

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**G.8080/Y.1304**

(02/2012)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA,  
DIGITAL SYSTEMS AND NETWORKS

Packet over Transport aspects – Ethernet over Transport  
aspects

SERIES Y: GLOBAL INFORMATION  
INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS  
AND NEXT-GENERATION NETWORKS

Internet protocol aspects – Transport

---

## **Architecture for the automatically switched optical network**

Recommendation ITU-T G.8080/Y.1304



**Documents**

Document Sharing Hub

# 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        |
| PACKET OVER TRANSPORT ASPECTS                                                                                                                | G.8000–G.8999        |
| <b>Ethernet over Transport aspects</b>                                                                                                       | <b>G.8000–G.8099</b> |
| MPLS over Transport aspects                                                                                                                  | G.8100–G.8199        |
| Quality and availability targets                                                                                                             | G.8200–G.8299        |
| Service Management                                                                                                                           | G.8600–G.8699        |
| ACCESS NETWORKS                                                                                                                              | G.9000–G.9999        |

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



# Recommendation ITU-T G.8080/Y.1304

## Architecture for the automatically switched optical network

### Summary

Recommendation ITU-T G.8080/Y.1304 describes the reference architecture for the control plane of the automatically switched optical network as applicable to connection-oriented circuit or packet transport networks, as defined in Recommendation ITU-T G.805. This reference architecture is described in terms of the key functional components and the interactions between them.

### History

| Edition | Recommendation                    | Approval   | Study Group |
|---------|-----------------------------------|------------|-------------|
| 1.0     | ITU-T G.8080/Y.1304               | 2001-11-29 | 15          |
| 1.1     | ITU-T G.8080/Y.1304 (2001) Amd. 1 | 2003-03-16 | 15          |
| 1.2     | ITU-T G.8080/Y.1304 (2001) Amd. 2 | 2005-02-22 | 15          |
| 2.0     | ITU-T G.8080/Y.1304               | 2006-06-06 | 15          |
| 2.1     | ITU-T G.8080/Y.1304 (2006) Cor. 1 | 2007-09-06 | 15          |
| 2.2     | ITU-T G.8080/Y.1304 (2006) Amd. 1 | 2008-03-29 | 15          |
| 2.3     | ITU-T G.8080/Y.1304 (2006) Amd. 2 | 2010-09-06 | 15          |
| 3.0     | ITU-T G.8080/Y.1304               | 2012-02-13 | 15          |

## 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 received notice of intellectual property, protected by patents, 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 TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## Table of Contents

|    |                                                                        | Page |
|----|------------------------------------------------------------------------|------|
| 1  | Scope .....                                                            | 1    |
| 2  | References.....                                                        | 1    |
| 3  | Definitions .....                                                      | 2    |
| 4  | Abbreviations and acronyms .....                                       | 3    |
| 5  | Overview .....                                                         | 5    |
|    | 5.1 Call and connection control.....                                   | 7    |
|    | 5.2 Interaction between control, transport and management planes ..... | 8    |
| 6  | Transport resources and their organization.....                        | 10   |
|    | 6.1 Transport entities .....                                           | 10   |
|    | 6.2 Routing areas .....                                                | 13   |
|    | 6.3 Topology and discovery .....                                       | 17   |
|    | 6.4 Domains.....                                                       | 18   |
|    | 6.5 Multi-layer aspects .....                                          | 20   |
|    | 6.6 Interlayer client support.....                                     | 23   |
|    | 6.7 Calls supported by calls at same layer.....                        | 25   |
|    | 6.8 Mapped server interlayer relationships .....                       | 25   |
| 7  | Control plane architecture.....                                        | 26   |
|    | 7.1 Notation .....                                                     | 27   |
|    | 7.2 Policy and federations .....                                       | 28   |
|    | 7.3 Architectural components.....                                      | 31   |
|    | 7.4 Protocol controller (PC) components .....                          | 62   |
| 8  | Reference points .....                                                 | 64   |
|    | 8.1 UNI .....                                                          | 65   |
|    | 8.2 I-NNI .....                                                        | 66   |
|    | 8.3 E-NNI .....                                                        | 66   |
|    | 8.4 User architecture.....                                             | 66   |
|    | 8.5 Inter-layer NCC interactions .....                                 | 67   |
| 9  | Control plane use of signalling control networks (SCNs) .....          | 68   |
| 10 | Identifiers.....                                                       | 69   |
|    | 10.1 Name spaces .....                                                 | 69   |
|    | 10.2 Names and addresses .....                                         | 70   |
|    | 10.3 Relationships between identifiers.....                            | 71   |
| 11 | Connection availability enhancement techniques.....                    | 71   |
|    | 11.1 Protection.....                                                   | 72   |
|    | 11.2 Restoration.....                                                  | 72   |
|    | 11.3 Nested routing domains .....                                      | 75   |

|                                                                                | <b>Page</b> |
|--------------------------------------------------------------------------------|-------------|
| 12 Resilience.....                                                             | 76          |
| 12.1 Principles of control and transport plane interactions .....              | 76          |
| 12.2 Principles of protocol controller communication .....                     | 77          |
| 12.3 Control and management plane interactions .....                           | 77          |
| Annex A – Connection services.....                                             | 78          |
| Appendix I – Resilience relationships .....                                    | 81          |
| I.1 Control plane – DCN relationships .....                                    | 81          |
| I.2 Control plane – Transport plane relationships.....                         | 82          |
| I.3 Control plane – Management plane relationships .....                       | 83          |
| I.4 Intra-control plane relationships.....                                     | 84          |
| Appendix II – Example of layered call control.....                             | 86          |
| Appendix III – Component interactions for connection set-up.....               | 87          |
| III.1 Hierarchical routing.....                                                | 87          |
| III.2 Source and step-by-step routing .....                                    | 89          |
| III.3 Connection protection .....                                              | 94          |
| III.4 Restoration – Hard re-routing – Intra-domain – Hierarchical method ..... | 95          |
| III.5 Restoration – Soft re-routing – Intra-domain – Source method .....       | 99          |
| III.6 Restoration – Revertive re-routing – Intra-domain – Source method .....  | 103         |
| III.7 Source routing using a routing query interface .....                     | 106         |
| Appendix IV – Combining protection and restoration domains .....               | 109         |
| Appendix V – Example of explicit multi-layer routing topology .....            | 114         |
| Bibliography.....                                                              | 115         |

Copyright © 2012 ITU-T. All rights reserved.

# Recommendation ITU-T G.8080/Y.1304

## Architecture for the automatically switched optical network

### 1 Scope

This Recommendation specifies the architecture and requirements for the automatic switched transport network as applicable to connection-oriented circuit or packet transport networks, as defined in [ITU-T G.805].

This Recommendation describes the set of control plane 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.

For the purposes of this Recommendation, components are used to represent abstract entities rather than instances of implementable software. UML-like notation is used to describe components of the architecture of the automatically switched optical network.

### 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.705]  | Recommendation ITU-T G.705 (2000), <i>Characteristics of plesiochronous digital hierarchy (PDH) equipment functional blocks.</i>                                                                                                             |
| [ITU-T G.800]  | Recommendation ITU-T G.800 (2012), <i>Unified functional architecture of transport networks.</i>                                                                                                                                             |
| [ITU-T G.805]  | Recommendation ITU-T G.805 (2000), <i>Generic functional architecture of transport networks.</i>                                                                                                                                             |
| [ITU-T G.7718] | Recommendation ITU-T G.7718/Y.1709 (2005), <i>Framework for ASON management.</i>                                                                                                                                                             |
| [ITU-T G.8081] | Recommendation ITU-T G.8081/Y.1353 (2010), <i>Terms and definitions for automatically switched optical networks.</i>                                                                                                                         |
| [ITU-T M.3100] | Recommendation ITU-T M.3100 (2005), <i>Generic network information model.</i>                                                                                                                                                                |
| [ITU-T X.25]   | Recommendation ITU-T X.25 (1996), <i>Interface between Data Terminal Equipment (DTE) and Data Circuit-terminating Equipment (DCE) for terminals operating in the packet mode and connected to public data networks by dedicated circuit.</i> |
| [ITU-T Y.1311] | Recommendation ITU-T Y.1311 (2002), <i>Network-based VPNs – Generic architecture and service requirements.</i>                                                                                                                               |
| [ITU-T Y.1312] | Recommendation ITU-T Y.1312 (2003), <i>Layer 1 Virtual Private Network generic requirements and architecture elements.</i>                                                                                                                   |

### 3 Definitions

This Recommendation uses the following terms defined elsewhere:

- 3.1 **access group (AG):** [ITU-T G.805].
- 3.2 **adaptation:** [ITU-T G.805].
- 3.3 **address:** [ITU-T G.8081].
- 3.4 **administrative domain:** [ITU-T G.805].
- 3.5 **agent:** [ITU-T G.8081].
- 3.6 **allocated (resource) label range:** [ITU-T G.8081].
- 3.7 **assigned SNPs:** [ITU-T G.8081].
- 3.8 **call:** [ITU-T G.8081].
- 3.9 **call segment:** [ITU-T G.8081].
- 3.10 **characteristic information:** [ITU-T G.800].
- 3.11 **closed user group:** [ITU-T X.25].
- 3.12 **component:** [ITU-T G.8081].
- 3.13 **configured (resource) label:** [ITU-T G.8081].
- 3.14 **connection:** [ITU-T G.8081].
- 3.15 **connection point (CP):** [ITU-T G.8081].
- 3.16 **connection termination point (CTP):** [ITU-T G.8081].
- 3.17 **control domain:** [ITU-T G.8081].
- 3.18 **control plane:** [ITU-T G.8081].
- 3.19 **control plane configured protection:** [ITU-T G.8081].
- 3.20 **E-NNI:** [ITU-T G.8081], See also clause 8.
- 3.21 **forwarding port (FPt):** [ITU-T G.800].
- 3.22 **hard re-routing:** [ITU-T G.8081].
- 3.23 **I-NNI:** [ITU-T G.8081], See also clause 8.
- 3.24 **interface:** [ITU-T G.8081].
- 3.25 **layer network:** [ITU-T G.805].
- 3.26 **link:** [ITU-T G.805].
- 3.27 **link connection:** [ITU-T G.805].
- 3.28 **management plane:** [ITU-T G.8081].
- 3.29 **multi-homed:** [ITU-T G.8081].
- 3.30 **name:** [ITU-T G.8081].
- 3.31 **permanent connection:** [ITU-T G.8081].
- 3.32 **policy:** [ITU-T G.8081].

