in setting up a connection using source routing assisted by RC-RC route query. The notation RC_{A1}, RC_{A2}, etc., represent the routing controller in areas A1, A2, etc. The actual communication components may be facilitated by other intermediate components – for example, the communication from RC_{A0} on node C to RC_{A2} on node D may be performed by transferring the message through RC_{A0} on node D.

The steps involved are listed below.

1) A connection request arrives at the connection controller (CC_A) from the connection_request_in interface, specified as a pair of names (A and Z) at the edge of the subnetwork.

2) The routing controller RC_{A1} on node A is queried (using the Z end SNP over the route query interface).

3) The routing controller RC_{A1} on node A recognizes that the destination address is not visible within area A1 so it sends a route query to RC_{A0} on node C for assistance over the route query interface. While RC_{A1} on node C has the same routing information as RC_{A1} on node A as they are in a common routing area, RC_{A0} on node C has visibility to the destination making the computation of a path possible.

4) In the process of computing a path to the destination, RC_{A0} on node C recognizes that to reach the destination it needs to reach area A3. However, since there are multiple paths between area A1 and area A3, it needs the assistance of RC_{A2} and RC_{A3} to determine the best path. Thus, a query is sent by RC_{A0} on node C to RC_{A3} on node H to determine which link from A2 to A3 should be used.

5) RC_{A3} on node H computes the possible paths from the links entering area A3 from area A2 to the destination within area A3. From this, it can determine the costs of using either of the paths, and returns this information to RC_{A0} on node C.

6) As with RC_{A3} on node H, RC_{A0} on node C sends a query to RC_{A2} on node D to determine the paths between the egress links that egress area A2 and enter area A3 and the ingress links that enter area A2 from area A1.

7) RC_{A2} on node D computes the possible paths across area A2, and returns this information to RC_{A0} on node C.

8) RC_{A0} on node C provides to RC_{A1} on node A the list of paths developed from the edge of area A1 to the destination in area A3 and includes the aggregate cost for each path developed.

9) RC_{A1} on node A now has the necessary information to compute a path across area A1 utilizing the cost information provided by RC_{A0} on node C to determine the lowest cost end-to-end path. For the remainder of this example, we assume the path chosen is from A, via L1 to B, via L2 to C, via L3 to E, via L4 to F, via L5 to G, and via L6 to I. It then sends the response back to CC on node A, which starts the process to form the end-to-end connection request using route (A, L1, L2, L3, L4, L5, L6 and Z).

10) L1 is local to node A, and a link connection for L1 is obtained from LRM_A over the link connection request interface.

11) The appropriate SNC is established on the local switch (controller not shown).

12) The connection request (L2, L3, L4, L5, L6 and Z) is then forwarded to the next CC on node B (over the peer coordination_out/in interface).

13) LRM_B controls L2, so a link connection is obtained from this link over the link connection_request interface.

14) The appropriate SNC is established on the local switch (controller not shown).

15) The connection request (L3, L4, L5, L6 and Z) is then forwarded to the next CC on node C (over the peer coordination_out/in interface).

16) LRM_C controls L3, so a link connection is obtained from this link over the link connection_request interface.

17) The appropriate SNC is established on the local switch (controller not shown).

18) The connection request (L4, L5, L6 and Z) is then forwarded to the next CC on node E (over the peer coordination_out/in interface).

19) LRM_E controls L4, so a link connection is obtained from this link over the link connection_request interface.

20) The appropriate SNC is established on the local switch (controller not shown).

21) The connection request (L5, L6 and Z) is then forwarded to the next CC on node F (over the peer coordination_out/in interface).

22) LRM_F controls L5, so a link connection is obtained from this link over the link connection_request interface.

23) The appropriate SNC is established on the local switch (controller not shown).

24) The connection request (L6 and Z) is then forwarded to the next peer CC on node G (over the peer coordination_out/in interface).

25) LRM_G controls L6, so a link connection is obtained from this link over the link connection_request interface.

26) The appropriate SNC is established on the local switch (controller not shown).

27) The connection request (Z) is then forwarded to the next CC on node I.

28) LRM_I controls the egress link to the destination node, so a link connection is obtained from this link over the link connection request interface.

29) The appropriate SNC is established on the local switch (controller not shown).

