



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.1248.2

(07/2001)

SERIES Q: SWITCHING AND SIGNALLING

Intelligent Network

**Interface Recommendation for Intelligent
Network Capability Set 4: SCF-SSF interface**

ITU-T Recommendation Q.1248.2

ITU-T Q-SERIES RECOMMENDATIONS
SWITCHING AND SIGNALLING

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4 AND No. 5	Q.120–Q.249
SPECIFICATIONS OF SIGNALLING SYSTEM No. 6	Q.250–Q.309
SPECIFICATIONS OF SIGNALLING SYSTEM R1	Q.310–Q.399
SPECIFICATIONS OF SIGNALLING SYSTEM R2	Q.400–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.799
Q3 INTERFACE	Q.800–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
SPECIFICATIONS OF SIGNALLING RELATED TO BEARER INDEPENDENT CALL CONTROL (BICC)	Q.1900–Q.1999
BROADBAND ISDN	Q.2000–Q.2999

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

ITU-T Recommendation Q.1248.2

Interface Recommendation for Intelligent Network Capability Set 4: SCF-SSF interface

Summary

The Q.1248.x family of Recommendations defines the Intelligent Network (IN) Application Protocol for IN capability set 4 (IN CS-4). This family of Recommendations Q.1248.x defines the INAP for IN CS-4 based upon IN CS-2 ITU-T Q.1228 and Q.1224 (1997), and the general rules for INAP provided in ITU-T Q.1208, and is consistent with the scope of IN CS-4 as defined in ITU-T Q.1241.

Within the Q.124x series of Recommendations, ITU-T Q.1248.x family describes the protocol realizing the Q.1231 Distributed Functional Plane in a service and vendor implementation independent manner, as constrained by the capabilities of the embedded base of network technology.

This provides the flexibility to allocate distributed functionality into multiple physical network configurations and to evolve IN from IN CS-4 to some future CS-N.

This Recommendation belongs to the Q.1248.x family of Recommendations for IN capability set 4. It covers the SSF-SCF interface including the description of the aspects of the Functional Entities SSF and SCF which are relevant to this interface.

This Recommendation includes an electronic attachment containing Annex A, ASN.1 definitions.

Source

ITU-T Recommendation Q.1248.2 was prepared by ITU-T Study Group 11 (2001-2004) and approved under the WTSA Resolution 1 procedure on 13 July 2001.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. 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.

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, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2002

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

CONTENTS

	Page
1 Scope	1
2 References	1
3 Abbreviations and acronyms	1
4 Relationships	1
4.1 SSF-CCF relationship	1
4.2 SSF-SCF relationship	2
5 CCF Functional Model	2
5.1 Introduction	2
5.2 Basic Call Control Entity	3
5.2.1 Overview	3
5.2.2 BCSM Model	6
5.2.3 BCSM Resume Points and BCSM Transitions in the IN CS-4 Call Model ..	31
5.2.4 BCSM Indications for the CS-4 call model	41
5.2.5 Mapping from Cause to DP	48
5.2.6 Signalling Terminations	56
5.3 Bearer Control Entity	62
5.3.1 Overview	62
5.3.2 Bearer State Model	63
5.4 Functional Interface between the CCF and the SSF	65
5.4.1 Interactions from the CCF to the SSF	65
5.4.2 Interactions from the SSF to the CCF	65
6 SSF Function Model Components	66
6.1 DP Criteria	66
6.2 Trigger Types and Trigger Precedence	69
6.3 Feature interaction manager (FIM) call manager (CM)	69
6.4 IN-Switching Manager	70
6.4.1 IN-switching state model (IN-SSM)	70
6.4.2 The Connection View Model	74
6.4.3 Connection View State Transitions	78
6.5 Out-Channel Call-Related User Interaction (OCCRUI)	99
7 SCF Model	99
8 FSM for SSF	99
8.1 SSF management finite state model (SSME-FSM)	100
9 FSM for SCF	101

	Page
10	FSM for USI..... 101
11	Operation procedures 101
	11.1 Modified operations 101
	11.1.1 AnalyzeInformation 102
	11.1.2 CallGap..... 102
	11.1.3 Connect..... 105
	11.1.4 ConnectToRessouce 106
	11.1.5 ContinueWithArgument 106
	11.2 CreateOrRemoveTriggerData procedure 106
	11.2.1 General description 106
	11.2.2 Invoking entity (SCF)..... 107
	11.2.3 Responding entity (SSF) 107
	11.2.4 DisconnectLeg..... 108
	11.2.5 InitialDP 109
	11.2.6 InitiateCallAttempt..... 109
	11.2.7 ManageTriggerData 109
	11.2.8 MergeCallSegments 110
	11.2.9 MoveCallSegments 113
	11.2.10 MoveLeg 114
	11.2.11 RequestReportBCSMEvent..... 115
	11.2.12 SelectFacility 115
	11.2.13 SelectRoute..... 115
	11.2.14 SplitLeg 115
	11.3 New operations..... 116
	11.3.1 CallFiltering procedure 116
	11.3.2 MonitorRouteRequest procedure 118
	11.3.3 MonitorRouteReport Procedure 119
	11.4 Operation Timers..... 120
12	Parameter Descriptions..... 122
	12.1 Modified parameters 122
	12.1.1 scfID 122
	12.1.2 BearerCapability..... 123
	12.1.3 DpSpecificCommonParameters 123
	12.1.4 EventSpecificInformationBCSM 123
	12.1.5 GapCriteria 126
	12.1.6 Profile 126
	12.1.7 ServiceInteractionIndicatorsTwo 127
	12.2 New parameters..... 130
	12.2.1 AALParameters..... 130

	Page
12.2.2 AdditionalATMCellRate	130
12.2.3 AESACalledParty	130
12.2.4 AESACallingParty	130
12.2.5 AlternativeATMCellRate	130
12.2.6 ATMCellRate	130
12.2.7 BISDNParameters	130
12.2.8 CDVTDescriptor	131
12.2.9 ConnectionIdentifier	131
12.2.10 CumulativeTransitDelay	131
12.2.11 Cug-Index	131
12.2.12 Cug-Interlock	131
12.2.13 Cug-OutgoingAccess	131
12.2.14 DetachSignallingPath	131
12.2.15 DestinationIndex	131
12.2.16 EndtoEndTransitDelay	131
12.2.17 ExportSignallingPath	131
12.2.18 GenericIdentifier	132
12.2.19 IncomingSignallingBufferCopy	132
12.2.20 IPRelatedInformation	132
12.2.21 MergeSignallingPaths	132
12.2.22 MinimumATMCellRate	133
12.2.23 MonitoringCriteria	133
12.2.24 MonitoringTimeout	133
12.2.25 NoAnswerCause	133
12.2.26 QoSParameter	133
12.2.27 RouteCounters	133
12.2.28 StartTime	133
13 Errors	133
14 Services assumed from TCAP	136
14.1 Handling of SSME-Related messages	136
14.1.1 SSF to SCF messages	136
14.1.2 SCF to SSF messages	136
14.1.3 Use of TC_BEGIN primitive parameters	137
Annex A – ASN.1 Definitions	137

ITU-T Recommendation Q.1248.2

Interface Recommendation for Intelligent Network Capability Set 4: SCF-SSF interface¹

1 Scope

This Recommendation belongs to the Q.1248.x family of Recommendations for IN capability set 4. It specifies the protocol on the SSF-SCF interface and provides a description of the aspects of the SSF and SCF Functional Entities which are involved in the realization of this interface.

2 References

All ITU-T Recommendations and other references in this text are identified in ITU-T Q.1248.1.

3 Abbreviations and acronyms

All abbreviations and acronyms used in this text are defined in ITU-T Q.1248.1.

4 Relationships

4.1 SSF-CCF relationship

Call/service processing builds upon the current call process infrastructure of existing digital exchanges. It does so by using a generic model of existing call control functionality to process basic two-party calls, then adding service switching functionality to invoke and manage IN service logic. Once invoked, IN service logic is executed under the control of service control functionality, in conjunction with service data functionality. With this distributed approach to call/service processing, the existing call control functionality retains ultimate responsibility for the integrity of calls, as well as for the control of call processing resources.

The following call/service processing constraints apply:

- 1) Call control and service switching functionality are tightly coupled, thus the relationship between SSF and CCF is not standardized.
- 2) A call is either between two or more end users that are external to the network and addressable via a directory number or combination of directory number and bearer capability, or a call is between one or more end users and the network itself.
- 3) A call may be initiated by an end user, or by an SCF within the network on behalf of an end user. To supplement a call, IN service logic may either be invoked by an end user served by an IN exchange, or by the network on behalf of an end user.
- 4) A call may span multiple exchanges. As such, each exchange only controls the portion of the call in that exchange: call processing is functionally separated between exchanges. IN service logic invoked on IN exchanges in such an inter-exchange call is managed independently by each IN exchange.
- 5) Existing exchanges can be viewed as having two functionally separate sets of call processing logic that coordinate call processing activities to create and maintain a basic two-party call. This functional separation is provided between the originating portion of the call and the terminating portion of the call. This functional separation should be maintained in an IN exchange to allow IN service logic invoked on the originating portion of the call

¹ This Recommendation includes an electronic attachment containing Annex A, ASN.1 definitions.

- (i.e. on behalf of the calling party) to be managed independently of IN service logic invoked on the terminating portion of the call (i.e. on behalf of the called party).
- 6) It is desirable to allow multiple IN-supported service logic instances to be simultaneously active for a given end user. It is also recognized that non-IN service logic will continue to exist in the network. As such, service feature logic instances mechanisms should:
 - i) determine which service logic to invoke for a given service request. This mechanism should select the appropriate IN-supported service logic or non-IN-supported service logic, and block the invocation of any other service logic for that particular service request;
 - ii) limit simultaneously active IN- and non-IN-supported service logic instances;
 - iii) ensure that simultaneously active IN-supported service logic instances adhere either to the single-ended, single point of control (SPC) restriction or to the single-ended, multiple points of control (MPC) capability as introduced with IN CS-4. For each half call a CS-4 compliant CCF/SSF may behave either according to SPC rules or MPC rules depending on trigger table information provided at IN service triggering.
 - 7) The distributed approach and added complexity of call/service processing requires mechanisms for fault detection and recovery, allowing graceful termination of calls and appropriate treatments for end users.

4.2 SSF-SCF relationship

The SCF-SSF relationship is used for communication between an SCF and an SSF in the public network. This relationship, with the aid of possible other relationships (e.g. the SCF-SRF, SCF-SCF and SCF-SDF), provides a variety of services and service features. Details of service drivers for IN CS-4 can be found in ITU-T Q.1241.

A relationship between the SCF and SSF is established either as a result of the SSF sending a request for instruction to the SCF, or at the request of the SCF for initiation of a call or for some non-call-related reason.

A relationship between a SCF and a SSF is normally terminated at the request of the SCF. The SSF may also terminate the relationship, e.g. when no pending monitor requests prevails or in error cases.

A single SCF instance may have concurrent relationships with multiple SSF instances. A single SSF instance may have concurrent relationships with multiple SCF instances in the case where the IN CS-4 Multiple Points of Control applies to a half call instance. If Single Point of Control applies a single SSF instance it may only have a relationship with one SCF instance at a time for any given half-call instance. Note that this refers to control as opposed to monitor relationships.

It should be noted that the selection of an SRF is not always performed by the SCF. In some cases selection is performed by an SSF, for example, where assist/hand-off procedures are being used. For details regarding the relationship to the SRF see ITU-T Q.1248.3.

5 CCF Functional Model

5.1 Introduction

The Call Control Function (CCF) provides an abstraction of the part of a switch which implements basic call and connection control to establish communication paths for users, and to interconnect such communication paths.

Figure 1 depicts a functional decomposition of the CCF and its relationship to other components that are involved in call and service processing. Within the CCF, the Basic Call Control Entity is

responsible for the overall co-ordination of basic call processing. It interacts with the Bearer Control Entity that controls media resources.

The CCF and the media resources may reside in the same physical entity (e.g. traditional switching centre) or be implemented in two different physical nodes (e.g. see BICC architecture). This is not visible to IN service logic instances.

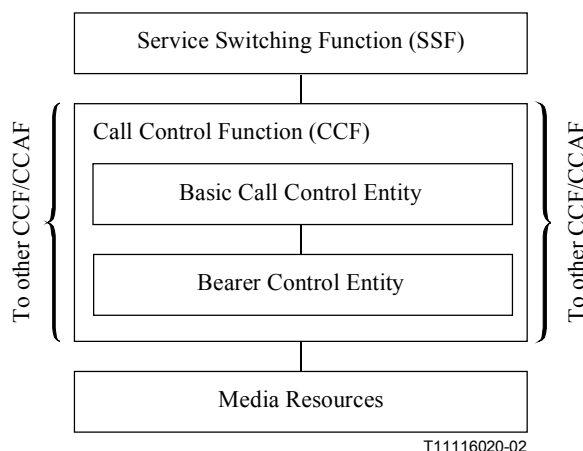


Figure 1/Q.1248.2 – Relationships between the CCF and other components

5.2 Basic Call Control Entity

5.2.1 Overview

5.2.1.1 Functional decomposition

The Basic Call Control Entity is responsible for the overall co-ordination of basic call processing in the CCF. As illustrated by Figure 2, the Basic Call Control Entity can be further modelled as being composed of a Basic Call Controller (BCC) and a set of Signalling Terminations (ST).

The Basic Call Controller manages a set of call processing state machines, which behaviour is described by the Basic Call State Model (BCSM). This model includes an originating half BCSM and a terminating half BCSM. This reflects the functional separation in the CCF model between the originating and terminating portions of calls that serve to isolate single-ended service logic instances related to the calling party from single-ended service logic instances related to the called party for the same call, i.e. the "half call" model approach.

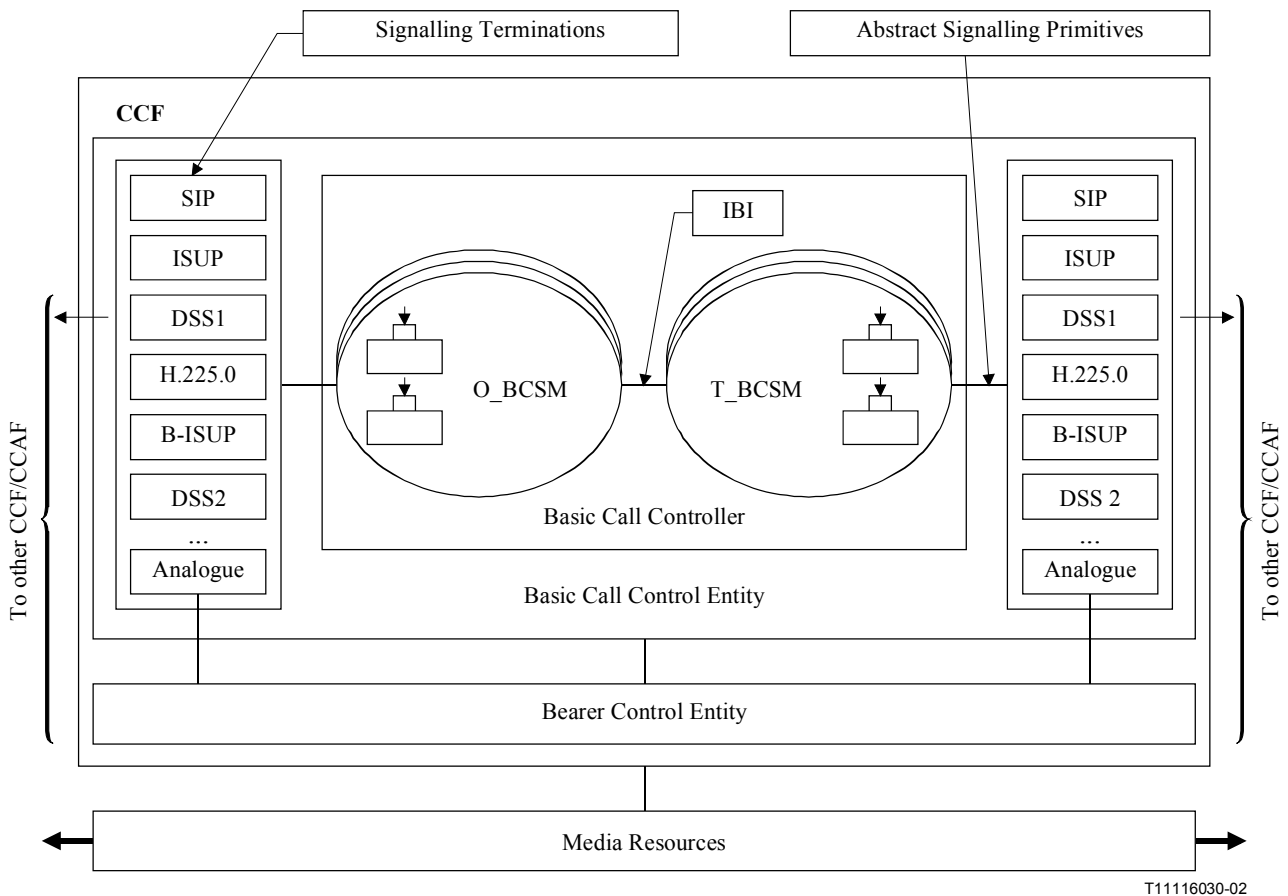


Figure 2/Q.1248.2 – Basic Call Control Entity

5.2.1.2 Basic Call Controller

The activities of the Basic Call Controller include:

- Creating and deleting BCSM instances.
- Routing signals (abstract signalling primitives) between BCSM instances and Signalling Terminations.
- Requesting services from the Bearer Controller (BRC).

The Basic Call Controller is the interface to the SSF. This relationship involves:

- Executing instructions that request the manipulation of one or more BCSM instances.
- Reporting events (i.e. DP) received from BCSM instances.

The Basic Call Controller is also in charge of relaying non-call processing related events between the SSF and the Signalling Terminations (e.g. USI information) or between the SSF and other local processes (e.g. charging information).

When executing instructions received from the SSF, the Basic Call Controller is responsible for the consistency between the configuration of the underlying bearer connections and the state of the Signalling Terminations FSM. As such, it may send an abstract signalling primitive to a Signalling Termination in order to bring the FSM to the correct state.

Example: As a result of an SCF instruction (e.g. MergeCallSegments), the Basic Call Controller Process may request the Bearer Controller Process to establish a physical connection between two parties, while the Signalling Termination associated to the originating half-call is not in the appropriate state (i.e. non answer received). In that case, the Basic Call Controller Process is responsible for sending the missing signal to the Signalling Termination.

The Basic Call Controller Process creates an O_BCSM instance under the following conditions:

- Receipt of a primitive from a Signalling Termination to initiate call establishment (i.e. Setup.Req)
- Receipt of an instruction from the SSF resulting from one of the following operations:
 - InitiateCallAttempt,
 - Connect in response to a TDP-R or EDP-R for a T_BCSM.

The Basic Call Controller creates a T_BCSM instance when it receives an indication from an O-BCSM instance that the PIC Send Call has been entered. After the T_BCSM has been created, the communication between the two instances uses the Inter-BCSM interface (IBI).

The Basic Call Controller Process deletes a BCSM instance under the following conditions:

- Indication from the BCSM that the Null state has been reached
- Receipt of an instruction from the SSF resulting from a MergeCallSegments operation.

For each BCSM instance it creates, the Basic Call Controller Process maintains the following knowledge:

- The reference of the peer BCSM instance.
- The reference of the associated Signalling Termination.
- The reference of the underlying bearer resource (i.e. Bearer Termination).

Using these references, the BCC can determine how many, and which BCSM instances are associated with the same Signalling Termination.

5.2.1.3 Signalling Terminations

Signalling Terminations provide adaptation functionalities between the Basic Call Controllers and signalling interfaces. Different types of Signalling Terminations may be available in a CCF. This depends on the list of signalling protocols supported by the exchange in which the CCF resides. Typical types of signalling terminations are:

- Analogue
- Q.763 ISUP,
- Q.931 DSS1
- Q.2763 B-ISUP,
- Q.2931 DSS2,
- H.225.0,
- SIP

NOTE – The signalling termination for analogue interfaces, requires access to the Bearer Controller in order to detect physical events such as off-hook/on-hook.

Each Signalling Termination embodies the knowledge of the signalling procedures defined for the supported protocol.

Signalling Termination instances communicates with the BCSM Controller Process, using a set of abstract signalling primitives defined in 5.2.6.2.

5.2.2 BCSM Model

5.2.2.1 Overview

The BCSM provides a high-level model description of CCF activities required to establish and maintain communication paths for users. As such, it identifies a set of basic call and connection activities in a CCF and shows how these activities are joined together to process a basic call and connection (i.e. establish and maintain a communication path for a user).

Many aspects of the BCSM are not externally visible to IN service logic instances. However, aspects of the BCSM that are reflected upward to the SSF are visible to IN service logic instances. Only these aspects of the BCSM will be the subject of standardization. As such, the BCSM is primarily an explanatory tool for providing a representation of CCF activities that can be analysed to determine which aspects of the BCSM will be visible to IN service logic instances, if any, and what level of abstraction and granularity is appropriate for this visibility.

The BCSM identifies points in basic call and connection processing when IN service logic instances are permitted to interact with basic call and connection control capabilities. In particular, it provides a framework for describing basic call and connection events that can lead to the invocation of IN service logic instances or should be reported to active IN service logic instances, for those points in call and connection processing at which these events are detected, and for describing those points in call and connection processing when the transfer of control can occur.