- 3.33 **port controller:** [ITU-T G.8081].
- 3.34 **potential (resource) label range:** [ITU-T G.8081].
- 3.35 **potential SNPs:** [ITU-T G.8081].
- 3.36 **re-routing domain:** [ITU-T G.8081].
- 3.37 **restoration:** [ITU-T G.8081].
- 3.38 **route:** [ITU-T G.8081].
- 3.39 **routing area:** [ITU-T G.8081].
- 3.40 **routing control domain:** [ITU-T G.8081].
- 3.41 **routing level:** [ITU-T G.8081].
- 3.42 **service level agreement:** [ITU-T G.8081].
- 3.43 **soft permanent connection (SPC):** [ITU-T G.8081].
- 3.44 **soft rerouting:** [ITU-T G.8081].
- 3.45 **subnetwork:** [ITU-T G.8081].
- 3.46 **subnetwork connection (SNC):** [ITU-T G.8081].
- 3.47 **subnetwork point (SNP):** [ITU-T G.8081].
- 3.48 **subnetwork point pool (SNPP):** [ITU-T G.8081].
- 3.49 **subnetwork point pool link (SNPP link):** [ITU-T G.8081].
- 3.50 **supplementary services:** [ITU-T G.8081].
- 3.51 **switched connection (SC):** [ITU-T G.8081].
- 3.52 **termination connection point (TCP):** [ITU-T G.8081].
- 3.53 **third party signalling:** [ITU-T G.8081].
- 3.54 **trail:** [ITU-T G.805].
- 3.55 **trail termination point (TTP):** [ITU-T G.8081].
- 3.56 **transitional SNPP link:** [ITU-T G.8081].
- 3.57 **transport domain:** [ITU-T G.8081].
- 3.58 **transport plane:** [ITU-T G.8081].
- 3.59 **transport resource identifier:** [ITU-T G.8081].
- 3.60 **user-network interface for the control plane (UNI):** [ITU-T G.8081], see also clause 8.
- 3.61 **virtual private network:** [ITU-T Y.1311].

#### **4 Abbreviations and acronyms**

This Recommendation uses the following abbreviations and acronyms:

|     |                               |
|-----|-------------------------------|
| AG  | Access Group                  |
| AGC | Access Group Container        |
| CC  | Connection Controller         |
| CCC | Calling party Call Controller |
| CI  | Characteristic Information    |

|       |                                                         |
|-------|---------------------------------------------------------|
| CoS   | Class of Service                                        |
| CP    | Connection Point                                        |
| CPS   | Connection Point Status                                 |
| 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 |
| FwP   | Forwarding Point                                        |
| HOVC  | Higher Order Virtual Container                          |
| id    | identifier                                              |
| I-NNI | Internal Network-Network Interface (reference point)    |
| LOVC  | Lower Order Virtual Container                           |
| LRM   | Link Resource Manager                                   |
| MI    | Management Information                                  |
| MO    | Managed Object                                          |
| NCC   | Network Call Controller                                 |
| PC    | Protocol Controller                                     |
| PDP   | Policy Decision Point                                   |
| PEP   | Policy Enforcement Point                                |
| RC    | Routing Controller                                      |
| SCN   | Signalling Control Network                              |
| SNC   | Subnetwork Connection                                   |
| SNP   | Subnetwork Point                                        |
| SNPP  | Subnetwork Point Pool                                   |
| SPC   | Soft Permanent Connection                               |
| SC    | Switched Connection                                     |
| TAP   | Termination and Adaptation Performer                    |
| TCP   | Termination Connection Point                            |
| TTP   | Trail Termination Point                                 |
| UML   | Unified Modelling Language                              |
| UNI   | User-Network Interface (reference point)                |
| VPN   | Virtual Private Network                                 |

## 5 Overview

The purpose of the automatic switched optical network (ASON) control plane 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 plane architecture should give service providers control of their network, while providing fast and reliable call set-up. The control plane itself 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 plane components).

The ASON control plane is composed of different components that provide specific functions including that of route determination and signalling. The control plane 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 plane architectural components and the interaction between the control plane, management plane and transport plane. The management and transport planes are specified in other ITU-T Recommendations and are outside the scope of this Recommendation.

Figure 5.1 provides a high-level view of the interactions of the control, management and transport planes for the support of switched connections of a layer network. Also included in this figure is the DCN, which provides the communication paths to carry signalling and management information. The details of the DCN, management plane and the transport plane are outside the scope of this Recommendation. Functions pertaining to the control plane are described in this Recommendation. Management of the control plane is specified in [ITU-T G.7718].

Control plane 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 control plane 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.

The control plane 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 plane and provided to the control plane.

The control plane 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 plane (via the overhead/OAM) and via the management plane (including the DCN).

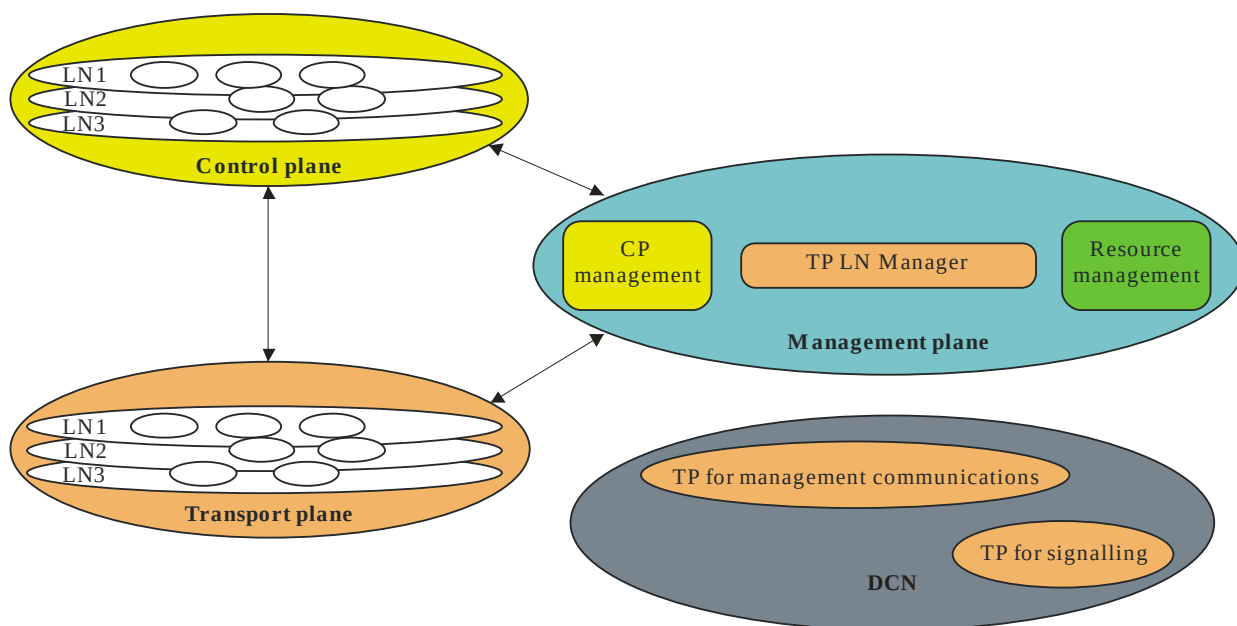
The interconnection between and within domains is described in terms of reference points. As domains are established via operator policies, inter-domain reference points are service demarcation points for a single service layer (i.e., points where call control is provided). The exchange of information 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 UNI, which represents a user-provider service demarcation point. The reference point between domains is the E-NNI, which represents a service demarcation point supporting multi-domain connection establishment. The reference point within a domain is an I-NNI, which represents a connection point supporting intra-domain connection establishment. The information flows across these reference points are further described in clause 8.

The control plane 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. The case where a portion of the resources are dynamically shared is for further study.

Separate descriptions are provided for the interactions between the:

- control plane and transport plane layer networks; and the
- management plane and transport plane resulting from the addition of the control plane 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.



G.8080-Y.1304(12)\_F5.1

**Figure 5.1 – Relationship between architectural components**

## 5.1 Call and connection control

This Recommendation separates the treatment of call and connection control, as call control is only needed at domain boundaries (e.g., UNI, E-NNI). Thus, within a domain (i.e., I-NNI) it is only necessary to support procedures for connection control. Additionally, call control is provided at interlayer 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.

### 5.1.1 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.

### 5.1.2 Call admission control

Call admission control is a policy function 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 user name 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.

### 5.1.3 Connection control

Connection control is responsible for the overall control of individual connections. Connection control may also be considered to be associated with link control. 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.

### 5.1.4 Connection admission control

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.

### 5.1.5 Relationship between call state and connection state

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.

## 5.2 Interaction between control, transport and management planes

Figure 5.1 illustrates the general relationships between the control, management and transport planes. Each plane is autonomous, but some interaction will occur. The following provides further details on the interactions between the various planes.

### 5.2.1 Management – Transport interaction

The management plane interacts 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 and the control component.

### 5.2.2 Control – 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 plane view of link connections. The TAP hides the interaction with the hardware.

### 5.2.3 Management – Control interaction

Clause 7.1 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. This is discussed further in clause 7.1.

The management plane interacts 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.