30) The CC on node I then sends a confirmation back to the CC on node G. The exchange of responses then repeats between pairs of CCs all the way going back to the connection originator CC on node A.

Appendix IV

Example of explicit multi-layer routing topology

(This appendix does not form an integral part of this Recommendation.)

In some situations, connection routing can benefit from a topology view that includes explicit detail from multiple layer networks. One example of this is an ODU layer network that contains within it subnetworks supported by transparent optical (OTSiA) layer networks.

If the OTSiA server layer network topology is projected into the ODU client layer it is not possible to associate different routing attributes or constraints specific to the transparent optical subnetworks with that topology. Figure IV.1 shows an example of an explicit multi-layer topology for such a network. In this example the transitions from the ODU layer to the OTSiA layer are shown using transitional SNPP links with an adaptation/termination icon. This indicates the transition from ODUk to OTUk to OTSiA occurs between the routing areas connected by these links. The structure of the OTSiA layer topology is two rings interconnected via 3R regenerators. The presence of the regenerators is shown using transitional SNPP links with singleton layer processor icons (diamond shape). This indicates the presence of a client layer (OTU) dependent function between the ends of these links.

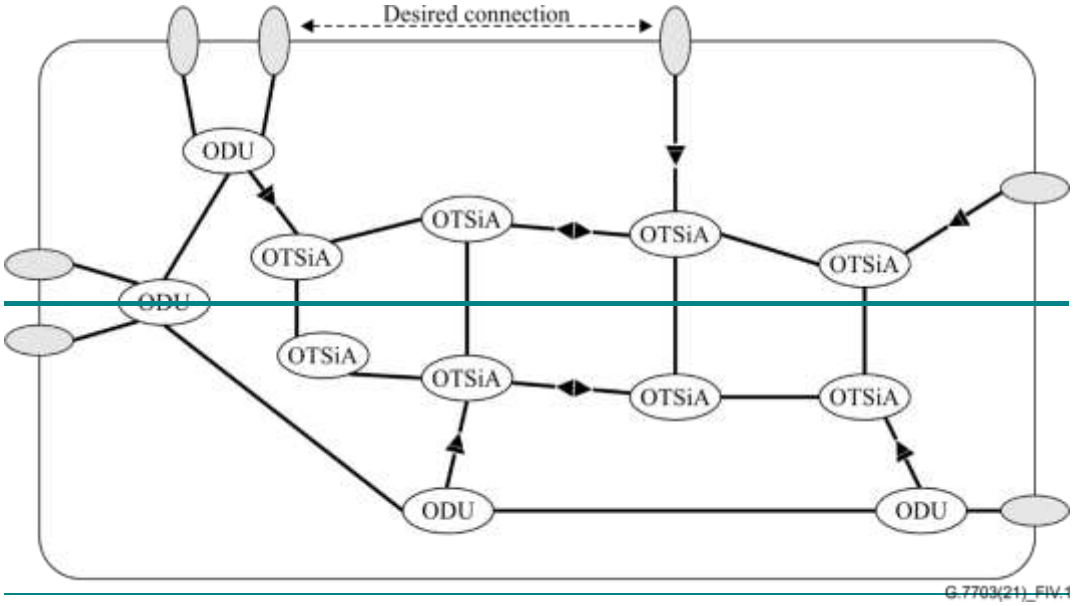


Figure IV.1 – Explicit multi-layer topology example

Attributes can be associated with the transitional SNPP links and OTSiA layer SNPP links that are specific to OTSiA layer network routing (e.g., modulation type, FEC type, wavelength, etc.). These OTSiA specific attributes would not be associated with the ODU SNPP links (i.e., those between a pair of ODU routing areas).

This topology may be used to calculate ODUk paths that cross both ODU and OTSiA routing areas and meet both ODU path constraints and, where necessary, OTSiA path constraints. This facilitates more optimal route selection across the entire network. The use of an explicit multi-layer topology in this case is particularly straightforward since the relationship between ODUk, OTUk, and OTSiA is 1:1:1. Therefore potential concerns about allocating more server-layer resources than are required by the client layer path do not arise.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	**Transmission systems and media, digital systems and networks**
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Published in Switzerland
Geneva, 2023