Figure 3 shows the components that have been identified to describe a BCSM, to include: points in call (PICs), detection points (DPs), BCSM transitions, and events. PICs identify CCF activities associated with one or more basic call/connection states of interest to IN service logic instances. DPs indicate states in basic call and connection processing at which transfer of control from non-IN to IN service logic can occur. Other states may exist (e.g. within PICs) where call processing is suspended but IN service logic cannot be invoked. BCSM transitions indicate the normal flow of basic call/connection processing from one PIC to another. Entry events cause BCSM transitions into PICs. Exit events represent the result of PIC processing.

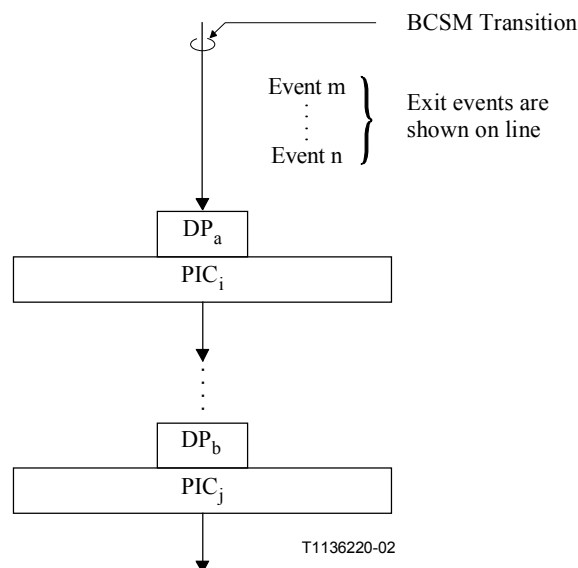


Figure 3/Q.1248.2 – BCSM components

The BCSM described in this clause is based on the overall BCSM in Annex A/Q.1204 and ITU-T Q.1224, refined as applicable.

The CCF model includes an originating half BCSM and a terminating half BCSM, each of which is managed by a functionally separate Basic Call Controller. This reflects the functional separation in the CCF model between the originating and terminating portions of calls that serve to isolate single-ended service logic instances related to the calling party from single-ended service logic instances related to the called party for the same call, i.e. the "half call" model approach.

The O_BCSM and T_BCSM respectively, model existing switch processing of a basic two-party call with the extended transitions as needed in support of IN call capabilities including CPH. In the following descriptions, the PICs are related at a high level to Q.931 ISDN call states. This is not intended to be a detailed formal definition of the relation between the PICs and Q.931 ISDN call states, but is intended as a point of reference to use in understanding the PICs. In particular, there are a number of possible ways in which the Q.931 call states may be traversed in certain situations that are considered below.

BCSM Information

In order to enable independence between services offered during one call session when the PICs may be traversed several times, it is necessary at each PIC to keep available a specific set of data until the calling (e.g. controlling) user releases the call and to ensure that software resources are returned to a coherent status when call processing passes through the PICs.

The BCSM Information depends on the applied signalling interface arrangement and is not listed in the following O/T_BCSM description. The general rule is that information that may be sent in the query (a service initiating operation as InitialDP or a DP specific operation) is to be maintained as long as an IN service could be triggered on the O/T_BCSM. More precise rules and mapping tables are provided in specific documents; e.g. ITU Q.1601.

The information that is sent to the SCF at a given trigger detection point is a subset of the information described here. Other information may be available at a given PIC that is not used by processing at the PIC or is only used by underlying call processing.

DP Naming

In order to maintain uniqueness of DP names between the originating and terminating half BCSMs, "O" and "T" are prefixed to certain originating and terminating DP names, respectively.

For ease of reference, the DPs associated with the BCSM transition implied by each entry and exit event for each PIC are listed along with the PIC descriptions.

BCSM Transitions

The BCSM description provided hereafter only describes the *basic transitions* which occur when processing a basic two party call. A basic call signifies a call between two users that consists of communication only, and does not include additional features (ITU-T Q.1290).

The PIC entry event is normally the Resume instruction received from the SSF after the processing of a DP: "basic" transition DP to PIC. PIC exit events are the result of the processing of an event received from the signalling interface arrangement supported by IN: "basic" transitions PIC to DP/PIC. The transitions which may occur due to an SCF instruction ("extended" transitions from DP to DP/PIC) and which may change the basic evolution of the call are not described here. Later subclauses describe the complete set (i.e. also the "extended" transitions from DP to DP/PIC) of possible BCSM transitions for the O_BCSM and the T_BCSM respectively.

5.2.2.2 Originating BCSM

The originating half of the BCSM corresponds to that portion of the BCSM associated with the originating party.

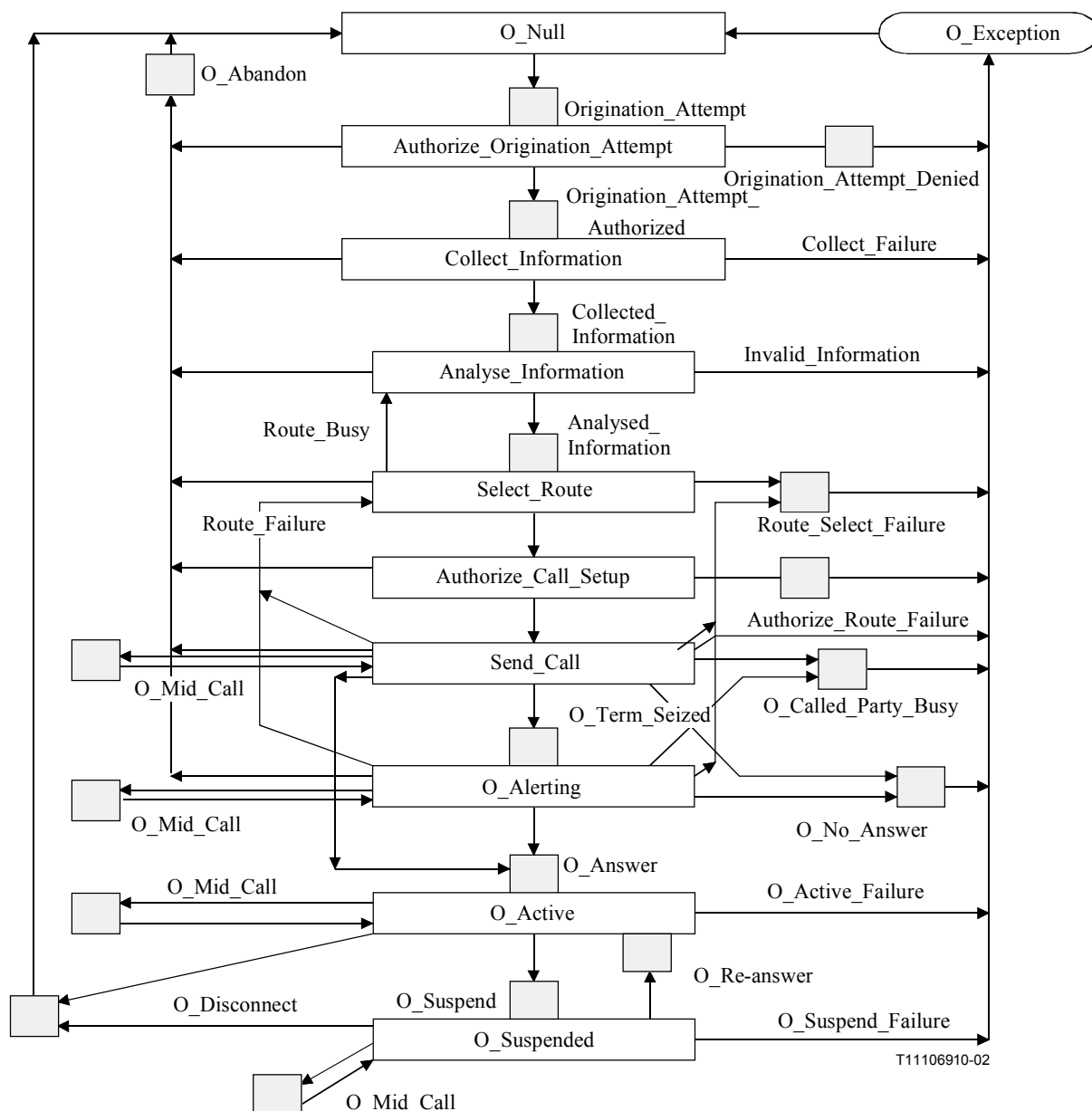


Figure 4/Q.1248.2 – Originating BCSM for IN CS-4: Basic Set of Transitions

The following information is available at all PICs in the O_BCSM

- Calling Party Class of Service (where locally available e.g. subscriber profile including calling party category information).
- Call Reference (where locally assigned in the switch).
- Terminal Type: see ITU-T Q.1290. The SCF uses this to determine the most appropriate form of user-interaction to use (e.g. in-band announcements). This information may only be available at originating local exchanges.
- Calling Party Facility Restriction Level (FRL). The permission level associated with an incoming facility, e.g. trunk line.
- Calling Party BGID (Business Group Identity).
- The Basic Business Group or Multiswitch Business Group Identity of the calling party, e.g. group-based services.

The above information applies in addition to Signalling Information received from the signalling interface and from the peer T_BCSM. The Signalling Information depends on the signalling interface arrangement applied: e.g. DSS1, DSS2, analogue user, conventional trunk, SS No. 7 trunk and N-ISUP or B-ISUP, private trunk, etc.

The description for each of the PICs in the originating half of the BCSM is provided below.

NOTE – See 5.2.4 "BCSM Indications for the CS-3 Call Model" for more information concerning PICs.

5.2.2.2.1 O_Null

Entry events: Disconnect and clearing of a previous call (DPs: O_Disconnect and O_Abandon), or default handling of exceptions by CCF/SSF completed.

Functions: Interface (line/trunk e.g. DSS1/N-ISUP or DSS2/B-ISUP interface) is idled (no call exists, no call reference exists, etc.). Supervision is being provided.

Information available: After detecting the Origination Attempt event, it is assumed that the CCF/SSF has the information corresponding to the Signalling Information received from the signalling interface arrangement as a Setup service request primitive is received (due to e.g. a DSS1 Setup, DSS2 Setup, N-ISUP IAM, B-ISUP IAM) available and associated with the originating call portion. If the CCF/SSF determines that the call origination attempt is denied, the cause of the failed authorization is also known. The above information applies in addition to the information available at all PICs in the O_BCSM.

Any information relating to switch-based features that have already been invoked for the call will also be available.

Exit events

- Indication of desire to place an outgoing call by the receipt of a Setup service request primitive (e.g. due to off-hook, Q.931/Q.2931 setup message, N/B-ISUP IAM message). This event causes call processing to move to the Origination_Attempt DP.
- The following exception exit events are applicable to the O_Null PIC. For this PIC, if the call encounters one of these exceptions during O_Null PIC processing, the exception event is not visible because there is no corresponding DP.

Corresponding Q.931 call state: O_Null.

NOTE – The O_Abandon event occurs when the originating party abandons the call before the call has been answered, i.e. when the calling party disconnects. For example, this event can result from one of the following:

- the CCF/SSF receives an on-hook indication from a caller served by a non-ISDN line, following switch-hook flash timing;
- the CCF/SSF receives a call clearing message from a caller served (e.g. by a narrow-band or broadband ISDN interface (DSS1/DSS2));
- the CCF/SSF receives a disconnect indication (e.g. from a conventional trunk or private facility trunk);
- the CCF/SSF receives a Release Message (e.g. from an SS No. 7 trunk, N-ISUP or B-ISUP interface).

5.2.2.2.2 Authorize_Origination_Attempt

Entry events: An indication is available that the originating terminal needs to be authorized. (DP: Origination_Attempt).

Functions

- The originating terminal rights should be checked using the calling party's identity and service profile. The authority/ability of the party to place the call with given properties (e.g. bearer capability, line restrictions) is verified. The types of authorization to be performed may vary for different types of originating resources (e.g. for lines vs. trunks).
- Other features which might be required during this PIC are not described.

Information available: After detecting the `Origination_Attempt_Authorized` event, it is assumed that the CCF/SSF has the same information available associated with the originating portion of the call as it did after detecting the `Origination_Attempt` event in the `O_Null` PIC.

- Information as per the `O_Null` PIC.
- Authorization result: If the CCF/SSF determines that the origination attempt is denied, the cause of the failed authorization is also known.

Exit events

- An indication is received that the authorization is successful. This event causes call processing to move to the `Origination_Attempt_Authorized` DP.
- The `O_Abandon` event occurs when an indication of clearing or of originating party abandon is received. This event causes call processing to move to the `O_Abandon` DP.
- An indication is received that the call origination is denied. The `O_BCSM` moves to the `Origination_Attempt_Denied` DP.

5.2.2.2.3 Collect_Information

Entry events: Authority/ability to place outgoing call verified (DP: `Origination_Attempt_Authorized`).

Functions

- Initial information package/dialling string (e.g. service codes, prefixes, dialled address digits) being collected from originating party. Information being examined according to dialling plan to determine end of collection. No further action may be required if an *en bloc* signalling method is in use (e.g. an ISDN user using *en bloc* signalling, an incoming SS No. 7 trunk).
- The CCF/SSF shall be able to support subsequent digit collection according to trigger criteria assigned before sending the query. For example, if a feature code (e.g. *64) is entered, the CCF/SSF may:
 - collect digits according to the normal dialling plan; or
 - collect a variable number of digits.

Information available: After the CCF/SSF determines that information collection is complete, it is assumed that the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the `O_Null` PIC.
- Collected Information: As obtained from the Setup service request and illustrated in the examples below.

The Collected Information may, e.g. consist of one or more of the following:

- Access Codes within a Customized Dialling Plan (CDP): see ITU-T Q.1290. The Customized Dialling Plan (CDP) in force may specify that after a given access code is dialled, more digits are to be collected according to the "normal dialling plan," i.e. the dialling plan in force. In this case, Access Code and Collected Address Information are known. If the CDP in force specifies that after a given access code is dialled, a variable number of digits are to be collected, then Access Code and Collected Digits are known.
- Feature Code: see Q.1290 and Q.762 Feature Code Signalling Information where this parameter is defined for national use only. If the numbering plan in force specifies that after a given feature code is dialled, more digits are to be collected according to the "normal dialling plan", then Feature Code and Collected Address Information are known. If the dialling plan in force specifies that after a given feature code is dialled, a variable number of digits are to be collected, then Feature Code and Collected Digits are known. The service associated with the feature code is dependent upon the users service profile.
- Facility Code: see ITU-T Q.1290. This information may be provided if and when facility selective service signalling is supported.
- Feature Activation: see Q.932 Feature Activation information element. If the CDP in force specifies that after a given feature activator is received, more digits are to be collected according to the numbering plan, then Feature Activation Indicator and Collected Address Information are known. If the CDP in force specifies that after a given feature activator is received, a variable number of digits are to be collected, then Feature Activation Indicator and Collected Digits are known.
- Prefix.
- Carrier Access Code/Carrier Identification Code: see ITU-T Q.1290. The caller may dial a Carrier Access Code (CAC) (e.g. a 101XXXX for use on this call). When the caller is served by an ISDN interface, a Carrier Identification Code, i.e. XXXX, may be received by the CCF/SSF within the transit network selection information element of the ISDN SETUP message.
- Collected Address Information: see ITU-T Q.1290. Available as per the numbering plan.
- Numbering Plan Indicator: see Q.762 Numbering Plan Indicator signalling information.
- Collected Digits: see ITU-T Q.1290. The numbering plan in force may specify that after a given Feature Activation, Feature Code, or Access Code within a CDP is dialled, a variable number of digits is to be collected using normal inter-digit timing. In this case, these collected digits are also known at this time.
- Charge Number: see ITU-T Q.1290. This is only known from a conventional trunk when Multi-Frequency signalling is used on the originating trunk (network operator specific). In this case, the charge number is provided in the second stage of overlap out pulsing.
- Numbering Plan Indicator: see Q.762 Numbering Plan Indicator signalling information. The address received is expected to conform to ITU-T E.164.
- Carrier Selection: see ITU-T Q.1290.
- Originating Line Information: see ITU-T Q.1290. This information is only known when MF signalling is used on the originating trunk (network operator specific). In this case, the Originating Line Information is sent during the second stage of overlap out pulsing.
- Travelling Class Mark (if provided) (included in the generic digits parameter of the IAM) see ITU-T Q.1290.

Exit events

- Availability of complete initial information package/dialling string from originating party. (This event may have already occurred in the case of en bloc signalling, in which case the waiting duration in this PIC is zero.). This event causes call processing to move to the Collected_Information DP.
- The following exception exit events are applicable to this PIC: Collect_Failure. The Collect_Failure event encompasses events such as CollectTimeout, CollectInfoFailure and InvalidInformation.
 - The CollectTimeout event is detected when enough information to process the call was not received by the CCF/SSF before a normal inter-digit timer expires. For an SS No. 7 trunk, this event corresponds to the IAM not containing the information necessary to process the call. In this case there may be no timing involved (timing may be involved for ISUP overlap sending).
 - The CollectInfoFailure event is detected when the CCF/SSF is unable to perform the information collection due to a lack of switch resources (e.g. no digit receivers are available).
 - The InvalidInformation event occurs when the information received from the caller is not valid, for instance the information received violates the dialling plan in force.
 - The O_Abandon event occurs when an indication of clearing or of originating party abandon is received. This event causes call processing to move to the O_Abandon DP.

Comment: Some digit analysis is required to determine the end of dialling. However, it is assumed that this analysis may be modelled as separable from the rest of digit analysis, which occurs in the Analyse_Information PIC. There is no intention to specify an implementation. However, a switch should externally present the separable view described for closed numbering plans (Note 1).

In the case of complete number (e.g. ISDN en bloc sending), the receipt of a SETUP message detected at the Origination_Attempt_Authorized DP causes the BCSM to pass through the Collect_Information PIC to the Collected_Information DP. Note that the BCSM transitions to Collected_Information DP when the initial information package/dialling string is received from the calling party. This occurs when enough information is received to proceed with call processing (e.g. as in the case of ISDN overlap sending of MF out pulsing). Specifically, for the digit-by-digit collection case, if the Collected_Information DP is armed as a Trigger Detection Point-Request (TDP-R), the SSF sends the query, i.e. an initialling DP request (i.e. InitialDP or CollectedInformation operation) to the SCF when enough digits are received to determine if the TDP criteria are met. It suspends BCSM processing but will collect further digits. It is network operator specific to determine when complete information is available (Note 2).

NOTE 1 – This separable view is provided by supporting distinct DPs. The Collected_Information DP is used after digit collection and the Analysed_Information DP is used after the rest of the digit analysis.

NOTE 2 – In some networks, it may be not possible for the CCF/SSF to determine when the called number information is complete. Therefore, TDP criteria for Collected_Information DP may be met in such networks before the called number information is complete.

Corresponding Q.931 call state: 1. Call Initiated and (optionally) 2. Overlap Sending.

5.2.2.2.4 Analyse_Information

Entry events: Availability of complete initial information package/dialling string from originating party (DP: Collected_Information) or Route_Busy event reported from the Select Route PIC.

Functions: Information being analysed and/or translated according to dialling plan to determine a destination (See ITU-T E.410) and call type (e.g. local exchange call, transit exchange call, international exchange call). To each destination, a list of routes is associated.

Based on the destination information derived from the analysis of the called party number, switched-based call gapping evaluation is performed. In case the called party number is modified by an IN service (i.e. in case a Connect operation is received), the switched-based call gapping evaluation is invoked each time the BCSM enters the Analyze_Info_PIC.

NOTE – The term "call gapping" should be understood as a CCF-based process which performs selective treatment of call establishment (e.g. releasing) in case of overload situations at switch-level. This process filters calls, as opposed to service triggering as is done by the SSF-based call gap process.

In the case that the SSF has received a list of CDPN in a Connect or AnalyzeInformation operation, the CCF/SSF shall use the next CDPN in the list.

The following provides details as to when the next CDPN/Route should be tried under each condition.

Parameters Received: DestinationRoutingAddress

Result: Busy or No Answer

Processing Order: CDPN1

Since no forwarding condition is specified, only the first CDPN is tried.

Parameters Received: DestinationRoutingAddress, ForwardingCondition

Result: Busy/No Answer

Processing Order: CDPN1

CDPN2

CDPN3

So long as the forwarding condition is met, the next CDPN is tried.

Parameters Received: RouteList, DestinationRoutingAddress

Result: No route out of local switch (busy)

Processing Order: Route1, CDPN1

Route2, CDPN1

Route3, CDPN1

No other CDPNs need be tried since there are no routes out of the switch.

Parameters Received: RouteList, DestinationRoutingAddress

Result: Route busy is found to be at a switch other than local switch

Processing Order: Route1, CDPN1

No other routes need be tried since the local route is successful, but a busy condition was encountered elsewhere. No other CDPNs are tried since forwarding condition does not apply.

Parameters Received: RouteList, DestinationRoutingAddress

Result: No answer received

Processing Order: Route1, CDPN1

No other routes need be tried since the route is successful, but a no answer condition was encountered.

No other CDPNs are tried since forwarding condition does not apply.

Only after the appropriate CDPNs/Routes have been tried will the call move onto the O_Called_Party_Busy DP, O_No_Answer DP, or Route_Select_Failure DP as appropriate.