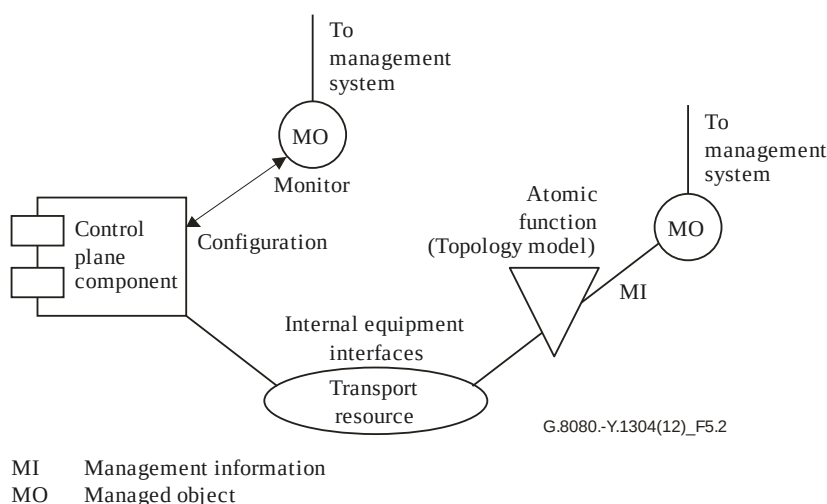


**Figure 5.2 – Management/transport plane interactions with transport resources**

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 control plane view, control plane components operate directly on the transport resources, so control plane operation appears autonomous to the management plane. Likewise, management plane operations appear autonomous to the control plane. 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 control plane is similar to that presented to management, it is not identical to the MI information. Control plane information overlaps the MI data because the control plane requires some but not all management 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 plane 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.

#### **5.2.4 Resource management**

Network resources may be partitioned between those under the authority of the management plane and those under the authority of the control plane. It shall not be possible for the control plane to modify resources that are under the authority of the management plane. This includes network resources not currently in use, but reserved for future use (e.g., by network planners). As such, resource management is performed by the management plane and is outside the scope of this Recommendation.

## **6 Transport resources and their organization**

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. For the purposes of control and management, each transport resource has a closely coupled agent that represents the role it has to play. These agents interact with other functions that are participating in the control and management through interfaces, and present information or execute operations as required. The transport resources are organized into routing areas and subnetworks for the purposes of control and management.

### **6.1 Transport entities**

For the purpose of managing connections within a layer network, the underlying transport plane resources are represented by a number of entities in the control plane. Figure 6.1 illustrates the relationship between the transport resources described in [ITU-T G.805], the entities that represent these resources for the purposes of network management (as described in [ITU-T M.3100]) and the view of the transport resources as seen by the control plane.

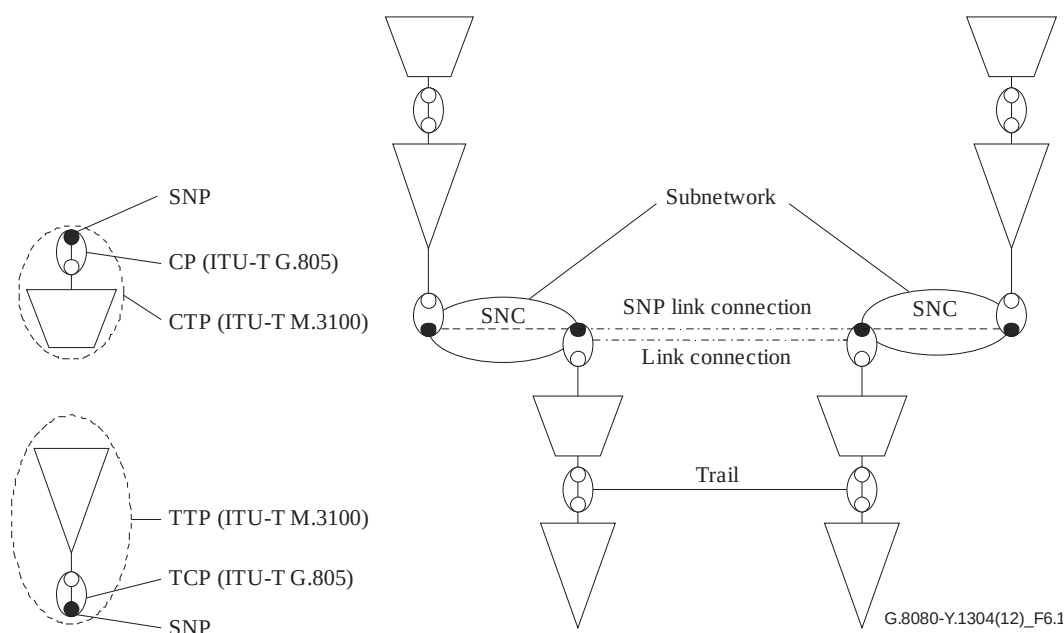


**Figure 6.1 – Relationship between architectural entities in the transport plane, management plane and the control plane**

An SNP has a number of relationships with other SNPs:

- A static relationship between two SNPs in different subnetworks. This is referred to as an SNP link connection.
- A dynamic relationship between two (or more in the case of broadcast connections) subnetwork points at the boundary of the same subnetwork. This is referred to as a subnetwork connection.

A subnetwork point may also be grouped with other SNPs for the purpose of routing. This is a subnetwork point pool (SNPP) and has a strong relationship with link ends (as defined in [b-ITU-T G.852.2]); however, this relationship is more flexible than the link end. An SNPP may be further subdivided into smaller pools. One use of this sub-structuring is to describe different degrees of route diversity. For example, all the SNPs in one subnetwork that have a relationship to a similar group on another subnetwork may be grouped into a single SNPP. This SNPP may be further sub-divided to represent diverse routes and further subdivided to represent, for example, individual wavelengths.

The association between SNPPs on different subnetworks is an SNPP link.

An SNPP link where each subnetwork is in a different layer is known as a transitional SNPP link. It may also be an SNPP link in which the subnetworks are in different sublayers of the same layer. They only occur across boundaries between layers or sublayers where [ITU-T G.800] transitional links can exist.

The SNP and SNP link connection states of interest to the control plane are described in clause 7.3.8 (Termination and adaptation performers) and clause 7.3.3 (Link resource manager) respectively.

### 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 GFP mappings for packet clients or different multiplexing structures for SDH/OTN. The description below illustrates the application to the latter.

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

Figure 6.2 shows an example of an STM-1 trail that can support either a single VC-4 or three VC-3s.

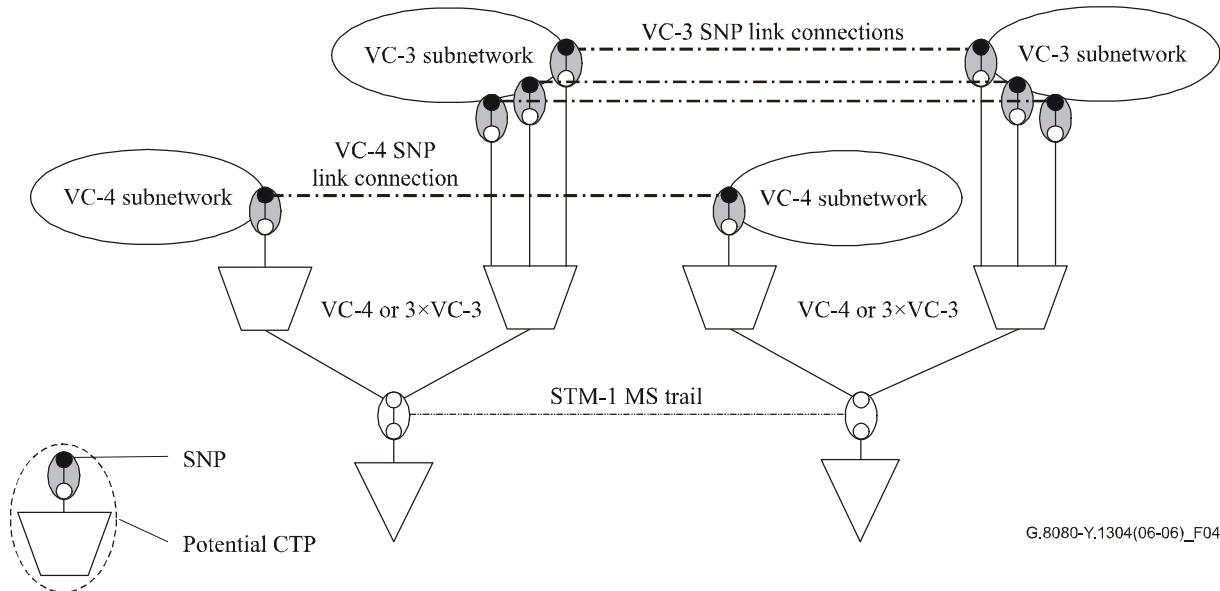


**Figure 6.2 – Example of variable adaptation  
(STM-1 trail supporting both  $3 \times \text{VC-3}$  or  $1 \times \text{VC-4}$ )**

### Link resources shared between VPNs

[ITU-T Y.1313] defines several basic service models through which layer one VPNs (L1VPNs) may be provided through the ASON architecture.

A VPN is a closed user group that can use a defined set of network resources. In the control plane, an SNPP can be public, that is, not associated with any VPN, or private, that is, associated with exactly one VPN. Connection routing in a VPN can only use the SNPPs associated with that VPN. In the transport plane, a CP can be assigned to an SNP in multiple SNPPs, public or private. Connectivity on a link that is shared between VPNs can be modelled by creating an SNP for each of the shared CPs in each VPN. When a CP is allocated to a particular SNP in one VPN, the SNPs representing the same resources in other VPNs become busy. Figure 6.3 shows an example of two VPNs, each with two SNPs in the control plane. In the transport plane, the first CP is assigned and allocated to the second SNP in VPN 2, the third CP is assigned and allocated to the second SNP in VPN 1, and the second CP is assigned to both the first SNP in VPN 1 and the first SNP in VPN 2. If the second CP is allocated to the first SNP in VPN 1, this SNP becomes available while the first SNP in VPN 2 becomes busy.