One of the results of processing in this PIC is determination of routing address:

- i) called party number only (called party number is served by the SSF);
- ii) called party number and route index, where the route index is a pointer to a trunk group to route an outgoing call attempt on (called party number is served by another SSF);
- iii) called party number and route index, where the route index is a pointer to a list of trunk groups to route an outgoing call attempt on (called party number is served by another SSF).

Information available: After the CCF/SSF determines that the information has been analysed, it is assumed that the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the O_Null PIC.
- Analysis Results (of the Collected Information), as described in the examples below.

The Analysis Results consists of e.g. one or more of the following:

- Called Party Number.
- Numbering Plan Indicator.
See Q.762 Numbering Plan Indicator signalling information.
- Type Of Call: see ITU-T Q.1290.
- Carrier: see ITU-T Q.1290.
- Carrier Identification Code: see ITU-T Q.1290. Available for Internetwork carrier calls.
- Carrier Selection: see ITU-T Q.1290. Available for Inter-Serving Area ID carrier calls.
- Collected Information: Access Code within a CDP, Feature Code, Feature Activation, Prefix, Carrier Access Code/Carrier Identification Code, Collected Address Information/Digits as described under the Collect_Information PIC.
- Charge Number: as defined in the O_Null PIC (for an SS No.7 trunk) or Collected_Information PIC (for a conventional trunk when EAMF signalling is used).
- Originating Line Information: see ITU-T Q.1290. Available for Inter-Serving Area ID carrier calls.
- Route Index: see ITU-T Q.1290. Available if this call does not terminate on this CCF/SSF.
- Travelling Class Mark: see ITU-T Q.1290. Available if received on the facility.

Exit events

- Availability of routing address and nature of address. This event causes call processing to move to the Analysed_Information DP.
- The following exception exit events are applicable to this PIC: InvalidInformation The InvalidInformation event occurs when the information received from the caller is not valid and no further treatment (like e.g. routing of the call to an announcement in case of a wrongly dialled number) can be applied to the call, for instance in case the information received violates the dialling plan in force.
- The O_Abandon event occurs when an indication of clearing or of originating party abandon is received. This event causes call processing to move to the O_Abandon DP.

Comments: Note that routing address does not necessarily mean that the final physical route has been determined (e.g. route list has not been searched, hunt groups have not yet been searched, directory number has not yet been translated to physical port address), though this may be the case (e.g. when routing to a specific private facility).

Corresponding Q.931 call state: Not applicable.

5.2.2.2.5 Select Route

Entry events: Availability of routing address and call type (DP: `Analysed_Information`) or route busy event reported from the `Send_Call` or `O_Alerting` PICs.

Functions

- Routing address and call type are being interpreted. The next route is being selected. This may involve sequentially searching a route list, translating a directory number into physical port address, etc. The individual destination resource out of a resource group (e.g. a multi-line hunt group, a trunk group) is not selected. In some cases (e.g. an analogue line interface), a single resource (not a group) is selected.

NOTE – In a broadband environment, the switching system may select the path (VPI) and channel (VCI) in both directions of the connection.

- The individual destination resource of a resource group (e.g. a specific line if a multi-line hunt group) is not selected.
- When the entry event is the route failure event from the `Send_Call` PIC (see below), the CCF/SSF must first check the Route Failure Condition 1, Route Failure Condition 2, or Route Failure Condition 3 as defined under the `Send_Call` PIC exit events. If these conditions are true, then the call shall proceed to the `Analyse_Information` PIC in order to use the next CDPN.

If these three conditions mentioned above are not met, then, depending on the location in the network where the route is busy, the action is one of the following:

- If the trunk group selected for the call is busy at this switch, the CCF/SSF attempts to route the call on the next trunk group that has been specified for the call (when a route list is being searched or alternate routes are specified by the SCF). Call processing moves to the `Analyse_Information` PIC when one of two conditions occurs: all private-facility trunk groups have been tried and routing over a public facility is allowed, or routing to a particular intra or internetwork carrier has been tried and an alternate carrier is allowed.
- If all of the trunk groups (private and public) have been tried and no route is available, the `Route_Select_Failure` event is detected.
- If route busy is detected at another switch, an indication of this condition may be received via SS No. 7 signalling. In this case, a `Route_Select_Failure` event is detected.

Information available: After the CCF/SSF determines the route has been selected, it is assumed the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the `O_Null` PIC.
- Analysis Results: See description in the `Analyse_Information` PIC.
- Routing Information: When more than one route has been specified for the call (either by the SCF or as part of the information stored at the CCF/SSF), the CCF/SSF remembers what routes have been tried for this call and which route to select next. If the call is to an Inter-Serving Area ID carrier, the routing information includes Circuit Code information.

Exit events

- Route Selected event: Route selection is successful and call processing moves to `Authorize_Call_Setup` PIC.
- Unable to select a route (e.g. unable to determine a correct route, no more routes on route list) or indication from the terminating half BCSM that call cannot be presented to the terminating party (e.g. network congestion). This event causes call processing to move to the `Route_Select_Failure` DP. The event indication received from `T_BCSM` that causes call

processing to move to the O_Route_Select_Failure DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".

- The Route_Busy event leading to the Analyse_Information PIC as described above, or the following four conditions are met:
 - a) unable to select a route (e.g. unable to determine a correct route, no more routes on route list) or indication from the terminating half BCSM that call cannot be presented to the terminating party (e.g. network congestion);
 - b) the forwarding condition is busy, as specified by the ForwardingCondition parameter;
 - c) the route was determined by switch translations at the Analyse_Information PIC;
 - d) there are more CDPNs left to try.

Route_Busy is a non-IN transition which is part of a basic call. This is needed in case the CCF/SSF needs to process a list of CDPNs. These CDPNs can be returned in a Connect or Analyse_Information message. Upon being unable to select a route, the Route_Busy BCSM transition returns to the Analyse_Information PIC to analyse the next CDPN in the list.

- Originating party abandons call (DP: O_Abandon).

5.2.2.2.6 Authorize_Call_Setup

Entry events: Route Selected event.

Functions: The authority of the calling party to place this particular call is verified.

Information available: After the CCF/SSF determines the call setup has been authorized, it is assumed the CCF/SSF has the following information available and associated with the originating portion of the call with restrictions as noted:

- Information as per the O_Null PIC.
- Analysis Results: see description in the Analyse_Information PIC.
- Routing Information: see description in the Select_Route PIC.

Exit events

- Call Setup Authorized event. The Call Setup Authorized event occurs when the authority to place the call is verified. For example for an SS No. 7-supported trunk interface, if the received IAM indicates that a continuity check is being performed on the call connection and the call terminates to a non-ISDN line or ISDN interface, the Call Setup Authorized event occurs when an ISUP Continuity Message (COT) with a successful indication is received.
- The O_Abandon event occurs when an indication of clearing, or originating party abandon, is received. This event causes call processing to move to the O_Abandon DP.
- The Authorization Failure event occurs when the authority to place the call is denied (e.g. business group restriction mismatch, toll restricted calling line). For example, for an SS No. 7-supported trunk interface, the Authorization Failure event occurs when the continuity check procedure results in failure. This event causes a BCSM transition to the Authorize_Route_Failure DP.

5.2.2.2.7 Send_Call

Entry events: Call Setup Authorized event.

Functions: The CCF/SSF sends an indication of the desire to set up a call to the specified Called Party ID to the terminating call portion.

The information that may be passed to the terminating call portion is, for example: Charge Number, Calling Party ID; Calling Party BGID; Calling Party Category (determined by the Class of Service information or ISUP originating line information parameter); Bearer Capability; Called Party ID; Calling Party Subaddress; Called Party Subaddress; Forward GVNS, Carrier; Route Index; Carrier Identification Code. Circuit Code, and Carrier Selection; and TCM. Other feature information not used by the processing modelled by this PIC (e.g. call forwarding, generic name, and business group information) may also be passed to the terminating call portion.

Examples of functions during the processing that this PIC models:

For an ISDN or B-ISDN caller, the CCF/SSF returns a CALL PROCEEDING message. For non-ISDN lines, conventional trunks, and private-facility trunks, no treatment is applied.

For SS No. 7-supported trunks, if the received IAM indicates a continuity check is required in this circuit, the procedures for performing the continuity check are followed. If the continuity check is successful, an indication of continuity success is passed to the terminating call portion. If the received IAM indicates a continuity check is being performed on a previous circuit in the connection, upon receiving a COT with a successful indication, this indication is passed to the terminating call portion.

For B-ISUP interfaces, the CCF/SSF returns an IAM acknowledgement message (IAA).

Information available: After the CCF/SSF determines the call has been delivered (to the terminating half), it is assumed the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the O_Null PIC.
- Analysis Results: see description in the Analyse_Information PIC;
- Routing Information: see description in the Select_Route PIC;
- Feature Activation Information: see description below: A service feature request is received from the originating party: e.g. hook-flash, ISDN feature activator, DTMF provided control code.

Exit events

- A route failure event is detected when:
 - i) an indication of a T_Busy event specifying route busy; or
 - ii) a Call Rejected event specifying route busy (received when the route is found to be busy at a switch other than the local switch) is received from the terminating call portion.
 - iii) the following four conditions are met, hereafter called Route Failure Condition 1:
 - a) an indication of a T_Busy event specifying route busy (received when the route at the local switch is found to be busy) is received from the terminating call portion (presentation failure event from the Present Call PIC);
 - b) the forwarding condition is busy, as specified by the ForwardingCondition parameter;
 - c) the route was determined by switch translations at the Analyse_Information PIC;
 - d) there are more CDPNs left to try.
 - iv) the following four conditions are met, hereafter called Route Failure Condition 2:
 - a) a Call Rejected event specifying route busy (received when the route is found to be busy at a switch other than the local switch) is received from the terminating call portion (presentation failure event from the Present Call PIC);

- b) the forwarding condition is busy, as specified by the ForwardingCondition parameter;
 - c) the route was determined by the switch translations at the Analyse_Information PIC;
 - d) there are more CDPNs left to try.
- v) the following three conditions are met, hereafter called Route Failure Condition 3:
- a) O_Called_Party_Busy event or O_No_Answer event occurs (as specified below);
 - b) the forwarding condition is met, as specified by the ForwardingCondition parameter;
 - c) there are more CDPNs left to try.

NOTE – The Route_Failure event takes precedence over the O_Called_Party_Busy and O_No_Answer events.

In all five cases, the originating call portion returns to the Select_Route PIC if this event is not detected at a DP (e.g. more CDPNs left to try).

Otherwise, this event causes call processing to move to the O_Route_Select_Failure DP or O_Called_Party_Busy DP (e.g. User Busy, Subscriber absent (not reachable) or O_No_Answer DP or O_Exception). Which DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".

- An O_Answer event occurs when an indication of a T_Answer event is received from the terminating call portion. This event causes call processing to move to the O_Answer DP.
 - An O_Term_Seized event occurs when an indication of a call accepted event is received from the terminating call portion or when certain abnormal cases occur (e.g. like in ISDN when the call is offered to an ISDN interface and no user equipment has responded, but an "indication to apply audible ringing" is sent from the terminating call portion to the originating call portion and as a result, audible ringing is to be sent to the caller). This event causes call processing to move to the O_Term_Seized DP.
- For example, in ISDN this certain abnormal case occurs when a call is offered to the interface with a SETUP message and the call setup timer T303 is initiated. If T303 expires and no response is received, the switch retransmits the SETUP message, re-initializes T303, and sends in-band audible ringing back to the caller. If the call is from an ISDN user, a PROGRESS message is also sent containing progress indicator #10, "delay in response at called interface" and progress descriptor #8, "in-band tone or pattern not available". In this case, the calling party receives in-band audible ringing, however, the called party has not accepted the call. When the O_Term_Seized event occurs, the treatment applied depends on the originating access type.
- A service feature request event is detected from the originating party: e.g. hook-flash, ISDN feature activator, DTMF provided control code. This event causes call processing to move to the O_Mid_Call DP.

No additional actions are taken in a non-ISDN line or private facility trunk.

For a call originating from an ISDN interface, the caller also receives an ALERTing message or under certain conditions a PROGRESS message containing progress indicator information set to "in-band information or pattern now available".

For SS No. 7-supported trunks, an Address Complete Message (ACM) is sent.

In these cases, audible ringing, if applicable, is being sent from the originating call portion of the terminating switch.

- The O_No_Answer event is an IN event or signalling event. It can only occur when an O_No_Answer trigger is assigned and detected or when requested by a RequestReportBCSMEvent. If the O_No_Answer timer expires or an indication of the T_No_Answer event is received before an O_Answer event is detected (i.e. before the called party answers), the CCF/SSF reports the event to the SCF. The event indication received that causes call processing to move to the O_No_Answer DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".
- The O_Called_Party_Busy event occurs when an indication of a T_Busy event specifying user busy is received from the terminating portion of the call (e.g. network_determined_user_busy, user not reachable). This event also occurs when an indication of a Call Rejected event specifying user busy (i.e. user-determined-user busy) is received from the terminating portion of the call. For calls originating from non-ISDN lines, conventional trunks, and private-facility trunks, if an indication of busy is received from the terminating portion of the call and no originating triggers or requested events apply, busy tone is provided.
In addition to these busy events, "Call Rejected" conditions are also treated as O_Called_Party_Busy events.
In this case, the terminating portion of the call is cleared. The event that causes call processing to move to the O_Called_Party_Busy DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".
- For SS No. 7-supported trunk interface, the Authorization_Route_Failure event occurs when the continuity check procedure results in failure. This event causes a BCSM transition to the O_Exception.
- The O_Abandon event occurs when an indication of clearing or of originating party abandon is received. This event causes call processing to move to the O_Abandon DP.

5.2.2.2.8 O_Alerting

Entry events: O_Term_Seized event (DP: O_Term_Seized).

Functions

- Wait for the terminating party to answer. At this point, the caller receives in-band audible ringing (from the terminating switch). For example for a call originating from an ISDN interface, the caller also receives an ALERTing message or, under certain conditions, a PROGRESS message containing progress indicator information set to "in-band information or pattern now available".
- An indication of a call progress event may be received from the terminating call portion. This may result in a call progress indication being sent backward, e.g. CPG being sent on an SS No. 7-supported trunk (if the originating access is an SS No. 7-supported trunk) or an ALERTing or PROGRESS message being sent on an ISDN interface (if the originating access is an ISDN interface).

Information available: When the CCF/SSF is in this PIC, it is assumed the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the O_Null PIC.
- Analysis Results: see description in the Analyse_Information PIC.
- Routing Information: see description in the Select_Route PIC.
- Feature Activation Information: see description below:
A service feature request is received from the originating party: e.g. hook-flash, ISDN feature activator, DTMF provided control code.

Exit events

- The O_Answer event occurs when an indication of a T_Answer event is received from the terminating portion of the call (e.g. terminating party goes off hook, Q.931 Connect message received, ISUP Answer message received). This event causes call processing to move to the O_Answer DP. When the O_Answer event occurs, the treatment applied is described in the Send_Call PIC.
- A service feature request event is detected from the originating party: e.g. hook-flash, ISDN feature activator, DTMF provided control code. The detection of this event causes call processing to move to the O_Mid_Call DP.
- A route failure event is detected when:
 - i) the following conditions are met, hereafter called Route Failure Condition 3:
 - a) O_Called_Party_Busy event or O_No_Answer event occurs (as specified below);
 - b) the forwarding condition is met, as specified by the ForwardingCondition parameter;
 - c) there are more CDPNs left to try.

NOTE – The Route Failure event takes precedence over the O_Called_Party_Busy and O_No_Answer events.

In this case, the originating call portion returns to the Select_Route PIC if this event is not detected at a DP (e.g. more CDPNs left to try).

Otherwise this event causes call processing to move to the O_Route_Select_Failure DP or O_Called_Party_Busy DP (e.g. User Busy, Subscriber absent (not reachable) or O_No_Answer DP or O_Exception). Which DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".

- The O_No_Answer event from this PIC is the same as the O_No_Answer event defined as an Exit Event from the Send_Call PIC. That is, the event indication received that causes call processing to move to the O_No_Answer DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".
- From this PIC, the O_Called_Party_Busy event occurs either when:
 - i) a Call Rejected event specifying user busy is received; or
 - ii) when an indication of a Call Rejected event not specifying busy is received from the terminating call portion (as described in the Send_Call PIC).

In addition, for a call to an ISDN user, after the SETUP message is offered and an ALERTing message has been received (i.e. the terminating call portion is in the T_Alerting PIC), the ISDN user may reject the call. This Call Rejected event is treated as an O_Called_Party_Busy event by the originating call portion (DP: O_Called_Party_Busy). The event that causes call processing to move to the O_Called_Party_Busy DP depends upon the event reason (cause value) as defined according to Table 3 "O_BCSM: Mapping Cause value to DP".

- The O_Abandon event occurs when an indication of call clearing or of originating party abandon is received. This event causes call processing to move to the O_Abandon DP.

5.2.2.2.9 O_Active

Entry events: Indication from the terminating half BCSM that the call is accepted and answered by terminating party. (DP: O_Answer).

Functions: In this PIC several processes may be initiated e.g.:

- Connection established between originating and terminating party. Message accounting/charging data may be being collected. Call supervision is being provided.

- The called party may be put on hold and returned to the active phase by a service logic.
- The called party may be put on hold by a service logic and when the calling party disconnects, the calling party can be reconnected to the held call by the Reconnect operation. The calling user receives appropriate information (e.g. ringing) and a reconnection timer is applied.

Information available: Once the CCF/SSF has received an indication from the terminating half BCSM that the call has been answered, it is assumed the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the O_Alerting PIC.
- Feature Activation Information – A service or feature request from a party.

Exit events

- A service/service feature request is received from a party (e.g. DTMF provided control code, hook flash, ISDN feature activator, Q.931 HOLD or RETRIEVE message from originating party only) or a new reply from calling party (Reconnect procedure). The detection of this event causes call processing to move to the O_Mid_Call DP.
- A disconnect indication is received from the terminating party (non-ISDN subscriber) via the terminating half BCSM. This event causes call processing to move to the O_Suspend DP. A disconnect timing is associated with this BCSM transition.
- A disconnect indication is received from the originating party or terminating party (ISDN). This event causes call processing to move to the O_Disconnect DP.
- An indication of expiration of the reconnection timer is received. (O_Exception).
- A connection failure occurs. (O_Exception).

Comments:

- If originating party abandons call while suspended at O_Answer DP a transition to DP: O_Abandon shall occur.
- Disconnect treatment and timing is different for call attempts originating from ISDN and analogue line interfaces, e.g. at a release in the ISDN network from an ISDN line a transition direct to O_Disconnect DP occurs.

Corresponding Q.931 call state: 10. Active

Q.931 call states corresponding to disconnect: 11. Disconnect request, 12. Disconnect indication and 19. Release request.

5.2.2.2.10 O_Suspended

Entry events: A suspend indication is received from the T_BCSM when the terminating party has disconnected (e.g. on-hook). (DP: O_Suspend).

Functions

- The connection between the originating and terminating party is maintained and depending on the incoming network connection, appropriate backward signalling takes place.
 - In the case that a disconnect indication is received from the T_BCSM, this PIC is immediately exited to the O_Disconnect DP without any action. As an option, the call can be continued for an appropriate period in order to offer, e.g. follow-on call initiated by O_Mid_Call. DP
 - If the re-answer indication from the T_BCSM is received, the originating and terminating parties are reconnected.

- Other features which might be required during this PIC are not described in this IN Capability Set BCSM.
- The called party may be put on hold and returned to the active phase by a service logic.
- The called party may be put on hold by a service logic and when the calling party disconnects, the calling party can be reconnected to the suspended call by the Reconnect operation. The calling user receives appropriate information (e.g. rering) and a reconnection timer is applied. The timer may have been started in the active phase.

Information available: It is assumed the CCF/SSF has the following information available and associated with the originating portion of the call:

- Information as per the O_Active PIC.

Exit events

- Connection to the terminating party is resumed. The O_BCSM returns to the O_Active PIC. This event causes call processing to move to the O_Re-answer DP.
- A service feature request is received from the originating party, e.g. hook flash, DTMF provided control code, ISDN feature activator of facility or a new reply from calling party (Reconnect procedure). The detection of this event causes call processing to move to the O_Mid_Call DP.
- A disconnection indication is received from the originating party. This event causes call processing to move to the O_Disconnect DP.
- A disconnection indication is received from the terminating party. This event causes call processing to move to the O_Disconnect DP.
- An indication of expiration of the timer waiting for re-answer request is received from the T_BCSM. This event causes call processing to move to the O_Disconnect DP.
- A trigger at O_Mid_Call is not initiated during an appropriate period (DP: O_Disconnect).
- An indication of expiration of the calling party reconnection timer is received (O_Exception).
- An exception event is encountered. (O_Exception).

NOTE 1 – A Call Retention timer may exist. Disconnect treatment and timing is different for call reconnection, call suspension and call retention.

NOTE 2 – After the release of the outgoing connection, the originating party may initiate another call, e.g. a follow-on call.

5.2.2.2.11 O_Exception

Entry events: An exception condition is encountered (as described above for each PIC).

Functions: Default handling of the exception condition is being provided. This includes general actions necessary to ensure no resources remain inappropriately allocated, such as:

- If any relationships exist between the SSF and SCF(s), send an error information to the SCF(s) closing the relationships and indicating that any outstanding call handling instructions will not run to completion (Note).
- If an SCF previously requested that call parameters be provided at the end of the call (see the CallInformationRequest operation), these should be included in the error information.
- The CCF/SSF should make use of vendor-specific procedures to ensure release of resources within the CCF/SSF so that line, trunk, and other resources are made available for new calls.

The following information is available at all PICs in the T_BCSM:

- Same information as associated with the originating BCSM is assumed to be available. See clause defining the information available at all PICs in the O_BCSM.
- Called Party Class of Service (where locally available, e.g. subscriber profile including called party category information).
- Call Reference (where locally assigned in the switch).
- Called Party Terminal Type: see ITU-T Q.1290. The SCF uses this to determine the most appropriate form of user-interaction to use (e.g. in-band announcements). This information may only be available at terminating local exchanges.
- Called Party Facility Restriction Level (FRL). The permission level associated with an outgoing facility, e.g. trunk line.
- Called Party BGID (Business Group Identity): The Basic Business Group or Multiswitch Business Group Identity of the called party, e.g. group-based services.

The above information applies in addition to Signalling Information received from the signalling interface and from the peer O_BCSM. The Signalling Information depends on the signalling interface arrangement applied: DSS1, DSS2, analogue user, conventional trunk, SS No. 7 trunk and N-ISUP or B-ISUP, private trunk, etc. The descriptions for each of the PICs in the terminating half of the BCSM are described below.

NOTE – See 5.2.4 "BCSM Indications for the CS-4 Call Model" for more information concerning PICs.

5.2.2.3.1 T_Null

Entry events: Disconnect and clearing of a previous call (DPs: T_Disconnect or T_Abandon), or default handling of exceptions by CCF/SSF completed.

Functions: The Interface (e.g. line/trunk; DSS1/N-ISDN or DSS2/B-ISDN) is idled (no call exists, no call reference exists, etc.). Supervision is being provided.

Information available: Once the CCF/SSF has detected the Termination_Attempt event, it is assumed that the CCF/SSF has the information corresponding to the Signalling Information received from the signalling interface obtained from the Setup service request primitive (e.g. DSS1 Setup, DSS2 Setup, N-ISUP IAM, B-ISUP IAM) available and associated with the terminating portion of the call. The above information applies in addition to the information available at all PICs in the T_BCSM.

NOTE 1 – Information associated with the originating portion of the call as per the Send_Call PIC is assumed to be available: This information is received from the originating call portion, i.e. the peer O_BCSM.

Any information relating to switch-based features that have already been invoked for the call will also be available.

Exit events

- Indication of incoming call received from originating half BCSM. This event causes call processing to move to the Termination_Attempt DP.
- The following exception exit event is applicable to this PIC: T_Abandon. If the call encounters T_Abandon during PIC processing, the exception event is not visible because there is no corresponding DP.

Corresponding Q.931 call state: 0. Null

NOTE 2 – The T_Abandon event occurs when an indication of call disconnection is received from the originating portion of the call before the call has been answered, i.e. when the calling party disconnects. For example, this event can result from one of the following:

- the CCF/SSF receives an on-hook indication from a caller served by a non-ISDN line, following switch-hook flash timing;
- the CCF/SSF receives a call clearing message from a caller served by an ISDN interface;
- the CCF/SSF receives a disconnect indication from a conventional trunk or private facility trunk;
- the CCF/SSF receives a Release Message from an SS No. 7 trunk.

5.2.2.3.2 Authorize_Termination_Attempt

Entry events: Termination_Attempt event. (DP: Termination_Attempt).

Functions: Verifies the authority to route this call to the terminating access (e.g. DN or trunk group), e.g. check business group restrictions, restricted incoming access to line, or bearer capability compatibility.

Information available: It is assumed that the CCF/SSF has the same information available for the terminating call portion after the Termination_Attempt_Authorized event is detected as it does when the Termination_Attempt event is detected in the T_Null PIC.

- Information as per the T_Null PIC.
- Authorization result: If the CCF/SSF determines that the termination attempt is denied, the cause of the failed authorization is also known.

Exit events

- Termination_Attempt_Authorized event. This event occurs when the switch has verified the authority to terminate the call to the terminating access. This event causes call processing to move to the Termination_Attempt_Authorized DP.
- The Termination_Attempt_Denied event occurs when the authority to route these calls to the terminating user is denied. This causes a BCSM transition to the Termination_Attempt_Denied DP.
- The T_Abandon event occurs when an indication of clearing, or of originating party abandon, is received from the originating portion of the call. This event causes call processing to move to the T_Abandon DP.

5.2.2.3.3 Select Facility

Entry events: Termination_Attempt_Authorized event (DP: Termination_Attempt_Authorized) or an SS No. 7 failure occurs causing a re-attempt. The SS No. 7 failure in the Present_Call can be caused by a timer expiry upon sending the first Circuit Reservation Message (CRM), if applicable or a continuity check failure.

Functions: The busy/idle status of the terminating access is determined.

- For a non-ISDN line, if the line is already involved with an existing call, the line is treated as network-determined user busy.
- For a call terminating to an ISDN interface (on a non-shared DN/CT), network-determined user busy is the detection of one or more of the following conditions:
 - Interface busy: That is, a B-channel is not available for the call.
 - Call-reference busy: There are no idle call reference values available on the terminating DN/CT with which the call can be offered.

In addition, if the terminating DN is associated with a Multi-Line Hunt Group, busy means that no hunt terminals within the group are available and the queue, if any, is full.

- For conventional trunks, SS No. 7-supported trunks, and private-facility trunks, busy is when all trunks within the selected trunk group are busy.
- For a connection terminating to a B-ISUP interface, network-determined user busy is the detection of one or more of the following cause values in an Incoming Resources Rejected Request:
 - Resource Unavailable – Unspecified: There are no idle Signalling Identification Values.
 - No VPCI/VCI Available: There is no VPCI or VCI available.
 - User Cell Rate Not Available: There is a lack of bandwidth for the connection.
- For a call terminating to a B-ISDN DSS2 interface, network-determined user busy is the detection of one or more of the following conditions:
 - VPCI/VCI Not Available: There are no VPCIs or VCIs available to allocate to the connection.
 - Requested VPCI/VCI Not Available: The requested VPCI or VCI is not available to allocate to the connection.
 - Quality of Service unavailable: Requested Quality of Service or End-to-End Transit Delay not acceptable.
 - Resources Unavailable – Unspecified: Requested Cell rate in the ATM Traffic Descriptor is not available.

Information available: When the Facility_Selected_and_Available event is detected, it is assumed the following information is available and associated with the terminating portion of the call:

- Information as per the Authorize_Termination_Attempt PIC.
- Facility Group: see ITU-T Q.1290. For calls routed out of this CCF/SSF, this identifies the Trunk Group (private or public) that has been selected to route the call on. For calls terminating to a non-ISDN line or DSS1 interface within the CCF/SSF, this may identify a particular Multi-line Hunt Group.
- Facility Group Member: see ITU-T Q.1290. For calls out of this CCF/SSF, this identifies the trunk (private or public) that has been selected to route the call on. For calls terminating to a non-ISDN line DSS1 interface on the CCF/SSF, this may identify the hunt-terminal within the Multi-line Hunt Group that has been selected for this call.

Exit events

- Facility_Selected_and_Available event: This event occurs when the terminating access is not busy (i.e. an idle facility (e.g. B-channel, VPCI/VCI, call reference or trunk) could be found). This event causes call processing to move to the Facility_Selected_and_Available DP.
- A T_Busy event occurs when the terminating access is busy (as defined above). The T_busy event may also be detected as a result of an analogue line being out of order, marked as busy by a customer make-busy key, or as a result of certain maintenance actions. (DP: T_Busy). The event indication that causes call processing to move to the T_Busy DP depends upon the event reason (cause value) as defined according to Table 4 "T_BCSM: Mapping Cause value to DP".

After detecting T_Busy, if IN service logic is not needed on the call and no switch-based features apply, an indication of the T_Busy event describing the type of busy (e.g. user or network) is passed to the originating call portion. If a terminating feature acts on the T_Busy event and changes the event (e.g. as in the call Waiting feature), the event is not passed to the Originating BCM.

- The T_Abandon event occurs when an indication of clearing or of originating party abandon is received from the originating portion of the call. This event causes call processing to move to the T_Abandon DP.

5.2.2.3.4 Present Call

Entry events: Facility_Selected_and_Available event. (DP: Facility_Selected_and_Available)

Functions: Terminating resource informed of incoming call (e.g. line seizure, Q.931 Setup message, ISUP IAM message). In the case of an analogue line, ringing is applied.

Information available: When the Call Accepted event is detected, it is assumed the following information is available and associated with the terminating portion of the call:

- Information as per the T_Null PIC.
- Facility Group, Facility Group Member: See description in the Select Facility PIC.
- Information regarding the call connection: This information includes whether the call is end-to-end SS No. 7 or not and whether the originating access is ISDN or non-ISDN.

Exit events

- Terminating party is being alerted (e.g. ringing being applied, Q.931 ALERTING message, ISUP-ACM message). This event causes call processing to move to the Call_Accepted DP.
- Call is accepted and answered by terminating party (e.g. terminating party goes off-hook, Q.931 Connect message received, ISUP answer message received). This event causes call processing to move to the T_Answer DP.
- The T_No_Answer event occurs when the terminating party does not answer before the No_Answer timer expires or when an ISDN user rejects the call with an explicit "no answer" indication. An indication of T_No_Answer event is passed to the originating half of the BCSM. The event indication received that causes call processing to move to the T_No_Answer DP depends upon the event reason (cause value) as defined according to Table 4 "T_BCSM: Mapping Cause value to DP".
- The T_Abandon event occurs when an indication of clearing or of originating party abandon is received from the originating portion of the call. This event causes call processing to move to the T_Abandon DP.
- A timer expiry upon sending the first Circuit Reservation Message (CRM) or a continuity check failure. (SS No. 7 failure). This event causes call processing to move to the Select Facility PIC.
- Presentation Failure exception event may happen if the call cannot be presented by the T_Busy event.
- The T_Busy event occurs if the call cannot be presented due to an ISDN user determined busy, ISUP release message with busy cause, not reachable subscriber condition; this event is notified to the originating call portion (send Call PIC). The event indication that causes call processing to move to the T_Busy DP depends upon the event reason (cause value) as defined according to Table 4 "T_BCSM: Mapping Cause value to DP" Otherwise this event causes call processing to move to the T_Exception.

Corresponding Q.931 call state: 6. Call present

5.2.2.3.5 T_Alerting

Entry events: Terminating party is being alerted of incoming call (DP: Call_Accepted).

Functions: An indication is sent to the originating half BCSM that the terminating party is being alerted. Continued processing of call setup (e.g. ringing, audible ring indication) is taking place. Waiting for the call to be answered by terminating party.

Information available: Once the terminating party is being alerted of the incoming call, it is assumed that the CCF/SSF has the following information available and associated with the terminating portion of the call:

- Information as per the Present_Call PIC.

Exit events

- Call is accepted and answered by terminating party (e.g. terminating party goes off-hook, Q.931 CONNect message received, ISUP answer message received). This event causes call processing to move to the T_Answer DP.
- The following exception exit events are applicable to this PIC: call rejected, T_No_Answer, T_Busy and T_Abandon.
 - Call rejected exception event may happen when an ISDN user rejects a call while being alerted.
 - The T_No_Answer event occurs when the terminating party does not answer before the No_Answer timer expires or when an ISDN user rejects the call with an explicit "no answer" indication. An indication of T_No_Answer event is passed to the originating half of the BCSM. The event indication received that causes call processing to move to the T_No_Answer DP depends upon the event reason (cause value) as defined according to Table 4 "T_BCSM: Mapping Cause value to DP".
 - The T_Busy event occurs if the call cannot be presented due to, e.g. ISDN user determined busy, ISUP release message with busy cause, not reachable subscriber, etc. This event causes call processing to move to the T_Busy DP and is notified to the originating call portion (Alerting PIC). The event that causes call processing to move to the T_Busy DP depends upon the event reason (cause value) as defined according to Table 4 "T_BCSM: Mapping Cause value to DP". Otherwise this event causes call processing to move to the T_Exception.
 - The T_Abandon event occurs when an indication of clearing or of originating party abandon is received from the originating portion of the call. This event causes call processing to move to the T_Abandon DP.

Comment: For terminations to SS No. 7 trunk groups, this PIC is entered upon the receipt of an address complete (ACM) message.

Corresponding Q.931 call states: 7. Call received and 8. Connect request.

5.2.2.3.6 T_Active

Entry events: Call is accepted and answered by terminating party (e.g. terminating party goes off hook, Q.931 Connect message received, ISUP answer message received). (DP: T_Answer).

Functions: In this PIC several processes may be initiated:

- An indication is sent to the originating half BCSM that the terminating party has accepted and answered the call. Connection established between originating and terminating party. Call supervision is being provided.
- The calling party may be put on hold and returned to the active phase by a service logic.
- The calling party may be put on hold by a service logic and when the called party disconnects, the called party can be reconnected to the held call by the Reconnect operation. The called user receives appropriate information (e.g. rering) and a reconnection timer is applied. The timer may have been started in the active phase.

Information available: Once the call is accepted and answered by the terminating party, it is assumed the following information is available and associated with the terminating portion of the call:

- Information as per T_Alerting.
- Feature Activation Information: see description below. A service feature request is received from a party: (e.g. hook-flash, ISDN feature activator, DTMF provided control code, Q.932 HOLD or RETRIEVE message (for a terminating party only)).

Exit events

- A service/service feature request is received from a party (e.g. DTMF provided control code, hook flash, ISDN feature activator, Q.931 HOLD or RETRIEVE message from a terminating party only). The detection of this event causes call processing to move to the T_Mid_Call DP.
- A disconnect indication (e.g. on-hook, Q.931 disconnect message, SS No. 7 release message) is received from the terminating party (non-ISDN). This event causes call processing to move to the T_Suspend DP.
- A disconnect indication is received from the originating party via the originating half BCSM or from the terminating party (ISDN). This event causes call processing to move to the T_Disconnect DP.
- An indication of expiration of the reconnection timer is received. (T_Exception).
- A connection failure occurs. (T_Exception).

Comments:

- If originating party abandons call while suspended at T_Answer DP, a transition to T_Abandon DP shall occur.
- Disconnect indications and treatment are asymmetrical in the way disconnect timing is applied.

Corresponding Q.931 call state: 10. Active

Q.931 call states corresponding to T_Disconnect: 11. Disconnect request, 12. Disconnect indication, and 19. Release request

5.2.2.3.7 T_Suspended

Entry events: An indication is received from the outgoing network that the terminating party has disconnected (e.g. on-hook). (DP: T_Suspend).

Functions: The physical resources associated with the call remain connected.

According to the received indication the following applies:

- A suspend indication is sent to the originating half BCSM.
- For an SS No. 7-supported trunk or an ISDN interface, in the case that a disconnect indication (e.g. Q.931 disconnect message, SS No. 7 release message) is received from the terminating party, this PIC is immediately exited to the T_Disconnect DP without any action.
- In the following cases, the timer is started and the call waits for re-answer request from the terminating party:
 - 1) for an SS No. 7-supported trunk, in case of receiving network initiated suspend message;
 - 2) for an analogue interface, in case of detecting on-hook.

If re-answer request (e.g. off-hook, SS No. 7 resume message) is received from the terminating party before the timer expires, the originating and terminating parties are reconnected.

NOTE – Both a Call Resume timer and a Call Retention timer may exist in this PIC. IN implementations may use a single timer for both conditions.

Information available: While in the T_Suspended PIC, it is assumed that the CCF/SSF has the following information available and associated with the terminating call portion.

- Information as per the T_Active PIC.

Exit events

- The terminating party re-answers or a resume message is received before the timer expires; the T_BCSM returns to the T_Active PIC. This event causes call processing to move to the T_Re-answer DP.
- The timer expires. This event causes call processing to move to the T_Disconnect DP.
- A disconnection indication is received from the terminating party. This event causes call processing to move to the T_Disconnect DP.
- A disconnection indication is received from the originating party. This event causes call processing to move to the T_Disconnect DP.
- An exception event is encountered. (T_Exception).

5.2.2.3.8 T_Exception

Entry events: An exception condition is encountered (as described above for each PIC).

Functions: An indication of the exception condition is sent to the originating half BCSM. Default handling of the exception condition is being provided. This includes general actions necessary to ensure no resources remain inappropriately allocated, such as:

- If any relationships exist between the SSF and SCF(s), send an error information to the SCF(s) closing the relationships and indicating that any outstanding call handling instructions will not be run to completion (Note).
- If an SCF previously requested that call parameters be provided at the end of the call (see the CallInformationRequest operation), these should be included in the error information.
- The CCF/SSF should make use of vendor-specific procedures to ensure release of resources within the CCF/SSF so that line, trunk, and other resources are made available for new calls.

NOTE – Depending on the connection view state, this should be handled in the physical plane via an ABORT protocol procedure to close the relationship (i.e. close the TCAP transaction) or via the sending of an EntityReleased operation with the relevant cause. This indicates in both cases that any outstanding operations on the corresponding entity (leg or CS) will not be run to completion.

Information available: Once the CCF/SSF has determined an exception condition has occurred, it is assumed the CCF/SSF has information available as when the exception within the PIC occurred.

Exit events: Default handling of the exception condition by CCF/SSF completed (BCSM transition to T_Null PIC).

5.2.3 BCSM Resume Points and BCSM Transitions in the IN CS-4 Call Model

5.2.3.1 Originating BCSM: Detailed Set of O_BCSM Transitions

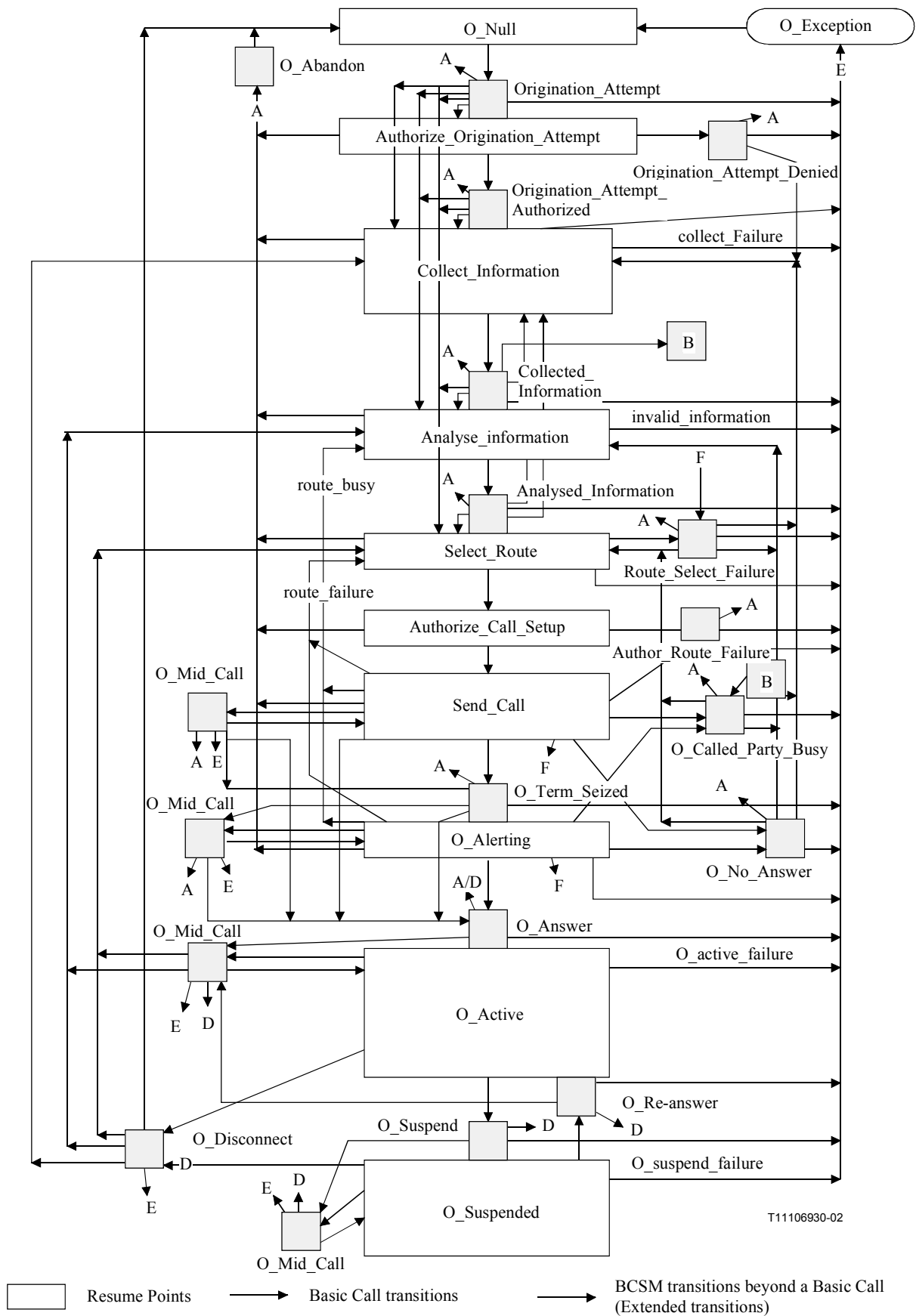


Figure 6/Q.1248.2 – Detailed Set of O_BCSM Transitions

Table 1 together with Figure 6 above describe the complete set of possible BCSM transitions for the originating call model. The nature of the O_BCSM transitions is given in the third column.