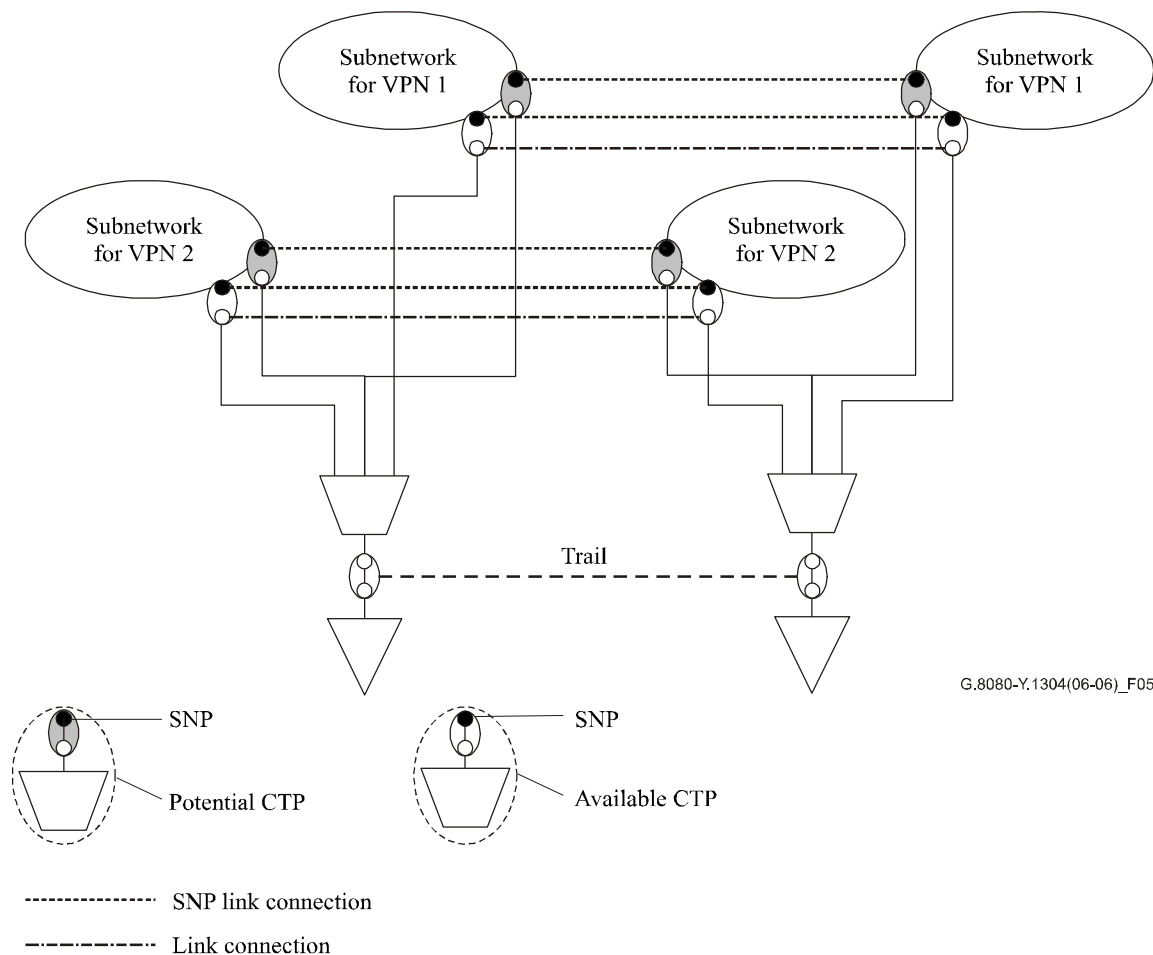


**Figure 6.3 – Allocation of link resources between VPNs**

In the case of circuit switching, the SNP is bound to a resource label, which provides a resource reservation and allocation. In the case of packet switching, the SNP is not directly bound to a resource label, and the resource label does not carry either any resource allocation. Therefore, in the case of packet switching, when a connection is established, an SNP is selected from a range of SNPs that is bound to a resource label. The connection request should include a resource reservation (CIR and EIR).

## 6.2 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 6.4.

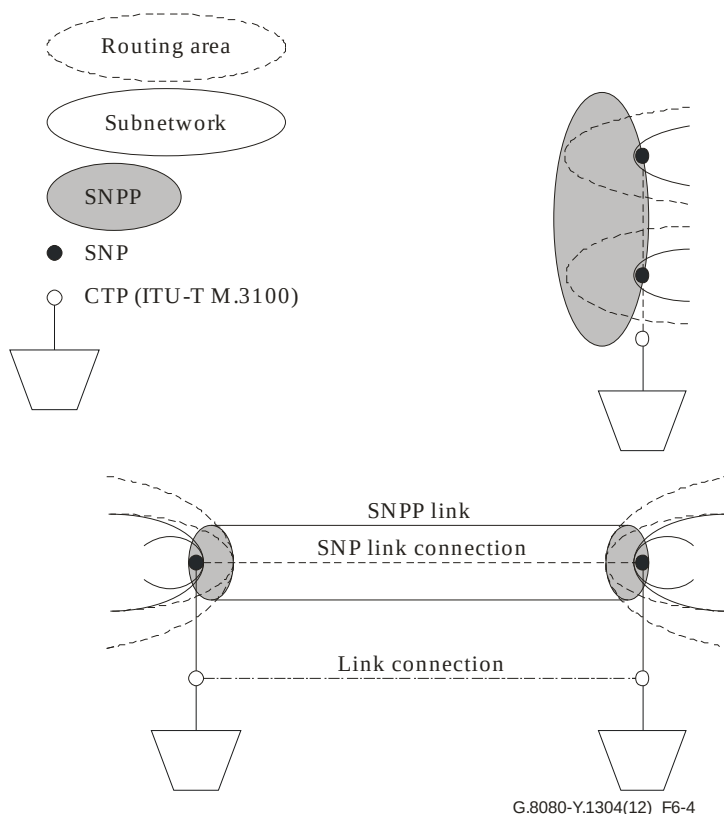


**Figure 6.4 – Relationship between routing areas, subnetworks, SNPs and SNPP**

### 6.2.1 Aggregation of links and routing areas

Figure 6.5 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.

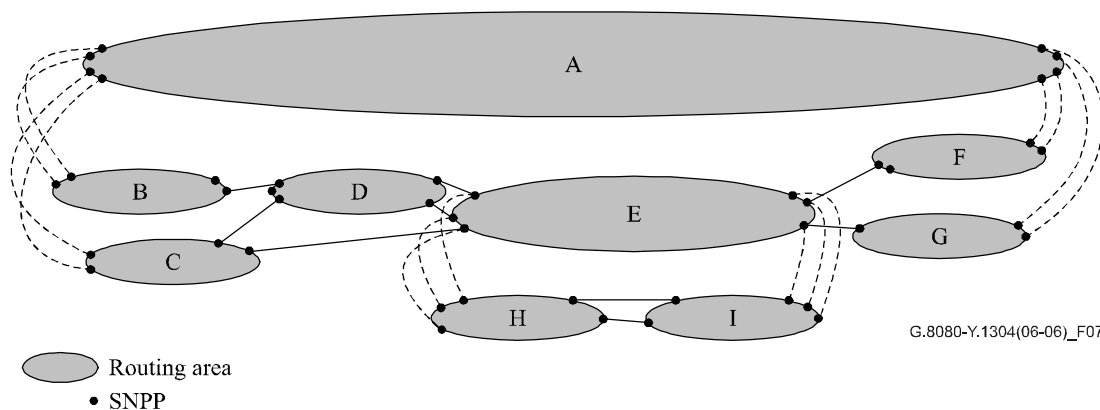


**Figure 6.5 – Example of a routing area hierarchy and SNPP link relationships**

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.

### 6.2.2 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 6.6. 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.

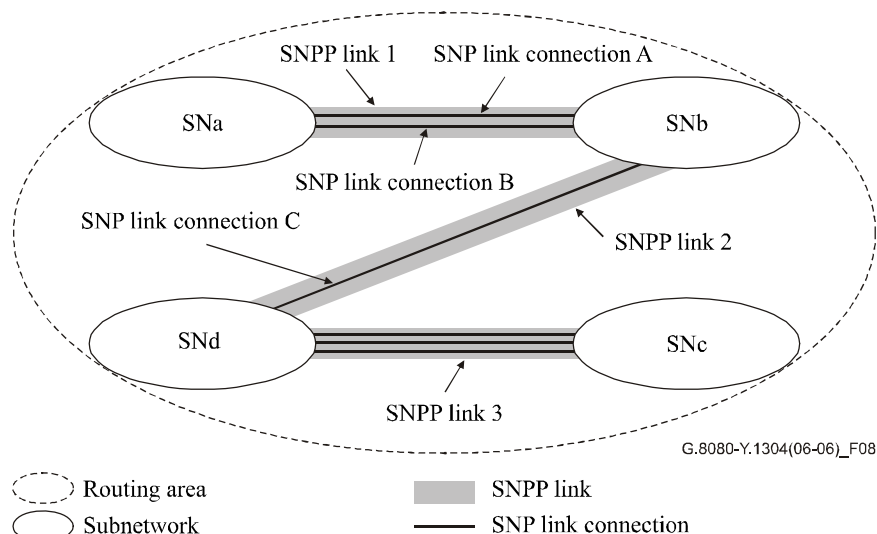


**Figure 6.6 – SNPP link relationship to subnetworks**

Figure 6.7 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).

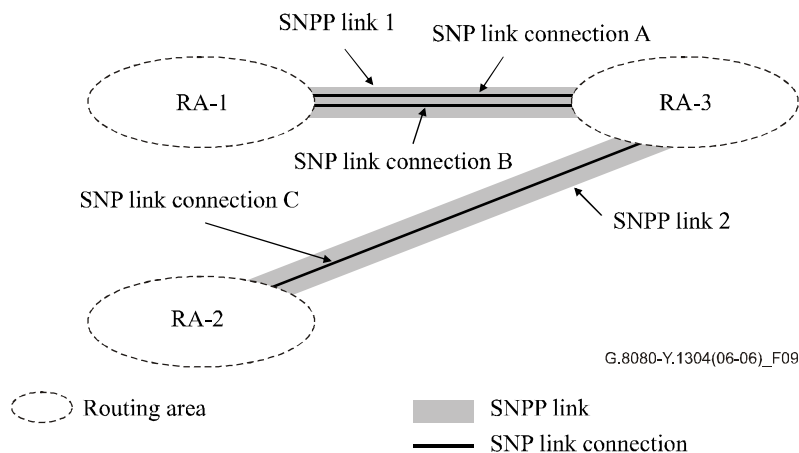


**Figure 6.7 – SNPP link relationships to routing areas**

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 6.8, 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 6.8. 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-2.

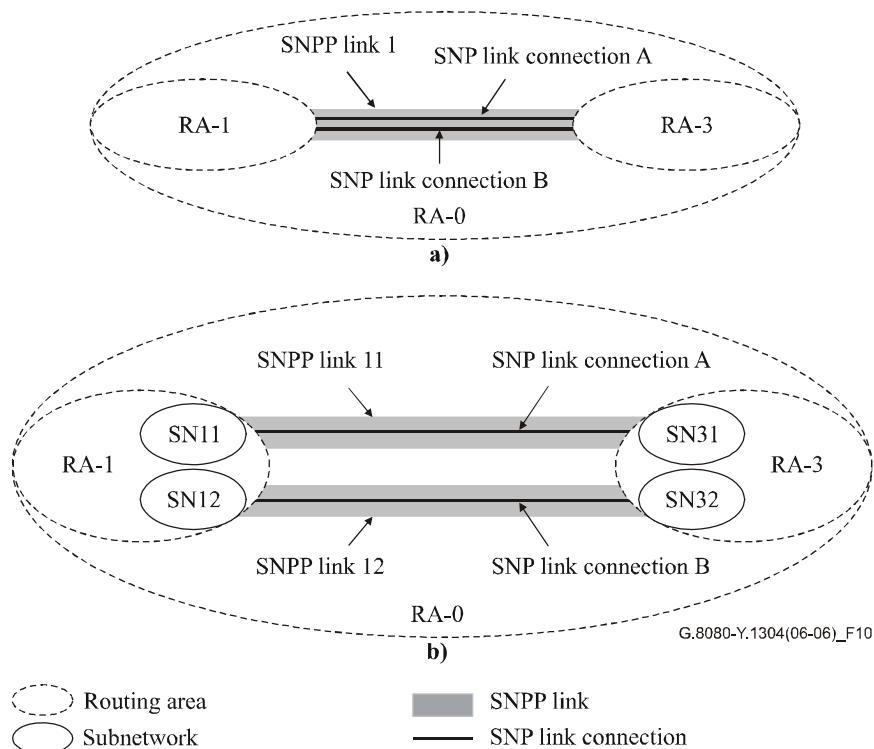


**Figure 6.8 – Routing scope**

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 6.8, SNPP link 11, link 12 and link 1 can all coexist.

Another reason for choosing SNPP link 11 (Figure 6.8-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.

### 6.3 Topology and discovery

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 the management plane.

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 the management plane.

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 7.3.3) 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 plane 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 and the local and interface SNP ids may be identical. Otherwise, at each end of the link the interface SNPP id is mapped to a local SNPP id and the interface SNP ids are mapped to local SNP ids. This is shown in Figure 6.9.

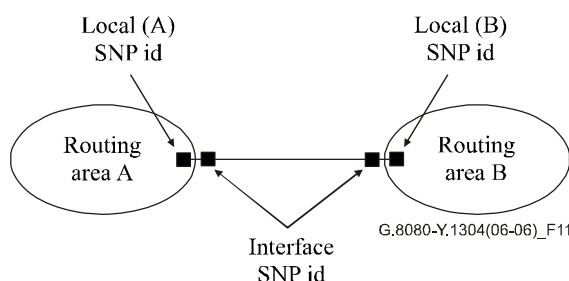


**Figure 6.9 – Relationship between local and interface ids**

The resulting SNPP 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 plane or management plane 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 the management plane or the control plane. If the control plane 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 7.3.2) of the SNPP link adjacency and the link characteristics, e.g., cost, performance, quality diversity, and bandwidth.

## 6.4 Domains

As introduced in clause 5, we have generalized the domain notion embodied in the ITU-T G.805 definition of administrative and management domains, 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.

A control domain is comprised of a collection of control plane components, and provides an architectural construct that encapsulates and hides the detail of a distributed implementation of a particular group of architectural component of one or more types. It allows for the description of a group of distributed components in such a way that the group can be represented by distribution interfaces on a single entity, the domain, that has identical characteristics to that of the interfaces of the original component distribution interfaces. The nature of the information exchanged between control domains captures the common semantics of the information exchanged between component distribution interfaces, while allowing for different representations inside the domain.

Generally a control domain is derived from a particular component type, or types, that interact for a particular purpose. For example, routing (control) domains are derived from routing controller components whilst a re-routing domain is derived from a set of connection controller and network call controller components that share responsibility for the re-routing/restoration of connections/calls that traverse that domain. In both examples the operation that occurs, routing or re-routing, is contained entirely within the domain. In this Recommendation, control domains are described in relation to components associated with a layer network.

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. Domains of the same type are restricted in that they may:

- fully contain other domains of the same type, but do not overlap
- border each other
- be isolated from each other.

An example of the relationships between components, domains and reference points is provided in Figure 6.10 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.

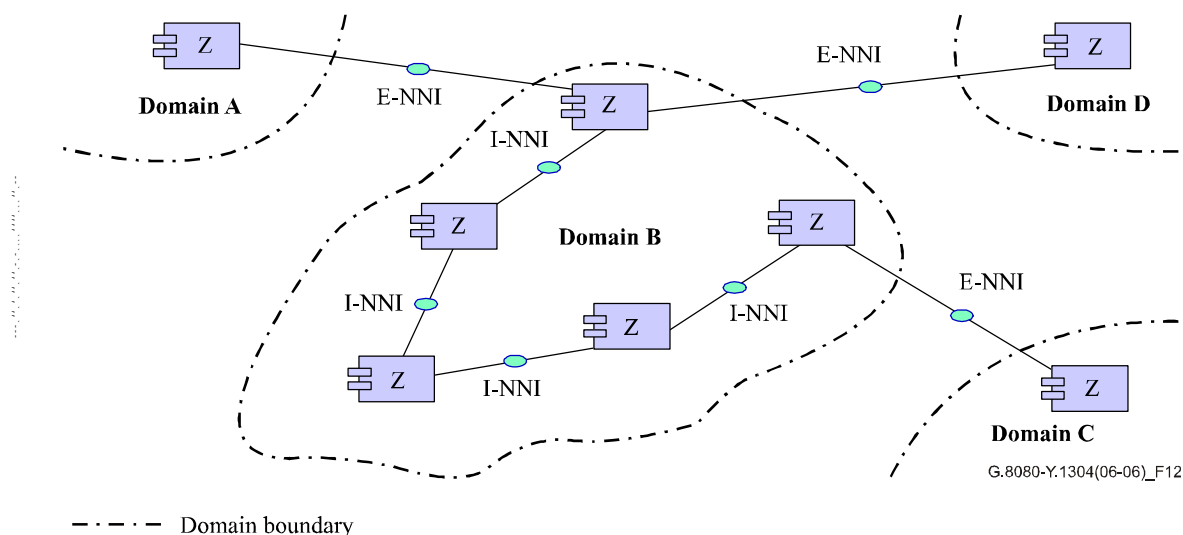


**Figure 6.10 – Relationship between domains, protocol controllers and reference points**

The same example is shown in Figure 6.11 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 to that used in B, and the protocol used between B and C to be different to 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.

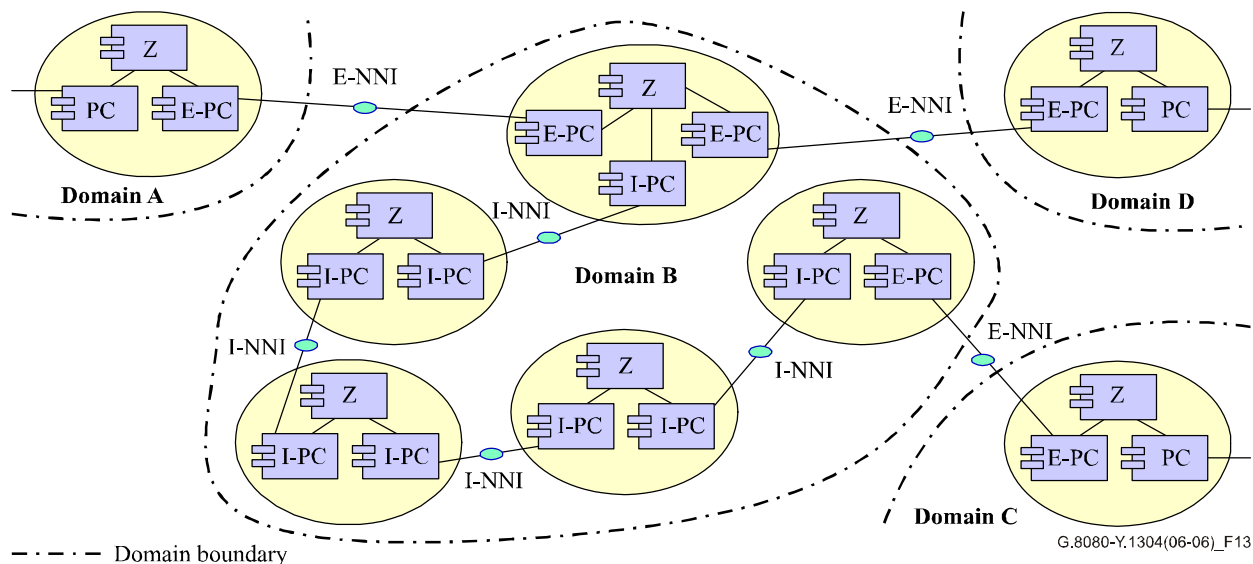


**Figure 6.11 – Relationship between domains, protocol controllers and interfaces**

#### 6.4.1 Relationship between control domains and control plane resources

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.

#### 6.4.2 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.

### 6.5 Multi-layer aspects

The description of the control plane can be divided into those aspects related to a single layer network, such as routing, creation and deletion of connections, etc., and those that relate to multiple layers. The relationship between the client and server layer networks is managed by means of the termination and adaptation performers (see clause 7.3.8). The topology and connectivity of the underlying server layer is not explicitly visible to the client layer, rather aspects of the server layer are encapsulated and presented to the client layer network. The server layer may be represented as a client layer SNPP link, or as client layer subnetworks interconnected by SNPP links. This abstraction of the server layer topology for the purpose of path computation in the client layer has similar properties to the hierarchical containment relationships described in clause 6.2.1. Thus the results of path computation in the client layer may not result in the optimum use of the server layer resources. If the resources available to the client layer network are insufficient to support a connection request, additional resources may be provided by activating or creating new connections in one or more server layer networks. Operator policies will govern the availability of underlying server layer resources to the client layer.