Basic

Basic transitions refer to normal processing of a basic call, i.e. a two-party call. The basic call process resumes in sequence of PIC and DP (e.g. Continue).

Extended

The basic call process progresses in non-sequence of PIC and DP compared to a basic call due to SCF instruction (e.g. Analysed_Information, ReleaseCall (e.g. early determined "busy" cause) or CCF call control instruction.

Table 1/Q.1248.2 – Complete Set of O_BCSM Transitions for the IN CS-4 Model

O_BCSM: Transitions from DP to DP/PIC_		
From	To	Nature of BCSM Transitions
Origination_Attempt DP	Authorize_Origination_Attempt PIC	Basic
	Collect_Information PIC	Extended
	Analyse_Information PIC	Extended
	Select_Route PIC	Extended
	O_Exception PIC	Extended
	O_Abandon DP	Extended
Origination_Attempt_Denied DP	Collect_Information PIC	Extended
	O_Abandon DP	Extended
	O_Exception PIC	Basic
Orig_Attempt_Authorized DP	Collect_Information PIC	Basic
	Analyse_Information PIC	Extended
	Select_Route PIC	Extended
	O_Exception PIC	Extended
	O_Abandon DP	Extended
Collected_Information DP	Collect_Information PIC	Extended
	Analyse_Information PIC	Basic
	Select_Route PIC	Extended
	O_Exception PIC	Extended
	O_Abandon DP	Extended
Analyse_Information DP	Collect_Information PIC	Extended
	Analyse_Information PIC	Extended
	Select_Route PIC	Basic
	O_Exception PIC	Extended
	O_Abandon DP	Extended
	O_Called_Party_Busy DP	Extended

Table 1/Q.1248.2 – Complete Set of O_BCSM Transitions for the IN CS-4 Model

O_BCSM: Transitions from DP to DP/PIC_		
From	To	Nature of BCSM Transitions
Authorize_Route_Failure DP	Analyze_Information PIC Select_Route PIC O_Abandon DP O_Exception PIC	Extended Extended Extended Basic
O_Term_Seized DP	O_Alerting PIC O_Answer DP (Note 1) O_Mid_Call DP (Note 3) O_Exception PIC O_Abandon DP	Basic Extended Extended Extended Extended
Route_Select_Failure DP	Collect_Information PIC Analyze_Information PIC Select_Route PIC O_Abandon DP O_Exception PIC	Extended Extended Extended Extended Basic
O_Called_Party_Busy DP	Collect_Information PIC Analyze_Information PIC Select_Route PIC O_Abandon DP O_Exception PIC	Extended Extended Extended Extended Basic
O_No_Answer DP	Collect_Information PIC Analyze_Information PIC Select_Route PIC O_Abandon DP O_Exception PIC	Extended Extended Extended Extended Basic
O_Answer DP	O_Active PIC O_Mid_Call DP (Note 3) O_Exception PIC O_Disconnect DP (Note 2) O_Abandon DP	Basic Extended Extended Extended Extended
O_Suspend DP	O_Suspended PIC O_Disconnect DP O_Mid_Call DP (O_Suspended PIC) O_Exception PIC	Basic Extended Extended Extended
O_Re-Answer DP	O_Active PIC O_Disconnect DP O_Mid_Call_Active DP O_Exception PIC	Basic Extended Extended Extended

Table 1/Q.1248.2 – Complete Set of O_BCSM Transitions for the IN CS-4 Model

O_BCSM: Transitions from DP to DP/PIC_		
From	To	Nature of BCSM Transitions
O_Mid_Call DP (Send Call PIC)	Send_Call PIC	Basic
	O_Mid_Call DP (Send_Call PIC)	Extended
	O_Term_Seized DP (Note 1)	Extended
	O_Answer DP (Note 1)	Extended
	O_Exception PIC	Extended
	O_Abandon DP	Extended
O_Mid_Call DP (O_Alerting PIC)	O_Alerting PIC	Basic
	O_Mid_Call DP (O_Alerting PIC)	Extended
	O_Answer DP (Note 1)	Extended
	O_Exception PIC	Extended
	O_Abandon DP	Extended
O_Mid_Call DP (O_Active PIC)	O_Active PIC	Basic
	Analyse_Information PIC	Extended
	Select_Route PIC	Extended
	O_Mid_Call DP (O_Active PIC)	Extended
	O_Exception PIC	Extended
	O_Disconnect DP	Extended
O_Mid_Call DP (O_Suspended PIC)	O_Suspended PIC	Basic
	Analyse_Information PIC	Extended
	Select_Route PIC	Extended
	O_Mid_Call DP (O_Suspended PIC)	Extended
	O_Exception PIC	Extended
O_Disconnect DP	O_Null PIC	Basic
	Collect_Information PIC	Extended
	Analyse_Information PIC	Extended
	Select_Route PIC	Extended
	O_Exception PIC	Extended
	O_Disconnect DP	Extended
O_Abandon DP	O_Null PIC	Basic

NOTE 1 – The DP reporting ensures that the SCF will receive the events in the correct order. For example, in the case where the **FSM for CS** is in "Waiting for Instructions" state due to an O_Mid_Call EDP-R (SendCall) being reported, and in this state O_Term_Seized and O_Answer events are detected, followed by the O_Disconnect event.

NOTE 2 – Release from called party.

NOTE 3 – If e.g. a DisconnectLeg (p) or MoveLeg (p) is received while call processing is suspended at DP O_Term_Seized respective O_Answer, the O_BCSM for leg c transits to the O_MidCall DP of the Send_Call PIC respective O_Alerting PIC (not of the O_Alerting PIC respective O_Active PIC as answer has not been sent backward on the c-leg).

Table 1/Q.1248.2 – Complete Set of O_BCSM Transitions for the IN CS-4 Model

O_BCSM: Transitions from PIC to PIC/DP		
From	To	Nature of BCSM Transitions
O_Null PIC	Origination_Attempt DP	Basic
Authorize_Origination_Attempt PIC	Origination_Attempt_Authorized DP Origination_Attempt_Denied DP O_Abandon DP O_Exception PIC	Basic Basic Basic Basic
Collect_Information PIC	Collected_Information DP O_Abandon DP O_Exception PIC	Basic Basic Basic
Analyse_Information PIC	Analysed_Information DP O_Abandon DP O_Exception PIC	Basic Basic Basic
Select_Route PIC	Analyse_Information PIC Authorize_Call_Setup PIC Route_Select_Failure DP O_Abandon DP O_Exception PIC	Basic Basic Basic Basic Basic
Authorize_Call_Setup PIC	Send_Call PIC Authorize_Route_Failure DP O_Abandon DP O_Exception PIC	Basic Basic Basic Basic
Send_Call PIC	O_Term_Seized DP O_Mid_Call DP (Send_Call PIC) O_Called_Party_Busy DP O_Answer DP O_No_Answer DP Select_Route PIC O_Abandon DP O_Exception PIC Analyse_Information PIC	Basic Basic Basic Basic Basic Basic Basic Basic Extended
O_Alerting PIC	Route_Select_Failure DP O_Mid_Call DP O_Answer DP O_No_Answer DP O_Called_Party_Busy DP O_Abandon DP O_Exception PIC Analyse_Information PIC	Basic Basic Basic Basic Basic Basic Basic Extended

Table 1/Q.1248.2 – Complete Set of O_BCSM Transitions for the IN CS-4 Model

O_BCSM: Transitions from PIC to PIC/DP		
From	To	Nature of BCSM Transitions
O_Active PIC	O_Midcall DP O_Disconnect DP O_Suspend DP O_Exception PIC	Basic Basic Basic Basic
O_Suspended PIC	O_Re-Answer DP O_Mid_Call DP (O_Suspended PIC) O_Disconnect DP O_Exception PIC	Basic Basic Basic Basic
O_Exception	O_Null PIC	Basic

Extended: The basic call process progresses in non-sequence of DP compared to a basic call due to SCF instruction PIC or CCF call control instruction

Table 2/Q.1248.2 – Complete Set of T_BCSM Transitions for the IN CS-4 Model

T_BCSM: Transitions from DP to DP/PIC		
From	To	Nature of BCSM Transition
Termination_Attempt DP	Authorize_Termination_Attempt PIC Select_Facility PIC Present_Call PIC (Note 1) T_Answer DP T_Abandon DP Call_Accepted DP T_Exception PIC	Basic Extended Extended Extended Extended Extended Extended
Termination_Attempt_Denied DP	Select_Facility PIC Present_Call PIC (Note 1) T_Abandon DP T_Exception PIC	Extended Extended Extended Basic
Termination_Attempt_Authorized DP	Select_Facility PIC Present_Call PIC (Note 1) T_Answer DP T_Abandon DP Call_Accepted DP T_Exception PIC	Basic Extended Extended Extended Extended Extended
Facility_Selected_and_Available DP	Present_Call PIC T_Answer DP T_Abandon DP Call_Accepted DP T_Exception PIC	Basic Extended Extended Extended Extended
Call_Accepted DP	T_Alerting PIC T_Answer DP (Note 2) T_Abandon DP T_Exception PIC T_MidCall DP (Note 4)	Basic Extended Extended Extended Extended
T_Busy DP	Select_Facility PIC Present_Call PIC (Note 1) T_Answer DP T_Abandon DP Call_Accepted DP T_Exception PIC	Extended Extended Extended Extended Extended Basic

Table 2/Q.1248.2 – Complete Set of T_BCSM Transitions for the IN CS-4 Model

T_BCSM: Transitions from DP to DP/PIC		
From	To	Nature of BCSM Transition
T_No_Answer DP	Select_Facility PIC Present_Call PIC (Note 1) T_Answer DP Call_Accepted DP T_Exception PIC	Extended Extended Extended Extended Basic
T_Answer DP	T_Active PIC T_MidCall DP (Note 4) T_Exception PIC T_Disconnect DP (Note 3) T_Abandon DP	Basic Extended Extended Extended Extended
T_Suspend DP	T_Suspended PIC T_Disconnect DP T_MidCall DP (T_Active PIC) T_Exception PIC	Basic Extended Extended Extended
T_Re-Answer DP	T_Active PIC T_Disconnect DP T_MidCall DP (T_Active PIC) T_Exception PIC	Basic Extended Extended Extended
T_Midcall DP	T_Active PIC T_Disconnect DP T_Exception PIC T_Midcall DP	Basic Extended Extended Extended
T_Disconnect DP	T_Null PIC T_Answer DP Call_Accepted DP T_Disconnect DP Present_Call PIC (Note 1)	Basic Extended Extended Extended Extended
T_Abandon DP	T_Null PIC	Basic
<p>NOTE 1 – When a Connect operation is received, the T_BCSM may be suspended in one of the indicated DPs. The T_BCSM moves to the Present_Call PIC.</p> <p>NOTE 2 – The DP reporting ensures that the SCF will receive the events in the correct order. For example, in case the FSM for CS is in "Waiting for Instructions" state due to CallAccepted EDP-R being reported, and in this state an T_Answer event is detected, followed by the T_Disconnect event.</p> <p>NOTE 3 – Release from called party.</p> <p>NOTE 4 – If, for example, a DisconnectLeg (c) or MoveLeg (c) is received while call processing is suspended at DP Call_Accepted respective T_Answer, the T_BCSM for leg p transits to a T_MidCall DP "Wait state" of the Present_Call PIC respective T_Alerting PIC (not of the T_Alerting PIC respective T_Active PIC as answer has not been sent backward on the p-leg). Notice that this Mid_Call DP is internal to the T_BCSM model (no Mid_Call DP reported to the SCF) and therefore not shown in the T_BCSM.</p>		

Table 2/Q.1248.2 – Complete Set of T_BCSM Transitions for the IN CS-4 Model

T-BCSM: Transitions from PIC to PIC/DP		
From	To	Nature of BCSM Transition
T_Null PIC	Termination_Attempt DP	Basic
Authorize_Termination_Attempt PIC	Term_Attempt_Authorized DP Termination_Attempt_Denied DP T_Abandon DP T_Exception PIC	Basic Basic Basic Basic
Select_Facility PIC	Facility_Selected_and_Available DP T_Busy DP T_Abandon DP	Basic Basic Basic
Present_Call PIC	T_No_Answer DP T_Answer DP Call_Accepted DP T_Alerting PIC Select_Facility PIC T_Abandon DP T_Busy DP T_Exception PIC	Basic Basic Basic Basic Basic Basic Basic Basic
T_Alerting PIC	T_Answer DP T_No_Answer DP T_Abandon DP T_MidCall DP (Note) T_Busy DP T_Exception PIC	Basic Basic Basic Extended Basic Basic
T_Active PIC	T_Midcall DP T_Disconnect DP T_Suspend DP T_Exception PIC	Basic Basic Basic Basic
T_Suspended PIC	T_Re-Answer DP T_Disconnect DP T_MidCall DP (T_Active PIC, Note) T_Exception PIC	Basic Basic Extended Basic
T_Exception	T_Null PIC	Basic
NOTE – Notice that this Mid_Call DP may be internal to the T_BCSM model (i.e. no Mid_Call DP reported to the SCF due to a CPH operation) and therefore it may not shown in the T_BCSM.		

5.2.4 BCSM Indications for the CS-4 call model

5.2.4.1 User – O_BCSM Access Signalling Indications (Category 1)

Definition

These Indications include the representation of the network's perception of possible actions taken by the calling party as well as the calling party's perception of actions taken by the network. The Indications are between a user (i.e. calling party) and a local exchange that is originating a call. They include the definition of how actions by the user (originating call model) affect the originating call model (user). These Indications are derived from Access Signalling (e.g. DSS1, analogue) as well as any other information that is available. Figure 8 illustrates these indications.

List of Indications

- (1) An Indication is sent from User to O_BCSM to initiate call establishment (e.g. SETUP).
- (2) An Indication is sent from O_BCSM to User that network is unable to initiate call (e.g. RELEASE_COMPLETE).
- (3) An Indication is sent from O_BCSM to User acknowledging the call initiation Indication (e.g. SETUP_ACKNOWLEDGE).
- (4) The User sends call (dialling) information to the O_BCSM (e.g. INFORMATION).
- (5) An Indication is sent from O_BCSM to the User to terminate the sending of call information (e.g. CALL_PROCEEDING).
- (6) An Indication is sent from the User to the O_BCSM upon completion of call information.
- (7) User is informed that call has been routed to another environment of network (e.g. PROGRESS).
- (8) An Indication is sent from the O_BCSM to the User when the called party is being alerted (e.g. ALERTING, CONNECT).
- (9) An Indication is sent from the O_BCSM to the User when the call is accepted.
- (10) The User acknowledges that the call is accepted.
- (11) The O_BCSM sends an Indication to the User that the called party is unable to accept the call, due to busy condition.
- (12) The O_BCSM sends an Indication to the User since the called party is unable to accept the call, due to no answer condition.
- (13) An Indication is received by the O_BCSM from the User to end the call.
- (14) The O_BCSM indicates to the User that the call is being disconnected.
- (15) The User acknowledges to the O_BCSM that the call is being disconnected.
- (16) An Indication is sent to the user when the connection towards the Called Party is suspended.
- (17) An Indication is sent to the user when the connection towards the Called Party is reconnected.

5.2.4.2 T_BCSM – User Access Signalling Indications (Category 2)

Definition

These Indications include the representation of the network's perception of possible actions taken by the called party as well as the called party's perception of actions taken by the network. The Indications are between a local exchange that is terminating a call and a user (i.e. called party). They include the definition of how actions by the terminating call model (user) affect the user (terminating call model). These Indications are derived from Access Signalling (e.g. DSS1, analogue) as well as any other information that is available. Figure 9 illustrates these Indications.

List of Indications

- (1) An Indication is sent from T_BCSM to the User to terminate the call to an idle facility (e.g. SETUP).
- (2) An Indication is sent from User to T_BCSM indicating that the User cannot accept the call (e.g. RELEASE_COMPLETE).
- (3) An Indication is sent from the User to the T_BCSM when the User determines compatibility with all call characteristics (e.g. SETUP_ACKNOWLEDGE).
- (4) The T_BCSM sends any remaining call information to the User (e.g. INFORMATION).
- (5) An Indication is sent from the T_BCSM to the User upon the sending of sufficient call information.
- (6) An Indication is sent from the User to the T_BCSM upon receipt of sufficient call information (e.g. CALL_PROCEEDING).
- (7) User sends an Indication to the T_BCSM that alerting is taking place (e.g. ALERTING).
- (8) An Indication is sent from the User to the T_BCSM upon acceptance of the incoming call (e.g. CONNECT).
- (9) An Indication is sent from the T_BCSM to the User acknowledging that the call can now be connected.
- (10) An Indication is sent from the User to the T_BCSM that the User suspends the call.
- (11) An Indication is sent from the User to the T_BCSM that the User resumes the call.
- (12) The T_BCSM sends an Indication to the User indicating that the calling party has gone on-hook.
- (13) An Indication is received by the T_BCSM from the User to end the call.
- (14) The T_BCSM indicates to the User that the call is being disconnected.
- (15) The User acknowledges to the T_BCSM that the call is being disconnected.

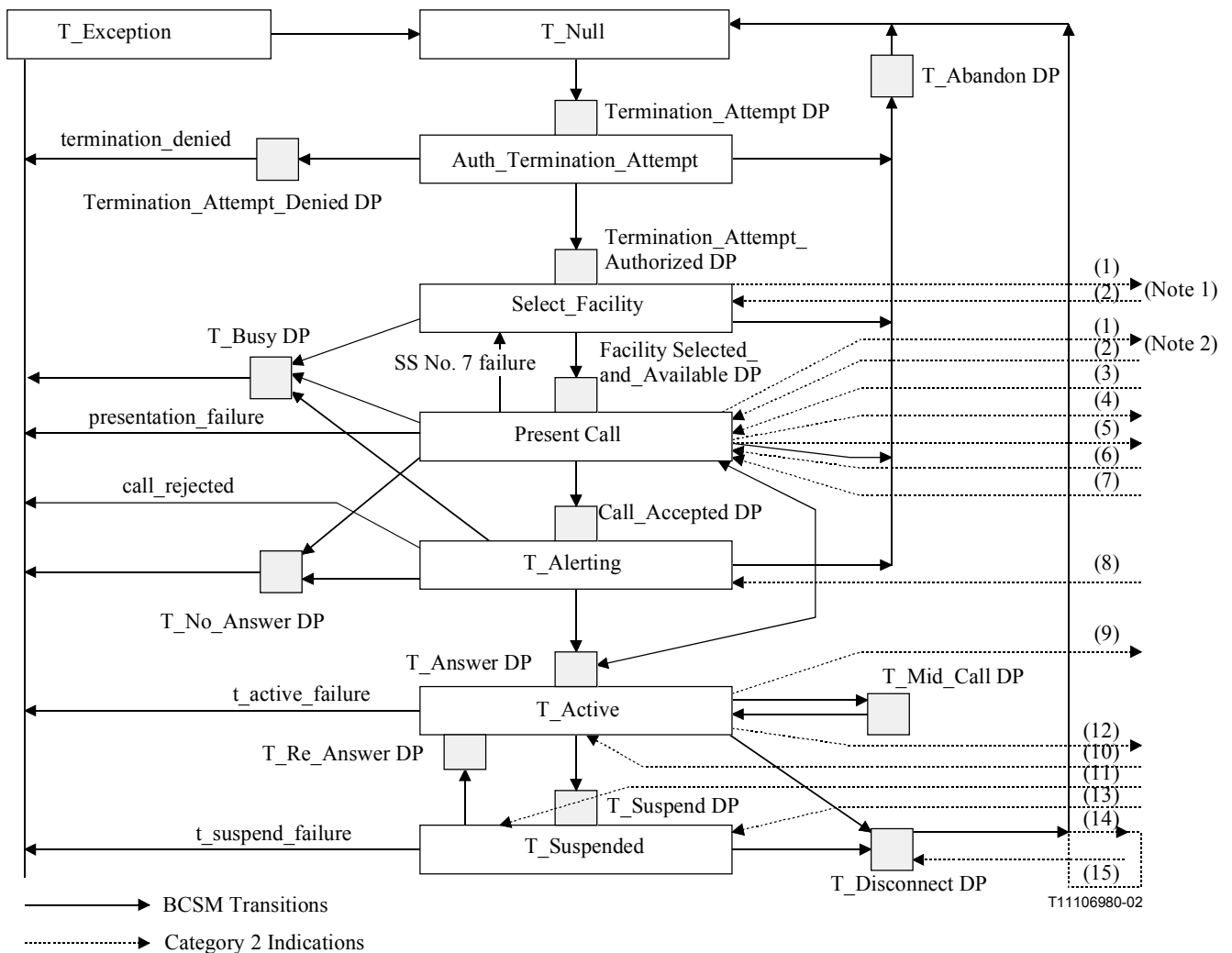


Figure 9/Q.1248.2 – Access Signalling for the BCSM Category 2 (T_BCSM-User)

NOTE – Indications which are shown as terminating on a DP in Figure 9 are received by the switch and are not part of the IN DP processing.