When the server layer is represented to the client layer as an SNPP link, this can be achieved by modifying SNPs from potential to available. If the server layer is represented as a set of SNPP links and subnetworks, then a new server layer connection must be requested directly by the control plane

depending on operator policies. In the case where the capacity of the resulting server layer network connection is greater than the capacity required to support the client layer connection, a new SNPP link will be created as a side effect of this operation. The mechanism to add this new SNPP link to the topology of the client layer is not within the scope of this Recommendation.

Alternatively, the server layer topology may not be made visible to the client layer. In this case the decision to add new server layer network connections may be made by a planning process. The use of a planning system to invoke the configuration of the server layer network to provide additional resources for the client layer network may result in more optimum use of the server layer resources. Details of the planning process and its interaction with the client and the server control planes is outside the scope of this Recommendation. The decision to create/remove the server layer connections is a business decision that may be represented in operator policy.

As described above, there are two different ways to represent server layer resources in the client layer. The representation used is dependent on whether server layer resources have already been selected to provide connectivity to the client layer. A control plane instance for a layer network may use one or both of these approaches at a time. Other representations of server layer resources are for further study.

### **6.5.1 Representation as SNP link connections**

A planning process may cause a pair of access points in the server layer to be connected, this creates a client layer SNPP link that contains multiple SNP link connections for the supported client layers. These SNP link connections may initially be either active or potential, as the underlying resource may or may not be allocated for the exclusive use of the client layer (see clause 6.1).

This process may be applied recursively in multiple client/server layer networks. Thus, the connection between the pair of access points may be supported by potential link connections. The connection of the access points in the server layer is not activated until a request to use one of the potential client layer SNP link connections is received by the control plane. At this time, the client layer request is "suspended" while the server layer connections are established. Once the server layer connections are in place, the client layer SNP link connections are made active, allowing the client layer signalling to resume. If a connection in the server layer cannot be established, the client layer SNP link connections cannot be made active, causing the client layer connection attempt to fail due to lack of resources.

When the SNP link connection is removed, the underlying resources that supported the connection are freed, allowing them to return to the potential state if allowed by policy. When all SNP link connections supported by a server trail return to the potential state, the server trail may be changed to potential if allowed by policy.

It should be noted that the SNPP link containing these SNP link connections can be represented using existing components and states as described in clause 6.1.

### **6.5.2 Representation as a set of SNPP links and subnetworks**

A planning process may also allow the client layer control plane more flexibility in the selection of the server layer resources to satisfy a connection request. To accomplish this, some of the server layer resources are represented to the client layer as SNPP links and subnetworks with the appropriate client layer control plane components. This representation is possible if the server layer provides flexibility which allows client layer subnetworks to be interconnected. The server layer flexibility and its corresponding client layer flexibility are shown in Figure 6.12. This enables the routing controller to know that a set of client layer SNPs can be reachable through a common server layer. That is, in Figure 6.12, absence of the server layer representation would not allow paths to be computed between all of the client layer SNPs.

The adaptation and termination functions used to transition from the client layer to the server layer are represented as SNPP links in the client layer. In this figure, we used dashed elements to distinguish this representation of the server layer resources to the client layer from other client layer resources. Additionally, client and server layer SNPs are represented as small solid-line white circles, respectively; however, to the client layer routing controller, there is no distinction between the represented elements and the other client layer elements.

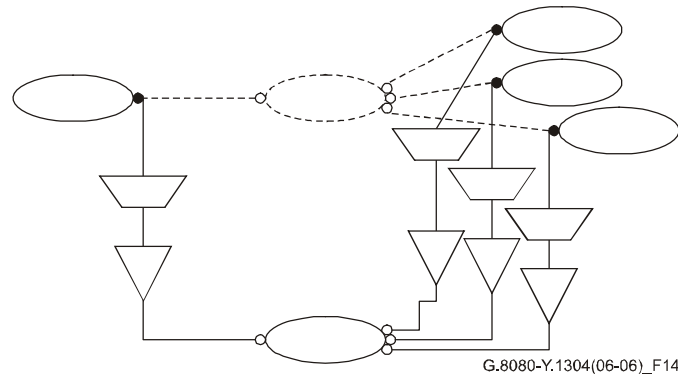


**Figure 6.12 – Relationship between client and server architectural elements**

As in the SNPP link representation, the SNP link connections connecting the client layer subnetwork to the server layer are potential, as the server layer resources have not been allocated for exclusive use to the client layer. When a connection request is received that utilizes a potential resource, a path through the server layer is calculated identifying the specific resources to be used. As a result of identifying these resources, client layer SNP link connections are created, and processing continues as above.

### 6.5.3 Multilayer routing topology

In addition to the single layer topology representation in clause 6.5.2, 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 plane 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 6.13 illustrates the [ITU-T G.800] representation and the corresponding multilayer routing topologies.

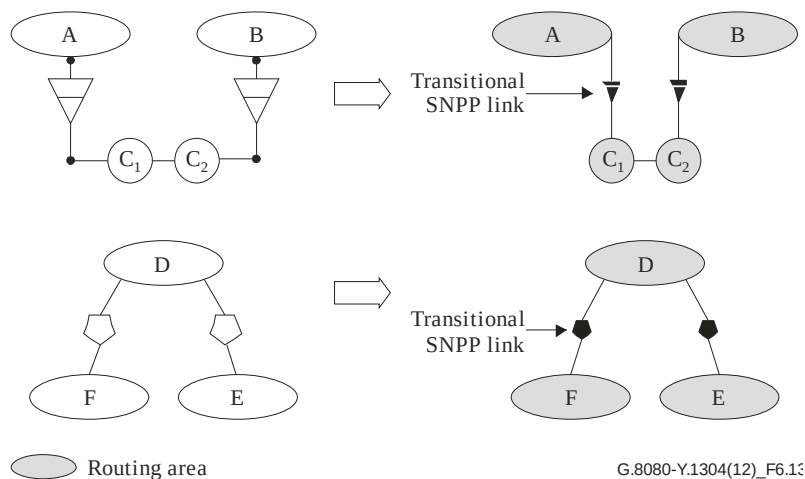


**Figure 6.13 – Multilayer routing topology representations**

A multilayer topology corresponding to the model in Figure 6.12 is illustrated in Figure 6.14 on the left. It differs from the model in Figure 6.12 in that the topology is multilayer (i.e., contains both client and server SNPPs).

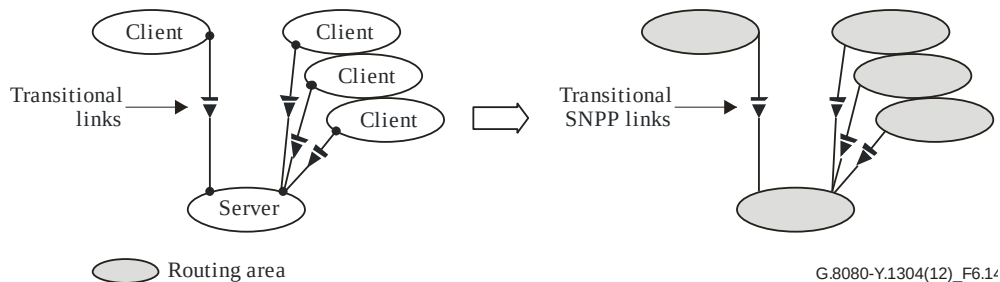


**Figure 6.14 – Multilayer routing topology representation of Figure 6.12**

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 multilayer 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 multilayer topology is shown in Appendix V.

NOTE – Use of transitional links for a 1:1 adaptation is described in this section. Usage for 1:n and m:1 is for further study.

## 6.6 Interlayer client support

In transport networks, network elements may support more than a single layer. For example at the edge of a multi-layer transport network, a network element may support client layer networks that are directly supported in the core of a multilayer transport network. For example, the edge network element may support the adaptation of service into a lower order VC whilst the core network

elements may only provide flexibility at the high order VC layer; or the edge network element may adapt an Ethernet service into high order VCs. A general problem faced is how to transfer client characteristic information (CI) when a continuous/connected client layer network is not present between two client AGCs.

There are two solutions to this problem. Either, client layer topology may be created from server layer connections as described in clause 6.5 or, the client CI could be adapted, possibly multiple times, onto server layer connections. This would not be visible to the client routing controller.

Interfaces between network call controllers (NCCs, see clause 7.3.5.2) in different layer networks are used to apply ASON functions to the second solution. This interlayer interface enables an association between calls in a client/server layer relationship. This association can recurse to mirror a set of "stacked" adaptations. That is, the NCCs recurse with ITU-T G.805 layers. NCCs at different layers may still be instantiated differently from each other. For example an NCC could be distributed at a client layer and centralized at a server layer. A server layer CC creates the connection(s). The client CI is mapped to the server layer connection and this association is maintained by the client/server NCC relationship. In this situation, a client layer link connection is created as a result of the server layer connection and CI mapping, but the client layer CC is not involved in this. This recurses upward and creates a link connection at each of the affected client layers.

Appendix II illustrates this capability with an example.

The interlayer NCC relationship may occur at points other than where access group containers are attached to the client layer network. In Figure 6.12, a call traverses a client subnetwork first before being supported by a server layer subnetwork. In Figure 6.15, the call may also be supported by connections in a client layer subnetwork that is not contiguous with client layer subnetworks at the ingress or egress. Here, interlayer NCCs relationships are found between client subnetwork C2 and server subnetworks S1 and S2.