5.2.4.3 Intra Local Exchange BCSM Indications (Category 3)

Figure 10 illustrates the communication between two call segments in the CCF/SSF for a basic two-party call, as described in the CCF/SSF Model. It shows the indications that flow between the originating BCSM and terminating BCSM. All possible indications are shown, except for any which may occur at the O_Exception and the T_Exception PICs. Note that these indications are not intended to be mapped to explicit information flows.

- (1) Initiate T_BCSM after the authority to place the call has been verified and a usable route has been identified. The O_BCSM is currently in the Send_Call PIC. The originating Basic Call Manager has sent the call attempt to the terminating Basic Call Manager for further processing.
- (2) For SS No. 7-supported trunks, if the received IAM indicates a continuity check is required and the resultant continuity check is successful, then an Indication is sent from the O_BCSM to the T_BCSM (causes T_Null PIC to Termination_Attempt DP BCSM transition in the T_BCSM).

- (3) An Indication is sent from the T_BCSM to O_BCSM that the terminating line or trunk is busy. (Causes Send_Call PIC to O_Called_Party_Busy BCSM transition in O_BCSM, if terminating to a line. If terminating to a trunk, then this Indication causes Send_Call PIC to Select_Route PIC BCSM transition in the O_BCSM.)
- (4) An Indication is sent from the T_BCSM to O_BCSM that the terminating line or trunk is busy. (Causes O_Alerting PIC to O_Called_Party_Busy DP BCSM transition in O_BCSM.)
- (5) An Indication is sent from the T_BCSM to O_BCSM that the call cannot be presented. (Causes Send_Call PIC to Select_Route PIC, O_Called_Party_Busy DP, or O_No_Answer DP.)
- (6) An Indication is sent from the T_BCSM to the O_BCSM that an ISDN capable Called Party has signalled call acceptance with immediate BCSM transition to an answered (i.e. CONNect message) condition (causes Send_Call PIC to O_Answer DP BCSM transition in O_BCSM).
- (7) An Indication is sent from T_BCSM to O_BCSM that Called Party is being alerted (causes O_BCSM to transit from Send_Call PIC O_Alerting PIC and prepare to send ring Indication to the Calling Party).
- (8) An Indication is sent from T_BCSM to O_BCSM that Called Party has rejected the call (this is indicated to the O_BCSM with a busy cause and causes O_BCSM to transit from O_Alerting PIC to Select_Route PIC or O_Called_Party_Busy DP).
- (9) An Indication is sent from T_BCSM to O_BCSM that Called Party has not answered within a specified time period (causes O_Alerting PIC to O_No_Answer DP BCSM transition in O_BCSM).
- (10) An Indication is sent from the T_BCSM to the O_BCSM that called party has not answered within a specified time period. (Causes Send_Call PIC to O_No_Answer DP BCSM transition in O_BCSM.)
- (11) An Indication is sent from T_BCSM to O_BCSM that Called Party has accepted and answered the call attempt (causes O_Alerting PIC to O_Answer DP BCSM transition in O_BCSM).
- (12) An Indication is sent from the T_BCSM to the O_BCSM that the called party has accepted and answered the call attempt. (Causes Send_Call PIC to O_Answer DP BCSM transition in O_BCSM.)
- (13) An Indication is sent from T_BCSM to O_BCSM that Called Party has disconnected (e.g. on-hook), (causes O_Active PIC to O_Suspend DP BCSM transition in O_BCSM).
- (14) An Indication is sent from T_BCSM to O_BCSM that Called Party re-answers is received before the timer expires (causes O_Suspended PIC to O_Re_Answer DP BCSM transition in O_BCSM). Note that the name and function of this timer is FFS.
- (15) An Indication is sent from O_BCSM to T_BCSM that Calling Party has disconnected, while T_BCSM was in T_Active PIC (causes T_Active PIC to T_Disconnect DP BCSM transition in T_BCSM).
- (16) An Indication is sent from O_BCSM to T_BCSM that Calling Party has disconnected, while T_BCSM was in T_Suspended PIC (causes T_Suspended PIC to T_Disconnect DP BCSM transition in T_BCSM).
- (17) An Indication is sent from T_BCSM to O_BCSM that Called Party has disconnected (causes O_Suspended PIC to O_Disconnect DP BCSM transition in O_BCSM).
- (18) An Indication is sent from the T_BCSM (T_Disconnect DP) to O_BCSM that the calling party has disconnected. (Causes O_Active PIC to O_Disconnect DP BCSM transition in O_BCSM.)

- (19) An Indication is sent from O_BCSM to T_BCSM that Calling Party has abandoned (causes Authorize_Termination_Attempt PIC, Select_Facility PIC, Present_Call PIC or T_Alerting PIC to T_Abandoned DP BCSM transition in T_BCSM).

NOTE – Indications (15) and (17) are mutually exclusive:

- these indications are for intra-switch;
- the indications do not explicitly include the modelling of SRFs;
- indications which are preceded by a DP may be affected depending on whether the DP is active and the SCF response.

5.2.5 Mapping from Cause to DP

A normative mapping between signalling release cause values and DPs is defined by Mapping Tables, i.e. the Table covers only cause values received by the basic call signalling. The tables are based on the Call States. In some cases the mapping of a particular cause value to a DP depends on the PIC in which the release message is received. This is indicated where appropriate by notes.

This Mapping from Cause value to DP as indicated in Tables 3 and 4 for O_BCSM and respective T_BCSM is normative, except where the table indicates that receipt of the cause value leads to the O/T_Exception PIC. In the latter case, network operators may decide to map some of these cause values to a specific DP.

5.2.5.1 O_BCSM: Mapping Table from Cause to DP

A flexible mapping is provided as the DP Route_Select Failure is used as a possible transition for e.g. the PICs Send_Call and O_Alerting. The transitions are done through an internal transition in the O_BCSM.

The following mapping of O_BCSM PICs is applied:

For ease of reference, the PICs have been categorized as follows:

- Category: **Originating Call setup:**
PICs: Authorize_Origination_Attempt, Collect_Information, Analyse_Information, Select_Route, Authorize_Call_Setup.
- Category: **Originating Stable Call:**
PICs: Send-Call, O_Alerting, O_Active.
- Category: **Originating Call Clearing:**
PICs: O_Suspended.

NOTE 1 – A transition to Route_Select_Failure DP occurs directly when due to the receipt of a route failure event from the called destination (provided that no more CDPNs left to try).

NOTE 2 – Transition valid for O_Active PIC only.

NOTE 3 – The following rule applies for the Originating CallSetup phase, the SendCall and O_Alerting PIC of the Originating Stable call phase: The receipt of any cause value received in forward direction during these call phases shall be mapped to the O_Abandon DP.

Table 3/Q.1248.2 – O_BCSM: Mapping Cause value to DP

No	Reason (see Q.850)	Originating CallSetup	Originating Stable Call		Originating CallClearing
			(Note 1)	(Note 2)	
1	Unallocated (unassigned) number	Route_Select_Failure	Route_Select_Failure	Exception	Exception
2	No route to specified transit network	Route_Select_Failure	Route_Select_Failure	Exception	Exception
3	No route to destination	Route_Select_Failure	Route_Select_Failure	Exception	Exception
4	Send special information tone	Route_Select_Failure	Route_Select_Failure	Exception	Exception
5	Misdialled trunk prefix	Route_Select_Failure	Route_Select_Failure	Exception	Exception
6	Channel unacceptable (Q.931 only)	Exception	Exception	Exception	Exception

Table 3/Q.1248.2 – O_BCSM: Mapping Cause value to DP

No	Reason (see Q.850)	Originating CallSetup	Originating Stable Call		Originating CallClearing
			(Note 1)	(Note 2)	
7	Call awarded and being delivered in an established channel (Q.931 only)	Exception	Exception	Exception	Exception
8	Pre-emption	Route_Select_Failure	Route_Select_Failure	Exception	Exception
9	Pre-emption – Circuit reserved for reuse	Route_Select_Failure	Route_Select_Failure	Exception	Exception
14	Ported subscriber	Route_Select_Failure	Route_Select_Failure	Exception	Exception
16	Normal call clearing received in backward direction	Route_Select_Failure	Route_Select_Failure	O_Disconnect	O_Disconnect
17	User busy	Exception	O_Called_Party_Busy	Exception	Exception
18	No user responding	Exception	O_No_Answer	Exception	Exception
19	No answer from user (user alerted)	Exception	O_No_Answer	Exception	Exception
20	Subscriber absent	Exception	O_Called_Party_Busy	Exception	Exception
21	Call rejected	Route_Select_Failure	Route_Select_Failure	Exception	Exception
22	Number changed	Route_Select_Failure	Route_Select_Failure	Exception	Exception
26	Non-selected user clearing (Q.931 only)	Exception	Exception	Exception	Exception
27	Destination out of order	Route_Select_Failure	Route_Select_Failure	Exception	Exception
28	Invalid number format (address incomplete)	Route_Select_Failure	Route_Select_Failure	Exception	Exception
29	Facility rejected	Route_Select_Failure	Route_Select_Failure	Exception	Exception
30	Response to STATUS ENQUIRY (Q.931 only)	Exception	Exception	Exception	Exception
31	Normal, unspecified	Route_Select_Failure (backward direction) (Note 3) O_Abandon (forward direction)	Route_Select_Failure (backward direction) (Note 3) O_Abandon (forward direction)	O_Disconnect	O_Disconnect
34	No circuit/channel available	Exception	O_Called_Party_Busy	Exception	Exception
35	Requested VPCI/VCI not available	Exception	O_Called_Party_Busy	Exception	Exception
36	VPCI/VCI assignment failure	Route_Select_Failure	Route_Select_Failure	Exception	Exception
37	User cell rate not available	Route_Select_Failure	Route_Select_Failure	Exception	Exception

Table 3/Q.1248.2 – O_BCSM: Mapping Cause value to DP

No	Reason (see Q.850)	Originating CallSetup	Originating Stable Call		Originating CallClearing
			(Note 1)	(Note 2)	
38	Network out of order	Route_Select_Failure	Route_Select_Failure	Exception	Exception
39	Permanent frame mode connection out of service (Q.931 only)	Exception	Exception	Exception	Exception
40	Permanent frame mode connection operational (Q.931 only)	Exception	Exception	Exception	Exception
41	Temporary failure	Route_Select_Failure	Route_Select_Failure	Exception	Exception
42	Switching equipment congestion	Route_Select_Failure	Route_Select_Failure	Exception	Exception
43	Access information discarded	Route_Select_Failure	Route_Select_Failure	O_Disconnect	Disconnect
44	Requested circuit/channel not available	Exception	O_Called_Party_Busy	Exception	Exception
45	No VPCI/VCI available	Exception	O_Called_Party_Busy	Exception	Exception
46	Precedence call blocked	Route_Select_Failure	Route_Select_Failure	Exception	Exception
47	Resource unavailable, unspecified	Route_Select_Failure	Route_Select_Failure	Exception	Exception
49	Quality of service unavailable	Route_Select_Failure	Route_Select_Failure	Exception	Exception
50	Requested facility not subscribed	Route_Select_Failure	Route_Select_Failure	Exception	Exception
53	Outgoing calls barred within CUG	Route_Select_Failure	Route_Select_Failure	Exception	Exception
55	Incoming calls barred within CUG	Route_Select_Failure	Route_Select_Failure	Exception	Exception
57	Bearer capability not authorized	Route_Select_Failure	Route_Select_Failure	Exception	Exception
58	Bearer capability not presently available	Route_Select_Failure	Route_Select_Failure	Exception	Exception
62	Inconsistency in designated outgoing access information and subscriber class	Exception	Exception	Exception	Exception
63	Service or option not available, unspecified	Route_Select_Failure	Route_Select_Failure	Exception	Exception
65	Bearer capability not implemented	Route_Select_Failure	Route_Select_Failure	Exception	Exception

Table 3/Q.1248.2 – O_BCSM: Mapping Cause value to DP

No	Reason (see Q.850)	Originating CallSetup	Originating Stable Call		Originating CallClearing
			(Note 1)	(Note 2)	
66	Channel type not implemented (Q.931 only)	Exception	Exception	Exception	Exception
69	Requested facility not implemented	Exception	Exception	Exception	Exception
70	Only restricted digital information bearer capability is available	Route_Select_Failure	Route_Select_Failure	Exception	Exception
73	Unsupported combination of traffic parameters	Route_Select_Failure	Route_Select_Failure	Exception	Exception
79	Service or option not implemented, unspecified	Route_Select_Failure	Route_Select_Failure	Exception	Exception
81	Invalid call reference value (Q.931 only)	Exception	Exception	Exception	Exception
82	Identified channel does not exist (Q.931 only)	Exception	Exception	Exception	Exception
83	A suspended call exists, but this call identity does not (Q.931 only)	Exception	Exception	Exception	Exception
84	Call identity in use (Q.931 only)	Exception	Exception	Exception	Exception
85	No call suspended (Q.931 only)	Exception	Exception	Exception	Exception
86	Call having the requested call identity has been cleared (Q.931 only)	Exception	Exception	Exception	Exception
87	User not member of CUG	Route_Select_Failure	Route_Select_Failure	Exception	Exception
88	Incompatible destination	Route_Select_Failure	Route_Select_Failure	Exception	Exception
90	Non-existent CUG	Route_Select_Failure	Route_Select_Failure	Exception	Exception
91	Invalid transit network selection	Route_Select_Failure	Route_Select_Failure	Exception	Exception
93	AAL parameters cannot be supported	Route_Select_Failure	Route_Select_Failure	Exception	Exception
95	Invalid message, unspecified	Exception	Exception	Exception	Exception
96	Mandatory information element is missing	Exception	Exception	Exception	Exception

Table 3/Q.1248.2 – O_BCSM: Mapping Cause value to DP

No	Reason (see Q.850)	Originating CallSetup	Originating Stable Call		Originating CallClearing
			(Note 1)	(Note 2)	
97	Message type non-existent or not implemented	Exception	Exception	Exception	Exception
98	Message not compatible with call state or message type non-existent or not implemented	Exception	Exception	Exception	Exception
99	Information element/parameter non-existent or not implemented	Exception	Exception	Exception	Exception
100	Invalid information element contents	Exception	Exception	Exception	Exception
102	Recovery on timer expire	Exception	Exception	Exception	Exception
103	Parameter non-existent or not implemented, passed on	Exception	Exception	Exception	Exception
110	Message with unrecognized parameter, discarded	Exception	Exception	Exception	Exception
111	Protocol error, unspecified	Exception	Exception	Exception	Exception
127	Interworking, unspecified	Route_Select_Failure	Route_Select_Failure	Exception	Exception

5.2.5.2 T_BCSM: Mapping Table from Cause to DP

The transitions are done through an internal transition in the BCSM.

The following mapping of T_BCSM PICs is applied:

For ease of reference, the PICs have been categorized as follows:

- Category: **Terminating Call setup**:
PICs: Authorize_termination_Attempt, Select_Facility Present_Call.
- Category: **Terminating Active Call**:
PICs: T_Alerting, T_Active.
- Category: **Terminating Call Clearing**:
PICs: T_Suspended.

NOTE 1 – The transition to the indicated DP is valid only for Present_Call PIC.

NOTE 2 – The transition to the indicated DP is valid except for the Authorized_Termination_Attempt PIC.

NOTE 3 – The following rule applies for the stable call phase: The receipt of any cause value received during the stable call phase (O_Active/T_Active) shall in such states be mapped to O/T Disconnect DP.

NOTE 4 – The following rule applies for the Terminating CallSetup phase and the T_Alerting PIC of the Terminating Active Call phase: The receipt of any cause value received in forward direction during these call phases shall be mapped to the T_Abandon DP.

Table 4/Q.1248.2 – T_BCSM: Mapping Cause value to DP

No.	Reason (see Q.850)	Terminating CallSetup	Terminating Stable Call		Terminating Call Clearing
			T_Alerting	T_Active	
1	Unallocated (unassigned) number	Exception	Exception	Exception	Exception
2	No route to specified transit network	Exception	Exception	Exception	Exception
3	No route to destination	Exception	Exception	Exception	Exception
4	Send special information tone	Exception	Exception	Exception	Exception
5	Misdialled trunk prefix	Exception	Exception	Exception	Exception
6	Channel unacceptable (Q.931 only)	Exception	Exception	Exception	Exception
7	Call awarded and being delivered in an established channel (Q.931 only)	Exception	Exception	Exception	Exception
8	Pre-emption	Exception	Exception	Exception	Exception
9	Pre-emption – Circuit reserved for reuse	Exception	Exception	Exception	Exception
14	Ported subscriber	Exception	Exception	Exception	Exception
16	Normal call clearing received in backward direction	Exception (backward direction) (Note 4) T_Abandon (forward direction)	Exception (backward direction) (Note 4) T_Abandon (forward direction)	T_Disconnect	T_Disconnect
17	User busy	T_Busy (Note 2)	T_Busy	Exception	Exception
18	No user responding	T_No_Answer (Note 1)	Exception	Exception	Exception
19	No answer from user (user alerted)	Exception	T_No_Answer	Exception	Exception
20	Subscriber absent	T_Busy (Note 2)	T_Busy	Exception	Exception
21	Call rejected	Exception	Exception	Exception	Exception
22	Number changed	Exception	Exception	Exception	Exception
26	Non-selected user clearing (Q.931 only)	Exception	Exception	Exception	Exception
27	Destination out of order	Exception	Exception	Exception	Exception
28	Invalid number format (address incomplete)	Exception	Exception	Exception	Exception
29	Facility rejected	Exception	Exception	Exception	Exception
30	Response to STATUS ENQUIRY (Q.931 only)	Exception	Exception	Exception	Exception

Table 4/Q.1248.2 – T_BCSM: Mapping Cause value to DP

No.	Reason (see Q.850)	Terminating CallSetup	Terminating Stable Call		Terminating Call Clearing
			T_Alerting	T_Active	
31	Normal, unspecified	Exception (backward direction) T_Abandon (forward direction)	Exception (backward direction) T_Abandon (forward direction)	T_Disconnect	T_Disconnect
34	No circuit/channel available	T_Busy (Note 2)	T_Busy	Exception	Exception
35	Requested VPCI/VCI not available	T_Busy (Note 2)	T_Busy	Exception	Exception
36	VPCI/VCI assignment failure	Exception	Exception	Exception	Exception
37	User cell rate not available	Exception	Exception	Exception	Exception
38	Network out of order	Exception	Exception	Exception	Exception
39	Permanent frame mode connection out of service (Q.931 only)	Exception	Exception	Exception	Exception
40	Permanent frame mode connection operational (Q.931 only)	Exception	Exception	Exception	Exception
41	Temporary failure	Exception	Exception	Exception	Exception
42	Switching equipment congestion	Exception	Exception	Exception	Exception
43	Access information discarded	Exception	Exception	T_Disconnect	T_Disconnect
44	Requested circuit/channel not available	T_Busy (Note 2)	T_Busy	Exception	Exception
45	No VPCI/VCI available	T_Busy (Note 2)	T_Busy	Exception	Exception
46	Precedence call blocked	Exception	Exception	Exception	Exception
47	Resource unavailable, unspecified	Exception	Exception	Exception	Exception
49	Quality of service unavailable	Exception	Exception	Exception	Exception
50	Requested facility not subscribed	Exception	Exception	Exception	Exception
53	Outgoing calls barred within CUG	Exception	Exception	Exception	Exception
55	Incoming calls barred within CUG	Exception	Exception	Exception	Exception
57	Bearer capability not authorized	Exception	Exception	Exception	Exception
58	Bearer capability not presently available	Exception	Exception	Exception	Exception
62	Inconsistency in designated outgoing access information and subscriber class	Exception	Exception	Exception	Exception

Table 4/Q.1248.2 – T_BCSM: Mapping Cause value to DP

No.	Reason (see Q.850)	Terminating CallSetup	Terminating Stable Call		Terminating Call Clearing
			T_Alerting	T_Active	
63	Service or option not available, unspecified	Exception	Exception	Exception	Exception
65	Bearer capability not implemented	Exception	Exception	Exception	Exception
66	Channel type not implemented (Q.931 only)	Exception	Exception	Exception	Exception.
69	Requested facility not implemented	Exception	Exception	Exception	Exception
70	Only restricted digital information bearer capability is available	Exception	Exception	Exception	Exception
73	Unsupported combination of traffic parameters	Exception	Exception	Exception	Exception
79	Service or option not implemented, unspecified	Exception	Exception	Exception	Exception
81	Invalid call reference value (Q.931 only)	Exception	Exception	Exception	Exception
82	Identified channel does not exist (Q.931 only)	Exception	Exception	Exception	Exception
83	A suspended call exists, but this call identity does not (Q.931 only)	Exception	Exception	Exception	Exception
84	Call identity in use (Q.931 only)	Exception	Exception	Exception	Exception
85	No call suspended (Q.931 only)	Exception.	Exception	Exception	Exception
86	Call having the requested call identity has been cleared (Q.931 only)	Exception	Exception	Exception	Exception
87	User not member of CUG	Exception	Exception	Exception	Exception
88	Incompatible destination	Exception	Exception	Exception	Exception
90	Non-existent CUG	Exception	Exception	Exception	Exception
91	Invalid transit network selection	Exception	Exception	Exception	Exception
93	AAL parameters cannot be supported	Exception	Exception	Exception	Exception
95	Invalid message, unspecified	Exception	Exception	Exception	Exception
96	Mandatory information element is missing	Exception	Exception	Exception	Exception
97	Message type non-existent or not implemented	Exception	Exception	Exception	Exception
98	Message not compatible with call state or message type non-existent or not implemented	Exception	Exception	Exception	Exception

Table 4/Q.1248.2 – T_BCSM: Mapping Cause value to DP

No.	Reason (see Q.850)	Terminating CallSetup	Terminating Stable Call		Terminating Call Clearing
			T_Alerting	T_Active	
99	Information element /parameter non-existent or not implemented	Exception	Exception	Exception	Exception
100	Invalid information element contents	Exception	Exception	Exception	Exception
102	Recovery on timer expire	Exception	Exception	Exception	Exception
103	Parameter non-existent or not implemented, passed on	Exception	Exception	Exception	Exception
110	Message with unrecognized parameter, discarded	Exception	Exception	Exception	Exception
111	Protocol error, unspecified	Exception	Exception	Exception	Exception
127	Interworking, unspecified	Exception	Exception	Exception	Exception

5.2.6 Signalling Terminations

5.2.6.1 Signalling Termination modelling

Signalling Terminations provide adaptation functionalities between the Basic Call Controller and signalling interfaces. Different types of Signalling Terminations may be available in a CCF. This depends on the list of signalling protocols supported by the exchange in which the CCF resides.