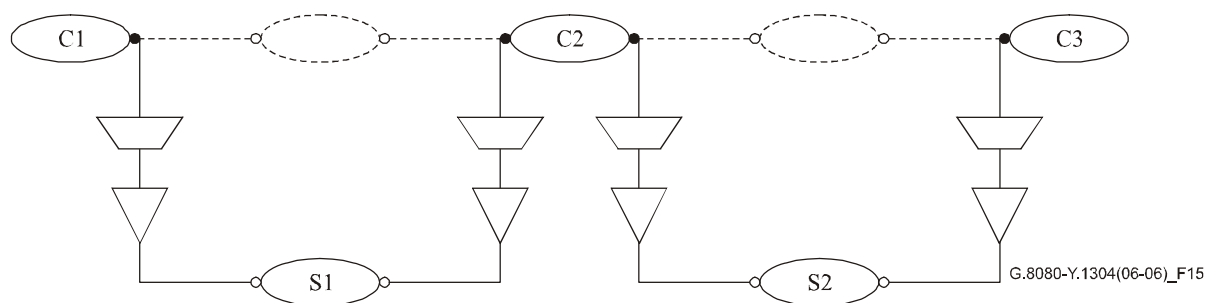


**Figure 6.15 – Non-contiguous client layer subnetworks**

## 6.7 Calls supported by calls at same layer

Similar to the arrangement of NCCs in an interlayer relationship, a call segment may be supported by a separate call at the same layer but not over an E-NNI. In this arrangement, an NCC to NCC call segment is supported by a complete call with calling/called party call controllers. Figure 6.16 illustrates this in a carrier's carrier business scenario. Here, a call between two clients associated with Carrier A is supported in two subnetworks that belong to Carrier A. Between those two subnetworks, the call is supported by a separate SPC within Carrier B. The connection returned by Carrier B is joined with the connections established in the Carrier A subnetworks.

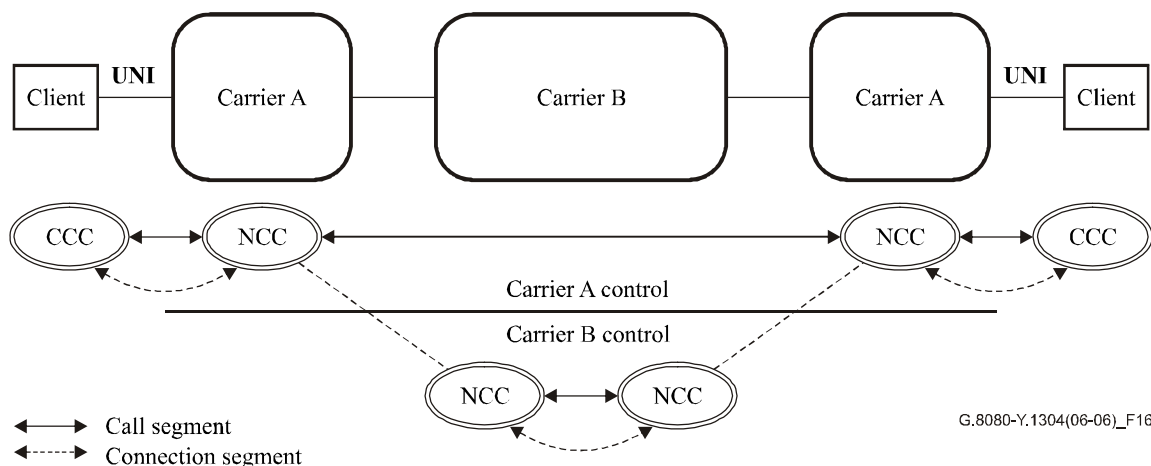


Figure 6.16 – Calls supported by calls in the same layer

## 6.8 Mapped server interlayer relationships

When a client layer CI is mapped to a server layer as described in clauses 6.5 and 6.6, several types of arrangements can exist. They include client/servers in 1:1 and 1:n relationships. The ratio refers to the number of connection points in each layer.

### 6.8.1 1:1 relationship

In the 1:1 relationship support for a client communication is supported by the server layer as a single trail. The  $NCC_{client}:NCC_{server}$  relationship is also 1:1. A specific example of this is shown in Appendix II where Ethernet CI is mapped to a single VC-3. Another example is a DS-3 into an STS-1.

As the  $NCC_{client}$  is trying to use the server layer, it must have knowledge of the relevant call parameters of the  $NCC_{server}$  including whether the client initiated the server layer call (server NCC coordination out interface) or whether the server layer already existed for client layer use (client NCC coordination out interface).

The  $NCC_{server}$  is not required to possess knowledge of the client layer call parameters, but should inform the  $NCC_{client}$  if there are changes in the server layer call.

### 6.8.2 1:n relationship

In a 1:n relationship, the client communication is supported by multiple connections in the server layer. This is supported either by the server NCC supporting multiple connections or multiple NCCs in the server layer supporting the one client layer NCC.

The example below in Figure 6.17 illustrates the latter case. The Ethernet call is mapped to a VC-4-2v VCAT call. The VCAT call is related to multiple server layer VC-4 calls. The Ethernet/GFP layer to VCAT relationship is 1:1.

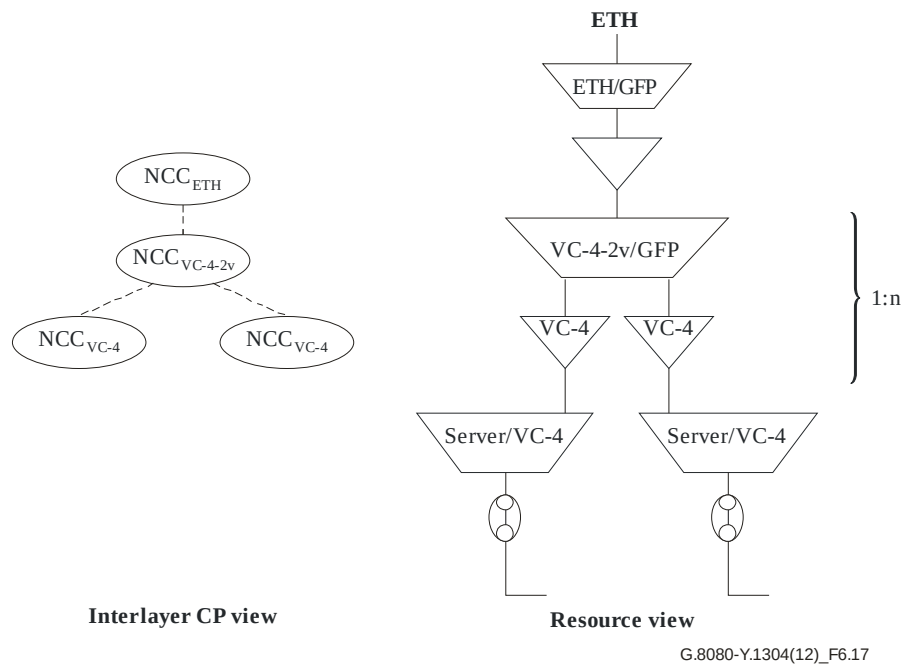


**Figure 6.17 – Example 1:n mapped server**

It is necessary for the client layer NCC (the VC-4-2v layer) to know about the call parameters of the server layer as it must ensure that there are a sufficient number of server layer calls as well as what their aggregate characteristics are. The server layer is not required to possess knowledge of the client layer call parameters, but should inform the client call if there are changes in the server layer call.

## 7 Control plane architecture

This clause describes a reference architecture for the control plane 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 control plane architecture should have the following characteristics:

- Support various transport infrastructures, such as those covered by [ITU-T G.805].
- 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 control plane has been 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 the control plane, 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.

Special components are defined in this Recommendation and are provided to allow for implementation flexibility. These components are protocol controllers and port controllers. 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. This is described in clause 7.4 and illustrated in Figure 7.28. 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.

Port controllers are provided to apply rules to system interfaces. Their purpose is 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. This is described in clause 7.2.1 and Figure 7.2.

## 7.1 Notation

In this clause, we consider the component architectural notation based upon some simple building blocks from the vocabulary of the unified modelling language (UML).

**Interface:** An interface supports a collection of operations that specify a service of a component, and is specified independently from the components that use or provide that service. Operations specify the information passed in or out together with any applicable constraints. Interface definitions are presented in the form of a table, an example of which is presented in Table 1. Each interface has an interface name that identifies the role. Input interfaces represent services provided by the component; the basic input parameters are required for the specific role and basic return parameters are a result of the action on the input parameters. Output interfaces represent services used by the component; the basic output parameters define the information provided, the basic return parameters (if identified) are those required in response to the output parameters. Notification interfaces represent unsolicited output actions by the component, and are represented by an output interface with no return parameters. These three interface types are described separately in interface specifications.

**Table 1 – Generic interface descriptions table format**

| Input interface | Basic input parameters | Basic return parameters |
|-----------------|------------------------|-------------------------|
| Interface name  | Input parameters       | Returned parameters     |

| Output interface | Basic output parameters | Basic return parameters |
|------------------|-------------------------|-------------------------|
| Interface name   | Output parameters       | Returned parameters     |

Transaction semantics associated with a particular transaction are assumed to be handled transparently, and there is no need to explicitly mention separate parameters for this purpose in interface description.

**Role:** A role is the behaviour of an entity when it is participating in a particular context. Roles allow for the possibility that different entities participate at different times, and are denoted by annotating a relationship with the name of an interface.

**Component:** In this Recommendation, components are used to represent abstract entities, rather than instances of implementation code. They are used to construct scenarios to explain the operation of the architecture. This component is represented as a rectangle with tabs. This is illustrated in Figure 7.1.

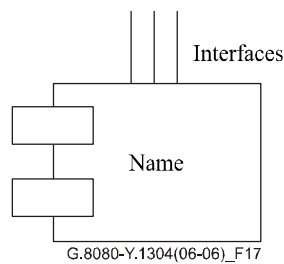


**Figure 7.1 – Representation of a component**

Generically, every 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 not mandatory, and are provided on specific components only where necessary. Where appropriate, the use of the monitor interface is described in individual component descriptions. Components are not assumed to be statically distributed.

When interfaces on components are described, only the different interface types are specified. All components have the property of supporting multiple callers and multiple providers, and resolution of concurrent requests is not mentioned explicitly.

As components are used in an abstract way, this specification is extendable by the techniques of component sub-classing and composition.

## **7.2 Policy and federations**

### **7.2.1 General model of policy**

For the purposes of this policy model, systems represent collections of components, and a system boundary provides a point where policy may be applied. Policy is defined as the set of rules applied to interfaces at the system boundary, and implemented by port controller components. Policy ports are used to simplify the modelling of policies that are applied to multiple ports. System boundaries are nested to allow for correct modelling of shared policies applied to any scope (full system, any set of components, individual components, etc.). Note that the order of policy application is that which is specified by the nesting.