Each Signalling Termination embodies the knowledge of the signalling procedures defined for the supported protocol.

Signalling Termination instances are created when an external event is received or is to be sent, to initiate call establishment. In the latter case, the Basic Call Controller creates them.

Signalling Termination instances communicates with the BCSM Controller Process, using a set of abstract signalling primitives defined in 5.2.4.

In addition to those supporting a particular signalling protocol, there are also specific types of Signalling Terminations that are used to temporarily store signalling information when no signalling link is attached to a BCSM or to relay signalling information between BCSM instances.

With regards to the Signalling Terminations, the Basic Call Controller performs the following actions:

- Receives abstract primitives from the BCSM instances and transmits them to the appropriate Signalling Termination.
- Receives abstract primitives from the Signalling Terminations and transmits them to the appropriate BCSM instance or to the SSF (charging events, USI related events, ...)

Signalling Termination instances use a Finite State Machine (FSM) to determine whether an abstract signalling primitive, received from the BCSM Controller should be ignored or mapped to an appropriate signalling message.

Example: In case a BCSM sends a "Setup.Resp" abstract primitive to an ISUP Termination, the FSM state would determine whether this signal has to be mapped to an ANM message or ignored (because the FSM is already in the state "Confirmed Path" due to a follow-on call configuration).

5.2.6.2 Abstract Signalling Primitives

The communication between the BCSM instances and the Signalling Terminations uses a set of Abstract Signalling primitives defined in the following paragraphs. Similar primitives are also used for inter-BCSM communication over the Inter-BCSM Interface (IBI).

The communication between the BCSM instances and the Signalling Termination uses a set of abstract signalling primitives. Similar primitives are also used for inter-BCSM communication (i.e. over the IBI interface). These primitives are derived from the set of abstract services described below, using standards conventions. As shown in Figure 11, these conventions are extended for use over the IBI.

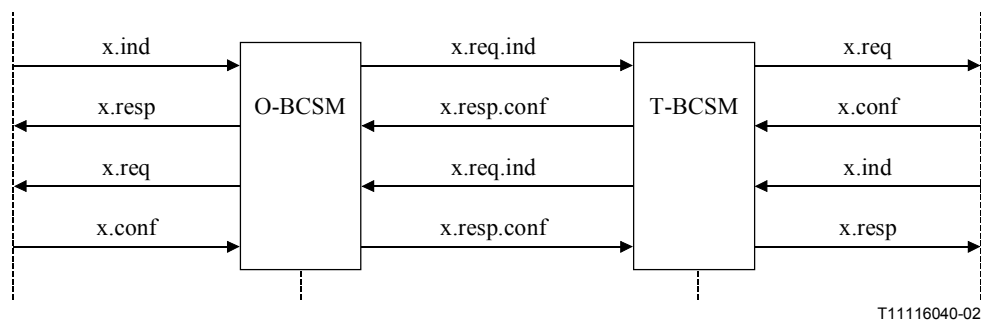


Figure 11/Q.1248.2 – Conventions for Abstract Signalling Primitives

Setup

The Setup service is used to request establishment of a call connection. This is a confirmed service, i.e. a response confirmation Setup primitive is used to confirm that the connection has been established. The request for establishment of a connection can be originated by either the user or the network (i.e. SCF).

Release

The Release service is used to notify that a user has disconnected from the connection and cannot be connected and to request disconnection of a call connection. This is an unconfirmed service.

SubsequentAddress

The SubsequentAddress service is used for conveying subsequent address information during the digit-by-digit methods of call setup, and for conveying information about last digit received, i.e. address end during the digit-by-digit methods of call setup. This is an unconfirmed service.

CallProgress

The CallProgress service is used to report status and/or other types of call information across the network. The type of information is indicated (e.g. "no indication", "alerting", "remote call hold" etc.). This is an unconfirmed service.

NetworkSuspend

The NetworkSuspend service is a signal used to suspend the call on behalf of the called party upon receipt of an on-hook indication from the terminating line or upon receipt of a network suspend message indication from terminating side. This is an unconfirmed service.

NetworkResume

NetworkResume service is used to resume the call on behalf of the called party upon receipt of a re-answer indication from the terminating line as the subscriber goes off-hook or upon receipt of a network resume message indication from terminating side. This is an unconfirmed service.

Failure

The Failure service is used to report the occurrence of a failure in the network

ServiceFeatureIndication

The ServiceFeatureIndication service is used to report the occurrence of a service feature activation request from user

ChargingEvent

The ChargingEvent service is used to report the occurrence of a charging event.

Data

The Data service is used to notify that a user has provided data and to request the transmission of data on a signalling connection. This is an unconfirmed bi-directional service.

5.2.6.3 FSM for Signalling Terminations

5.2.6.3.1 Overview

Figure 12 provides an overview of the states and transitions of the FSM for Call Control Signalling Terminations. These states and transitions are more precisely defined in the following subclauses.

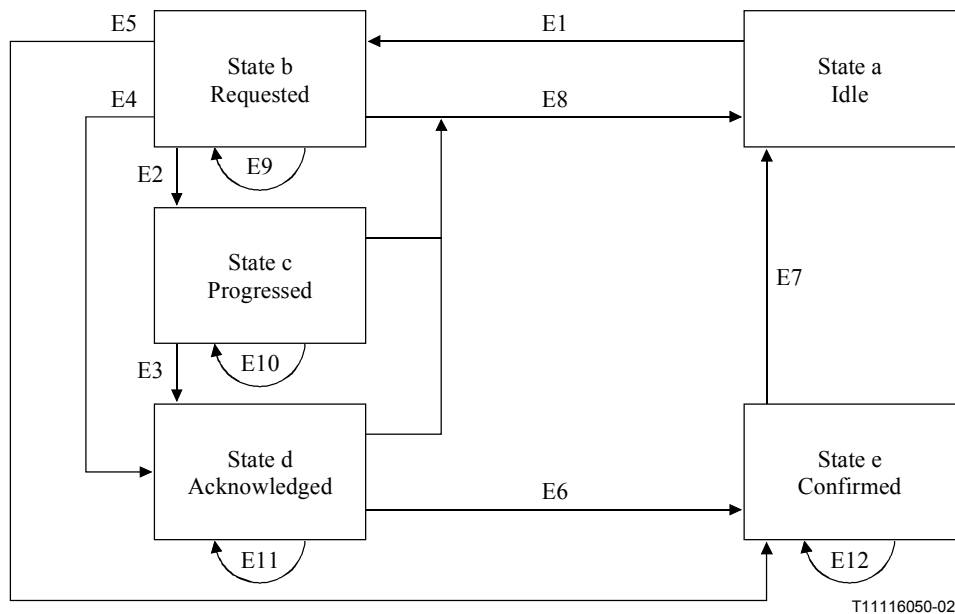


Figure 12/Q.1248.2 – FSM for Call Control Signalling Termination

5.2.6.3.2 FSM states

5.2.6.3.2.1 Idle

The termination is not engaged in any call.

5.2.6.3.2.1.1 Entry events

See exit events from all other states.

The FSM instance is deleted when the release procedure has been completed.

5.2.6.3.2.1.2 Exit Events

- To the Requested Path state (E1):
 - Setup.Ind sent to an O_BCSM
 - Setup.Req received from a T_BCSM

5.2.6.3.2.2 Requested Path

Set-up request has been sent or received but not acknowledged nor confirmed.

5.2.6.3.2.2.1 Entry Event

- Setup.Ind sent to an O_BCSM
- Setup.Req received from a T_BCSM

5.2.6.3.2.2.2 Exit Event

- To the Idle state (E8):
 - Release.Ind sent to an O_BCSM or a T_BCSM
 - Release.Req received from an O_BCSM or a T_BCSM
 - Failure.Ind sent to an O_BCSM or a T_BCSM
- To the Progressed Path state (E2):
 - CallProgress.Ind sent to a T_BCSM, with no alerting indication
 - CallProgress.Req received from an O_BCSM with no alerting indication
- To the Acknowledged Path state (E4):
 - CallProgress.Ind sent to a T_BCSM, with alerting indication
 - CallProgress.Req received from a O_BCSM with alerting indication
- To the Confirmed Path state (E5):
 - Setup.Conf sent to a T_BCSM
 - Setup.Resp received from an O_BCSM

The following primitive may be received or sent without causing any state transition (E9):

- SubsequentAddress

5.2.6.3.2.3 Progressed Path

Call progress information sent or received.

5.2.6.3.2.3.1 Entry Event

See exit events from Requested Path.

5.2.6.3.2.3.2 Exit Event

- To the Idle state (E8):
 - Release.Ind sent to an O_BCSM or a T_BCSM
 - Release.Req received from an O_BCSM or a T_BCSM
 - Failure.Ind sent to an O_BCSM or a T_BCSM
- To the Acknowledged Path state (E3):
 - CallProgress.Ind sent to a T_BCSM, with alerting indication
 - CallProgress.Req received from a O_BCSM with alerting indication

The following primitives may be received or sent without causing any state transition (E10):

- SubsequentAddress
- ServiceFeatureIndication
- ChargingEvent
- Data

5.2.6.3.2.4 Acknowledged Path

Alerting information sent or received

5.2.6.3.2.4.1 Entry Event

See exist events from Progressed (E3) and Requested (E4).

5.2.6.3.2.4.2 Exit Event

- To the Idle state (E8):
 - Release.Ind sent to an O_BCSM or a T_BCSM
 - Release.Req received from an O_BCSM or a T_BCSM
 - Failure.Ind sent to an O_BCSM or a T_BCSM
- To the Confirmed Path state (E6):
 - Setup.Conf sent to a T_BCSM
 - Setup.Resp received from an O_BCSM

The following primitives may be received or sent without causing any state transition (E11):

- SubsequentAddress
- ServiceFeatureIndication
- ChargingEvent
- Data

5.2.6.3.2.5 Confirmed Path

Set-up conformation sent or received

5.2.6.3.2.5.1 Entry Event

See exit events from the Requested state (E5) and Acknowledged state (E6)

5.2.6.3.2.5.2 Exit Event

- To the Idle state (E8):
 - Release.Ind sent to an O_BCSM or a T_BCSM
 - Release.Req received from an O_BCSM or a T_BCSM
 - Failure.Ind sent to an O_BCSM or a T_BCSM

The following primitives may be received or sent without causing any state transition (E12):

- SubsequentAddress
- ServiceFeatureIndication
- ChargingEvent
- Data
- NetworkSuspend
- NetworkResume

5.2.6.4 Examples of mappings

The following subclauses provide examples of possible mappings between Abstract Signalling Primitives and signalling messages, based on the state of the Signalling FSM. These examples are intended to facilitate the understanding of the behaviour and the role of Signalling Terminations. The actual inter-working between IN and other signalling systems is outside the scope of this Recommendation.

5.2.6.4.1 Mapping to ISUP

Table 5 illustrates a possible mapping performed by an ISUP Signalling Termination.

Table 5/Q.1248.2 – Mapping to ISUP

	Idle	Requested	Progressed	Acknowledged	Confirmed
Setup.Req	IAM Requested	/	/	/	/
Call Progress.Req (no alerting)	/	CPG Progressed	CPG Same State	CPG Same state	Discard or CPG Same State
Call Progress.Req (alerting)	/	ACM Acknowledged	ACM Acknowledged	Discard Same State	Discard or CPG Same State
Setup.Resp	/	ANM Confirmed	CON Same State	ANM Confirmed	Discard or CPG Same State
Release.Req	/	REL Idle	REL Idle	REL Idle	REL Idle

5.2.6.4.2 Mapping to DSS1

Table 6 illustrates a possible mapping performed by an DSS1 Signalling Termination.

Table 6/Q.1248.2 – Mapping to DSS1

	Idle	Requested	Progressed	Acknowledged	Confirmed
Setup.Req	SETUP Requested	/	/	/	/
Call Progress.Req (no alerting)	/	CALL PROCEEDING Progressed	CALL PROCEEDING Same State	Discard Same state	Discard Same State
Call Progress.Req (alerting)	/	ALERTING Acknowledged	ALERTING Acknowledged	Discard Same State	Discard Same State
Setup.Resp	/	CONNECT Confirmed	CONNECT Same State	CONNECT Confirmed	Discard Same State
Release.Req	/	REL Idle	REL Idle	REL Idle	REL Idle

5.2.6.4.3 Mapping to H.225

Table 7 illustrates a possible mapping performed by an H.225 Signalling Termination.

Table 7/Q.1248.2 – Mapping to H.225

	Idle	Requested	Progressed	Acknowledged	Confirmed
Setup.Reg	SETUP Requested	/	/	/	/
Call Progress.Reg (no alerting)	/	CALL PROCEEDING Progressed	CALL PROCEEDING Same State	Discard Same state	Discard Same State
Call Progress.Reg (alerting)	/	ALERTING Acknowledged	ALERTING Acknowledged	Discard Same State	Discard Same State
Setup.Resp	/	CONNECT Confirmed	CONNECT Same State	CONNECT Confirmed	Discard Same State
Release.Reg	/	REL Idle	REL Idle	REL Idle	REL Idle

5.2.6.4.4 Mapping to SIP

Table 8 illustrates a possible mapping performed by an SIP Signalling Termination.

Table 8/Q.1248.2 – Mapping to SIP

	Idle	Requested	Progressed	Acknowledged	Confirmed
Setup.Reg	INVITE Requested	/	/	/	/
Call Progress.Reg (no alerting)	/	100 Trying Progressed	Discard Same State	Discard Same state	Discard Same State
Call Progress.Reg (alerting)	/	180 Ringing Acknowledged	Discard Acknowledged	Discard Same State	Discard Same State
Setup.Resp	/	200 Ok Confirmed	200 Ok Same State	200 Ok Confirmed	Discard Same State
Release.Reg	/	BYE Idle	BYE Idle	BYE Idle	BYE Idle

5.3 Bearer Control Entity

5.3.1 Overview

The Bearer Control Entity (BRC) is responsible for the control of the media resources (e.g. the switch matrix). For each call, the BRC coordinates instructions received from all the involved half-calls and derive the appropriate action on media resources.

Detailed modelling of the BRC is outside the scope of this Recommendation. IN CS-4 assumes that each call includes a single bearer connection at a time. The BRC may have its own Signalling Terminations or use information embedded in call control signalling messages.

NOTE 1 – The above restriction may prevent using CPH operations when multiple connections are associated with the same call (e.g. multiple logical channels in a associated with a single H.323 call).

IN CS-4 does not provide IN service logic instances with a direct control on bearer control procedures. It provides service logic instances with a limited visibility on the state of bearer connections. A tied coupling is assumed between the call state and the underlying state of bearer connection (e.g. bearer connections are considered in the active phase as soon as the O_Answer DP has been encountered). However, modification of the bearer connection during the active state can be reported to IN service logic instances, using a simple bearer state model.

NOTE 2 – The above restriction may prevent using CPH operations when the call and connection signalling procedures are independent (e.g. in an H.323 environment without using the fast connect facility).

5.3.2 Bearer State Model

For B-ISDN, the BCSM creates a subordinate process to model the status of the bearer connection. The subordinate process is created in the O/T_Bearer_Active PIC, when the BCSM encounters the O/T_Answer DP. The subordinate process is removed when the O_BCSM transitions to the O/T_Disconnect DP, or on any transition of the O/T_BCSM to exception. The O_BCSM subordinate process must also be removed on any transition from O_Mid_Call DP to any PIC other than O_Active or O_Suspend.

The subordinate process evolution is driven by information received from the Signalling Terminations of the Basic Call Control Entity (e.g. B-ISDN Modify Request) or from the Signalling Terminations of the Bearer Control Entity (e.g. H.245 Signalling Terminations) .

5.3.2.1.1 O_BCSM Subordinate Process

5.3.2.1.2 O_Bearer_Active

Entry event: Request from O_BCSM to create subordinate process on transition of call to O_Active state.

Function

- Both call and bearer are monitored in the active state.

Exit events

- Indication of desire to update (modify) the connection characteristics is received (e.g. reception of a MODIFY request from originating user) (DP: O_Modify_Request).
- Deletion Request from the O_BCSM for, for example, one of the following reasons (subordinated process deleted):
 - An exception event is encountered in the O_BCSM;
 - Connection released in the O_BCSM by remote party;
 - Connection released in the O_BCSM by originating party;
 - Transition from O_Mid Call to any PIC other than O_Active or O_Suspend.
- Exception (subordinated process deleted).

5.3.2.1.3 O_Bearer_Modify

Entry event: Indication of desire to update (modify) the connection characteristics is received (e.g. reception of a MODIFY request from originating user) (DP: O_Modify_Request).

Functions

- The authority/ability of the calling party to modify the connection is verified. If updating is impossible or not permitted, the active state is resumed and a message is sent to the originating party (e.g. MODIFY reject).
- If the modification is possible, an appropriate message is sent to the next relay.

- An answer (e.g. MODIFY acknowledgement/confirm/reject) is awaited. If a reject message is obtained, the connection is resumed to the active state, and a message is sent to the originating party (e.g. MODIFY reject). If an acknowledgement message is received, local network resources are modified and the modification is confirmed, both to the originating party and to the other end of the connection (MODIFY confirm).

Exit events

- Connection update completed successfully (DP: O_Modify_Result).
- Connection update unauthorized or unsuccessful (DP: O_Modify_Result).
- Connection modification failure (DP: O_Modify_Result).
- Deletion Request from the O_BCSM for, for example, one of the following reasons (subordinated process deleted):
 - An exception event is encountered in the O_BCSM;
 - Connection released in the O_BCSM by remote party;
 - Connection released in the O_BCSM by originating party;
 - Transition from O_Mid Call to any PIC other than O_Active or O_Suspend.
- Exception (subordinated process deleted).

5.3.2.1.4 T_BCSM Subordinate Process

5.3.2.1.5 T_Bearer_Active

Entry event: Request from T_BCSM to create subordinate process on transition of call to T_Active state.

Function

- Both call and bearer are monitored in the active state.

Exit events

- Indication of desire to update (modify) the connection characteristics is received (e.g. reception of a MODIFY request from originating user) (DP: T_Modify_Request).
- Deletion Request from the T_BCSM for, for example, one of the following reasons (subordinated process deleted):
 - An exception event is encountered in the T_BCSM;
 - Connection released in the T_BCSM by remote party;
 - Connection released in the T_BCSM by originating party;
- Exception (subordinated process deleted).

5.3.2.1.6 T_Bearer_Modify

Entry event: Indication of desire to update (modify) the connection characteristics is received (e.g. reception of a MODIFY request from originating user) (DP: T_Modify_Request).

Functions

- The authority/ability of the calling party to modify the connection is verified. If updating is impossible or not permitted, the active state is resumed and a message is sent to the originating party (e.g. MODIFY reject).
- If the modification is possible, an appropriate message is sent to the next relay.
- An answer (e.g. MODIFY acknowledgement/confirm/reject) is awaited. If a reject message is obtained, the connection is resumed to the active state, and a message is sent to the originating party (e.g. MODIFY reject). If an acknowledgement message is received, local

network resources are modified and the modification is confirmed, both to the originating party and to the other end of the connection (MODIFY confirm).

Exit events

- Connection update completed successfully (DP: T_Modify_Result).
- Connection update unauthorized or unsuccessful (DP: T_Modify_Result).
- Connection modification failure (DP: T_Modify_Result).
- Deletion Request from the T_BCSM for, for example, one of the following reasons (subordinated process deleted):
 - An exception event is encountered in the T_BCSM;
 - Connection released in the T_BCSM by remote party;
 - Connection released in the T_BCSM by originating party;
- Exception (subordinated process deleted).

5.4 Functional Interface between the CCF and the SSF

Within the CCF, the Basic Call Controller is the interface to the SSF. The following subclauses describe the interactions that may occur between these entities.

5.4.1 Interactions from the CCF to the SSF

The following information may be sent from the CCF to the SSF:

- DP reporting.
- BCSM deletion indication.
- Processing errors in response to SSF instructions.
- Report of signalling events that are not related to call control (i.e. receipt of a Data.ind abstract primitive).
- Report of charging events received from a signalling termination (i.e. receipt of ChargingEvent.ind abstract primitive).
- Report of charging events resulting from the local charging GENERation function.