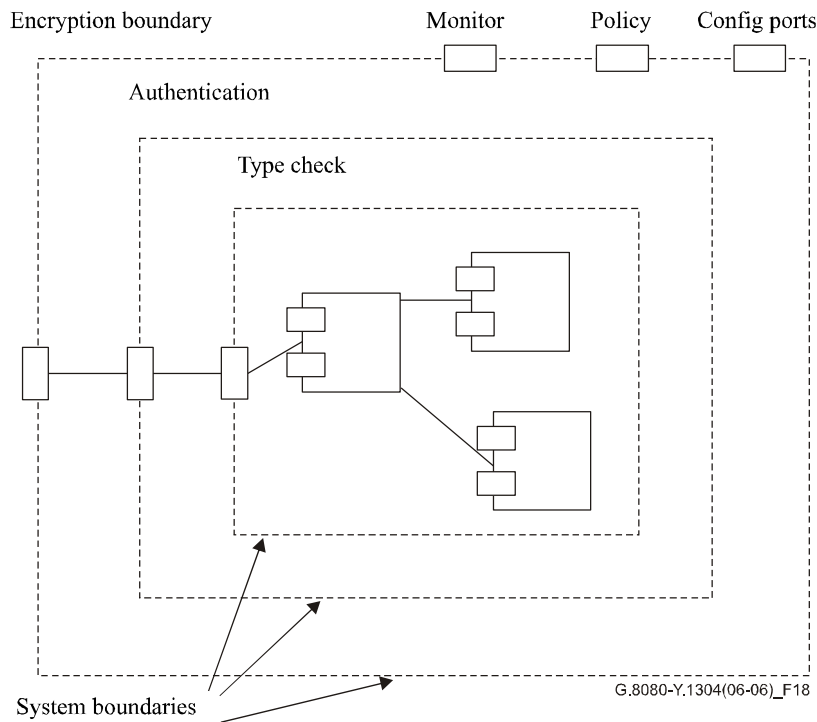


**Figure 7.2 – Example of system boundaries related to policy control**

In Figure 7.2, the dashed boxes represent system boundaries, while the closed rectangles on the boundary, called ports, represent port controller components.

The monitor, policy, and configuration ports may be available on every system (and component) without further architectural specification. The monitor port allows management information to pass through the boundary relating to performance degradations, trouble events, failures, etc., for components, subject to policy constraints. The policy port allows for the exchange of policy information relating to components. The configuration port allows for the exchange of configuration, provisioning and administration information relating to components (subject to policy constraints) that may dynamically adjust the internal behaviour of the system.

Figure 7.2 shows an example of how encryption, authentication and type checking may be implemented as a set of three nested port controllers, where the policy application order follows the nesting order. The components inside the authentication boundary do not specify encryption or authentication requirements, as these are properties of the component environment. Port controllers are defined for each independent aspect of port policy, and combined policy is achieved by composition of port controllers. This allows the creation of reusable components, which are distinguished by a descriptive prefix. Policy violations are reported via the monitoring port.

The policy port may be seen as a filter of incoming messages, where messages that are rejected have violated the policy. Policies may be dynamically changed via the system policy port, and in this way, dynamic behavioural changes may be described.

It is common to discuss how policy is applied at a reference point, but policy can only be applied to the individual interfaces crossing the reference point. A method of combining several interfaces into a single implementation interface is described later in clause 7.4, protocol controllers.

Other aspects of policy relate to variable behaviour of the components (such as schedules, access rights, etc.) and these aspects are specified and implemented by the components. Component behaviour may also be dynamically changed, and the ability to do this may be controlled by policy. This allows us to determine which aspects of system behaviour are specified where.

Policy, as other aspects of the system, may be distributed. An example of a suitable model for distribution could be the COPS protocol model of [b-IETF RFC 2753]. The policy enforcement point (PEP) (the point where the policy decisions are enforced) of that model corresponds to the port in this model. The policy decision point (PDP) is the point where policy decisions are made. This can be done within the port, though it may be distributed to a different system. This distribution decision depends on many factors that in turn depend on the actual policy. As an example, performance reasons may force the PDP to be within the port (encryption), while security reasons may force the PDP to be elsewhere (password lookup).

When the PEP and PDP are not collocated, cooperation is required.

### 7.2.2 General model of federation

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 7.3.1.)

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 7.3. This model is recursive with a parent connection controller at one level being a child to a parent at a higher level.

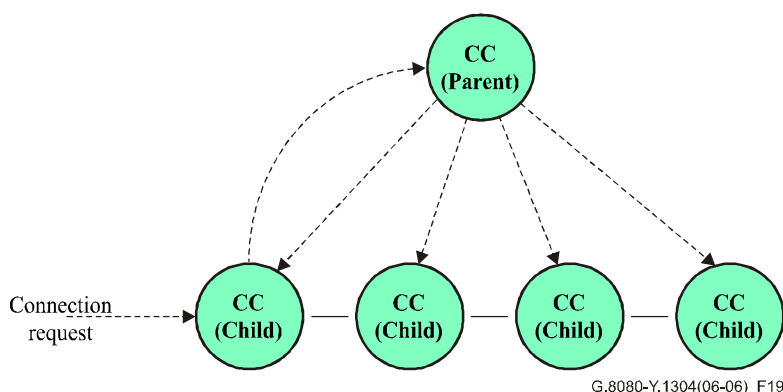


**Figure 7.3 – Joint federation model**

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 7.4, 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.

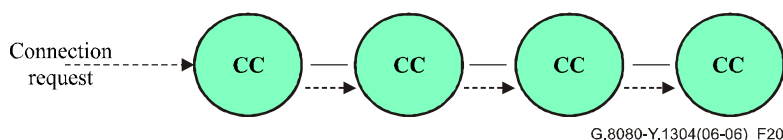


**Figure 7.4 – Cooperative federation model**

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 7.5. The principle described above can also be applied to federations of call controllers.

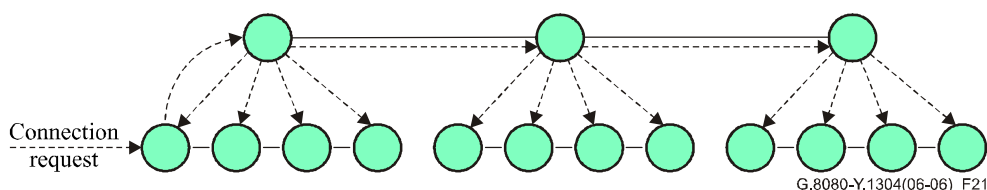


**Figure 7.5 – Combined federation model**

### 7.3 Architectural components

The components of the control plane architecture are described in this clause. Components can be combined in different ways, depending upon the required functionality. Appendix V 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 control plane 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 control plane component is provided by the protocol controller associated with that component.

#### 7.3.1 Connection controller (CC) component

The connection controller is responsible for coordination among the link resource manager, routing controller, and both peer and subordinate connection controllers for the purpose of the management and supervision of connection set-ups, releases and the modification of connection parameters for existing connections. This component services a single subnetwork, and provides the abstract interfaces to other control plane components given in Table 2. The connection controller component is illustrated in Figure 7.6.

NOTE – The route query interface does not apply for the CC interface at the UNI reference point.

In addition, the CC component provides a connection controller interface (CCI). This is an interface between a subnetwork in the transport plane and the control plane. It is used by control components to direct the creation, modification, and deletion of SNCs. Policy is not applied to the CCI.

**Table 2 – Connection controller component interfaces**

| Input interface       | Basic input parameters                                                             | Basic return parameters |
|-----------------------|------------------------------------------------------------------------------------|-------------------------|
| Connection request in | A pair of local SNP names and optionally a route                                   | A subnetwork connection |
| Peer coordination in  | 1) A pair of SNP names; or<br>2) SNP and SNPP; or<br>3) SNPP pair; or<br>4) route. | Confirmation signal     |

| Output interface           | Basic output parameters                                                            | Basic return parameters         |
|----------------------------|------------------------------------------------------------------------------------|---------------------------------|
| Route query                | Unresolved route fragment                                                          | Route                           |
| Link connection request    | –                                                                                  | A link connection (an SNP pair) |
| Connection request out     | A pair of local SNP names                                                          | A subnetwork connection         |
| Peer coordination out      | 1) A pair of SNP names; or<br>2) SNP and SNPP; or<br>3) SNPP pair; or<br>4) route. | Confirmation signal             |
| Remote topology status out | Topology information (link and/or subnetwork) including resource availability      | –                               |

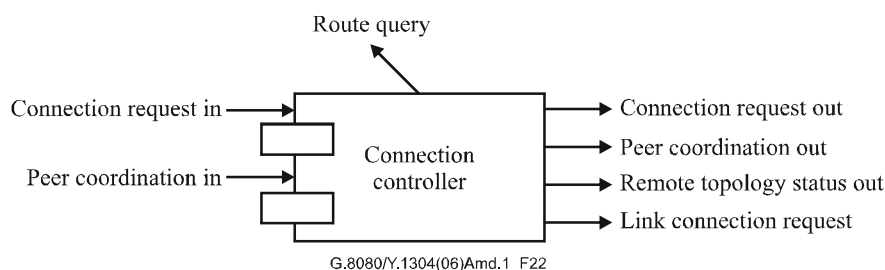


**Figure 7.6 – Connection controller component**

**Remote topology status out:** This interface is used to present topology status information learned by the connection controller. This status will include the ability to signal across a link.

### Connection set-up operation

Connection set-up is performed in response to either a connection request, from an enclosing scope connection controller, or from a peer connection controller. In the case of hierarchical routing where the superior (i.e., parent) CC selects the source and destination SNPs, the connection request in/out interface is used. In all other cases, the peer coordination in/out interfaces are used. Component operation is the same in both cases.

The first unresolved portion of the route is resolved, via the route table query interface, into a set of links to be traversed, and this new set of links adds to the set. The connection controller inspects the new set of links to see which of these links are available for link connection allocation. Link connections are obtained and their links are removed from the link set. Next, corresponding subnetwork connections are requested from subordinate (i.e., child) connection controllers via the connection request out interface. Any unallocated route components are passed on to the next downstream peer connection controller. The actual sequence of operations depends on many factors, including the amount of routing information available and the access to particular link