5.4.2 Interactions from the SSF to the CCF

The following information may be sent from the SSF to the SCF:

- Request for creation of a BCSM instance.
- Request for deletion of a BCSM instance.
- Instructions to the charging GENERation function.
- Instructions to resume call processing in sequence with the current DP (PICResume).
- Instructions to resume call processing at a particular PIC (PICInit).
- Instructions to the signalling terminations requesting the sending of messages that are not related to call control.

6 SSF Function Model Components

6.1 DP Criteria

Clause 6.3.7/Q.1238.2 is applicable with the following modification:

Table 6/Q.1238.2 is modified as follows:

Table 9/Q.1248.2 – Originating DP criteria

O_BCSM	Originating DP															
DP Criteria	ARF	OA	OAD	OAA	CI	AI	RSF	OCPB	ONA	OTS	OAns	OMC	OS	ORA	OD	OAb
Trigger Assigned	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Class of Service	O	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O
Specific Calling Party Number (Note 4)	O	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O
Bearer Capability (Note 5)	O	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O
Specific B-channel Identifier	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
Specific Digit String (Note 1)	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O
Feature Code (Note 1)	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O
Prefixes (Note 1)	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O
Access Codes (Note 1)	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O
Called Party Number (Note 1)	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O
Specific abbreviated dialling string (Note 1)	O	O	O	O	O	X	O	O	O	O	O	O	O	O	O	O
Nature of Address	O	–	–	–	–	X	O	O	O	O	O	O	O	O	O	O

Table 9/Q.1248.2 – Originating DP criteria

O_BCSM	Originating DP															
DP Criteria	ARF	OA	OAD	OAA	CI	AI	RSF	OCPB	ONA	OTS	OAns	OMC	OS	ORA	OD	OAb
Feature Activation (Note 3)	X	–	–	–	–	X	X	X	X	X	X	X	X	X	X	X
Facility Information (Note 2)	–	–	–	–	–	X	–	–	–	X	X	X	–	–	–	–
Cause	–	–	–	–	–	–	X	X	X	–	–	–	–	–	X	X
USIService-Indicator	O	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O
X Applicable – Not applicable O Optional																

The DPs in Table 9 "Originating DP Criteria" are abbreviated as follows:

ARF	Authorize_Route_Failure
OA	Origination_Attempt
OAD	Origination_Attempt_Denied
OAA	Origination_Attempt_Authorized
CI	Collected_Info
AI	Analysed_Info
RSF	Route_Select_Failure
OCPB	O_Called_Party_Busy
ONA	O_No_Answer
OTS	O_Term_Seized
OAns	O_Answer
OMC	O_Mid_Call
OS	O_Suspend
ORA	O_Re_Answer
OD	O_Disconnect
OAb	O_Abandon

Table 7/Q.1238.2 is replaced by the following Table 10:

Table 10/Q.1248.2 – Terminating DP criteria

T_BCSM	Terminating DP												
DP Criteria	TA	TAD	TAA	TB	FSA	TNA	CA	TAns	TMC	TS	TRA	TD	Tab
Trigger Assigned	X	X	X	X	X	X	X	X	X	X	X	X	X
Class of Service	X	X	O	O	O	O	O	O	O	O	O	O	O
Specific Calling Party Number (Note 4)	X	X	O	O	O	O	O	O	O	O	O	O	O
Bearer Capability (Note 5)	O	O	O	O	O	O	O	O	O	O	O	O	O
Specific B-channel Identifier	–	–	–	–	O	O	O	O	O	O	O	O	O
Specific Digit String (Note 1)	O	O	O	O	O	O	O	O	O	O	O	O	O
Feature Code (Note 1)	–	–	–	–	–	–	–	–	–	–	–	–	–
Prefixes (Note 1)	–	–	–	–	–	–	–	–	–	–	–	–	–
Access Codes (Note 1)	–	–	–	–	–	–	–	–	–	–	–	–	–
Called Party Number (Note 1)	O	O	O	O	O	O	O	O	O	O	O	O	O
Specific abbreviated dialling string (Note 1)	–	–	–	–	–	–	–	–	–	–	–	–	–
Nature of Address	O	O	O	O	O	O	O	O	O	O	O	O	O
Feature Activation (Note 3)	–	–	–	–	–	X	X	X	X	X	X	X	X
Facility Information (Note 2)	–	–	–	–	–	–	X	X	X	–	–	–	–
Cause	–	–	–	X	–	X	–	–	–	–	–	X	X
USIService-Indicator	X	X	O	O	O	O	O	O	O	O	O	O	O
X Applicable – Not applicable O Optional													

The DPs in Table 10 "Terminating DP Criteria", are abbreviated as follows:

TA	Termination_Attempt
TAA	Termination_Attempt_Authorized
TAD	Termination_Attempt_Denied
TB	T_Busy
FSA	Facility_Selected_and_Available
TNA	T_No_Answer
CA	Call_Accepted
TAns	T_Answer
TMC	T_Mid_Call
TS	T_Suspend
TRA	T_Re_Answer
TD	T_Disconnect
Tab	T_Abandon

ITU-T Q.1237 defines additional criteria for B-ISDN environments.

6.2 Trigger Types and Trigger Precedence

Clause 6.3.8/Q.1238.2 is applicable.

ITU-T Q.1237 defines additional criteria for B-ISDN environments.

6.3 Feature interaction manager (FIM) call manager (CM)

Clause 6.4/Q.1238.2 is applicable.

Tables 11 and 12 include extensions to ITU-T Q.1238.2 for the support of the broadband modify capability.

Table 11/Q.1248.2 – O_BCSM: Implicit EDP disarming table

EDP Disarmed	O Modify Request C-Leg	O Modify Request P-Leg	O Modify Result C-Leg	O Modify Result P-Leg
EDP Encountered				
O_Disconnect Controlling Leg	X		X	
O_Disconnect Passive Leg		X		X
O_Modify_Request	X	X		
O_Modify_Result			X	X
X Represents IN CS-4 SSF implicit disarming of EDP.				

Table 12/Q.1248.2 – T_BCSM: Implicit EDP disarming table

EDP Disarmed	T Modify Request C-Leg	T Modify Request P-Leg	T Modify Result C-Leg	T Modify Result P-Leg
EDP Encountered				
T_Disconnect Controlling Leg	X		X	
T_Disconnect Passive Leg		X		X
T_Modify_Request	X	X		
T_Modify_Result			X	X
X Represents IN CS-4 SSF implicit disarming of EDP.				

6.4 IN-Switching Manager

A brief description of the IN-SM is provided in ITU-T Q.1238.1.

The IN-SM centres around the IN-switching state model (IN-SSM) which provides a description of CCF/SSF IN call/connection processing in terms of IN call/connection states. Object-oriented techniques are used to describe the IN-SSM, based on the concepts and principles outlined in Annex B/Q.1204.

The IN-SM supports IN call party handling capabilities.

The IN-SM subjects described in the following subclauses include the IN-SM call party handling capabilities, IN-SSM, IN-SSM events that can be reported to active IN service logic instances, and SSF resource control.

A high-level description of these subjects is provided.

6.4.1 IN-switching state model (IN-SSM)

The IN-SSM provides an object-oriented finite state machine description of CCF/SSF IN call/connection processing in terms of IN call/connection states.

It provides a framework for describing the scope of view and control of CCF/SSF activities offered to an SCF. The extent to which the IN-SSM is visible to the SCF is defined by the INAP operations identified between the CCF/SSF and SCF. See clauses 11 and 12 providing the operation procedure description and the associated operation parameter descriptions.

IN call/connection states can be described in terms of the IN-SSM, which defines the set of CCF/SSF objects visible to the SCF.

Each IN-SSM instance provides the SCF with a limited aperture of visibility and influence into CCF/SSF IN call/connection processing. This aperture of visibility and influence is defined by the objects that constitute the IN-SSM. These objects are abstractions of CCF/SSF resources accessible to the SCF.

There can be various types of IN-SSMs, each type defined by the objects that constitute it. For example, a Call Segment Association IN-SSM would contain objects that are abstractions of switching and transmission resources.

This clause focuses on such a Call Segment Association IN-SSM, though it is recognized that other types of IN-SSMs may exist for accessing other types of resources.

There can also be various subtypes of a particular IN-SSM type, each defined by a subset of, or restriction on the use of, the total set of objects in the IN-SSM type.

It is anticipated that IN-SSM subtypes will be identified to align with specific IN capability sets as they are defined.

A Call Segment Association (CSA) instance is created by the Service Switching Function Management Entity-Control (SSME-Control) as further described in clause 8, when an IN service logic instance is invoked that requires IN connection control. It is either created as a result of encountering a TDP in a BCSM that satisfies DP criteria, or is initiated by the SCF independent of encountering TDPs. A CSA instance is destroyed when the SCF informs the SSF that the IN service logic instance is completed, or the CSA should be destroyed. The SSF can also initiate CSA destruction (e.g. during error or abnormal conditions).

The characteristics of CCF/SSF call processing represented by CSA objects for IN CS-4 are described below. These characteristics imply the attributes and functions related to CSA objects, to be reflected in the call processing messages/parameters defined for IN CS-4.

- a) The IN Call Segment Association provides the SCF with an abstract view of an isolated portion of a call managed by a functionally separate portion of the CCF/SSF. This isolated portion of a call is referred to as a "half-call" or *call segment* (see CCF/SSF model in Figure 1/Q.1238.2).

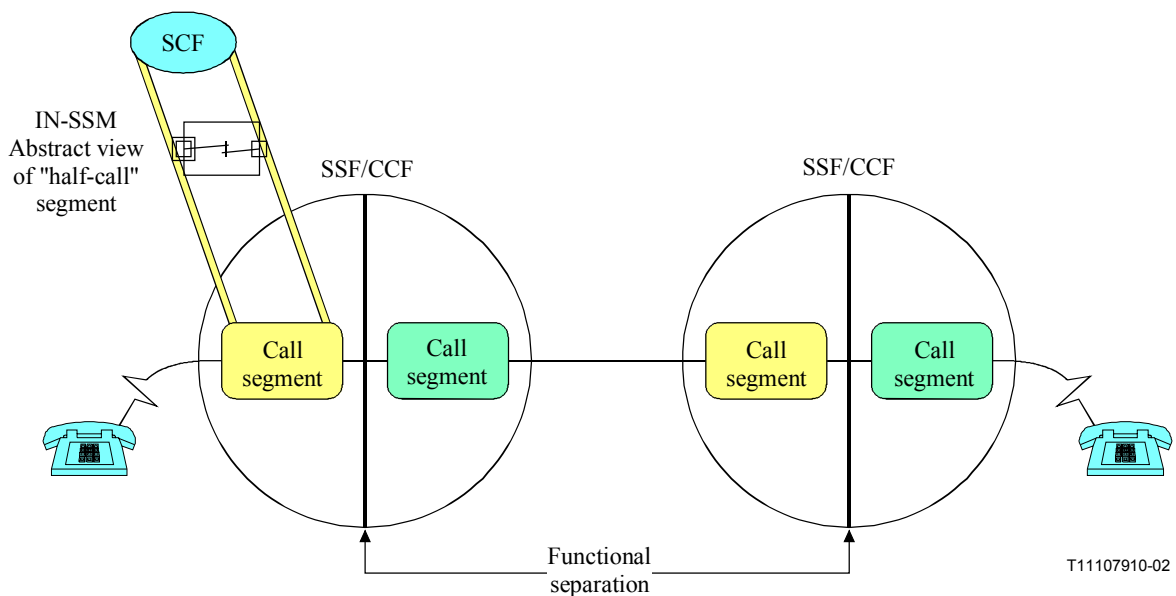


Figure 13/Q.1248.2 – Call segments in a two-party inter-CCF/SSF call

This term "half-call" is used to refer to the physical resources (e.g. connectivity and transmission resources represented by legs and connection points) and to the processes (e.g. basic call processes as modelled by BCSMs) that are involved in the isolated portion of a call.

The SCF does not have direct access to a call segment, but rather has access to the abstract representation of the call segment provided by the Call Segment Association. For IN, access via a Call Segment Association can be a single one-party, two-party or multi-party call segment, or to a pair of associated call segments (see Figure 14). A set of associated call segments are 2 to n call segments that can be related together by the CCF/SSF and manipulated as a set (e.g. to merge them together into a single call segment). For IN CS-4, two call segments can only be associated if both call segments are for the same end user. For example, the CCF/SSF can associate two call segments if the end user is involved in an existing call and would like to originate an additional call, or if the end user is involved in an existing call and there is a new call directed to that end user. This latter example is shown in Figure 14. The extent to which associated Call Segments are visible to the SCF

via a CSA is defined by each Call Segment being represented by a limited number of CSCV states.

- b) A Call Segment Association provides an SCF with an abstract view of a single two-party or multi-party call segment, or of a set of (1 to n) associated call segments. The CSA represents the properties of a call segment or pair of associated call segments of interest to the SCF (e.g. the connectivity and call processing aspects) and describes these properties in terms of objects (i.e. virtual resources) that can be manipulated by the SCF. For connection control, these objects include legs and connection points.

Note that an IN CS-1 connection control IN-SSM is identical to a Call Segment Association (CSA) in terms of connectivity context as defined here and described by the Connection View (CV) Model.

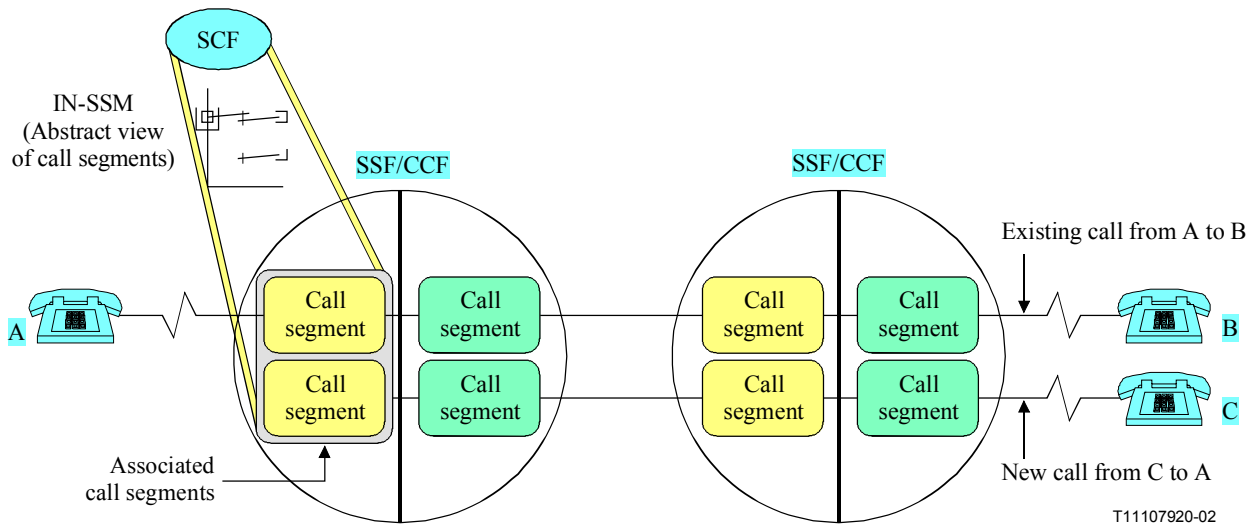


Figure 14/Q.1248.2 – Associated call segments

The call segment concept can be used to describe how the definitions of "single-ended service feature" and "single point of control" apply to the distributed functional plane.

A *single-ended service feature*, as mentioned in ITU-T Q.1231, is described in terms of:

- the scope of control of the service logic instance that realizes the service feature, with respect to the call; and
- the interaction of the service logic instance with respect to other single-ended service logic instances on the same call.

The scope of control of a single-ended service logic instance is restricted to the isolated "half-call(s)" in a CCF/SSF (i.e. the call segments) accessible to the SCF via a control relationship. This is illustrated in Figure 15 below for a two-party call, which shows the BCSMs related to each call segment.

This is also extended for a set of associated "half-calls", or a multi-party "half-call". These scenarios are illustrated in Figures 16 and 17.

All of these scenarios are based on the assumption that "half-calls" can be isolated from their complementary "half-calls" by the functional separation between an originating BCSM instance and its complementary terminating BCSM instance.

A single-ended service logic instance can only directly influence the processing of the isolated "half-call" (or associated "half-calls") in the CCF/SSF.

The other "half-calls" can only be indirectly influenced via information propagating from one "half-call" to another (i.e. between originating and terminating BCSMs, or between BCSMs in different CCF/SSFs).

As such, multiple single-ended service logic instances (one per "half-call") may be simultaneously active on a single call, each isolated from the other by the communication between "half-calls". The communication between originating and terminating BCSMs in the same CCF/SSF is described in the clause of the BCSM description, and is illustrated by the "Basic primitive signal interface model". The communication between BCSMs in different CCF/SSFs is assumed to be the same as existing signalling between exchanges.

Single point of control, as it applies to the distributed functional plane is as follows:

- a) an isolated "half-call" in the CCF/SSF can only be influenced by one SCF at a time;
- b) while one SCF is influencing an isolated "half-call" in the CCF/SSF, it shall be possible to:
 - send DP reports from the CCF/SSF to the same SCF or different SCFs;
 - end the control relationship between the controlling SCF and the CCF/SSF, or change the control relationship to a monitor relationship, then initiate a control relationship between the CCF/SSF and a different SCF.

Multiple Point of Control, as it applies to the distributed functional plane is as follows:

- a) an isolated "half-call" in the CCF/SSF can be influenced by more than one SCF at a time;
- b) while one SCF is influencing an isolated "half call" in the CCF/SSF it shall be possible to:
 - send DP reports from CCF/SSF to the same SCF or different SCFs;
 - initiate a new additional control relationship, or change an existing control relationship into a monitor relationship, or end a control relationship between the controlling SCF and the CCF/SSF.

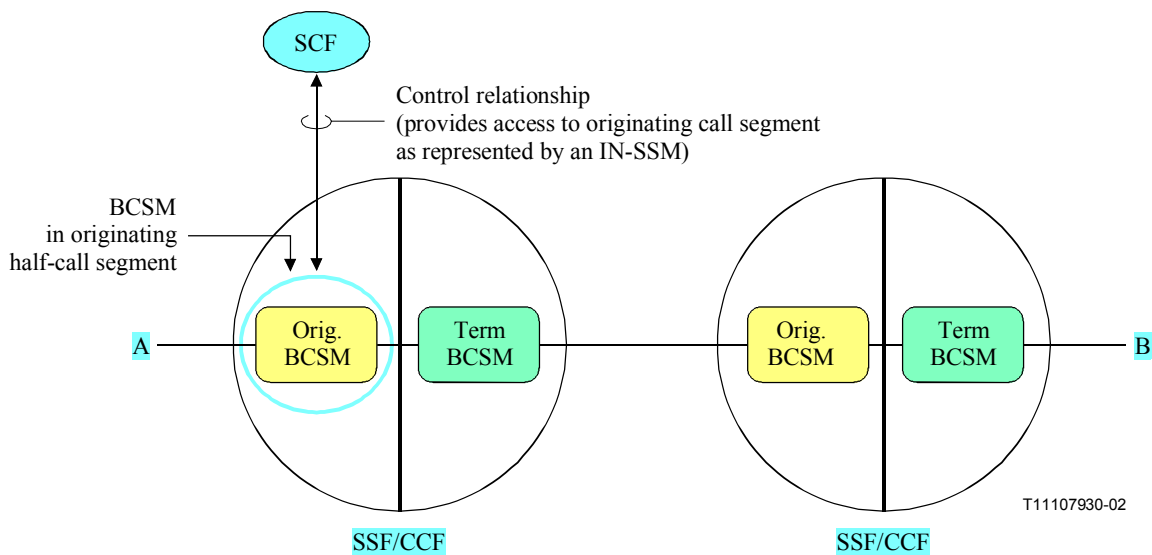


Figure 15/Q.1248.2 – Single-ended control of a two-party call

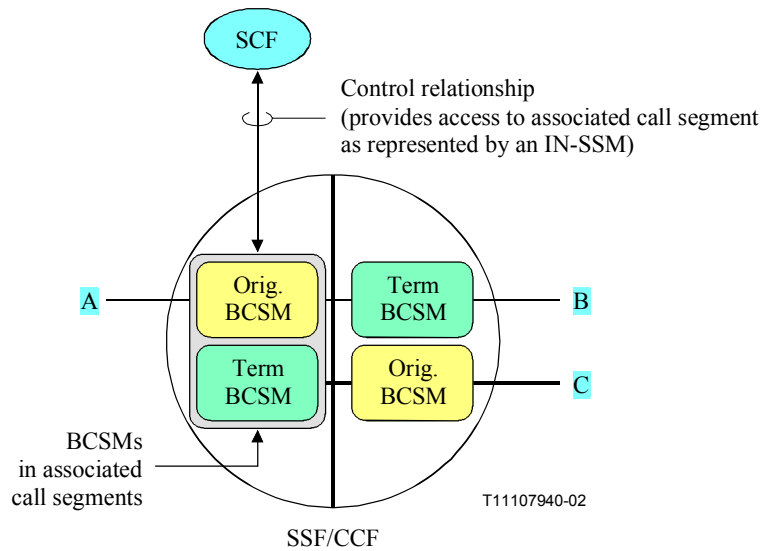


Figure 16/Q.1248.2 – Single-ended control of associated calls

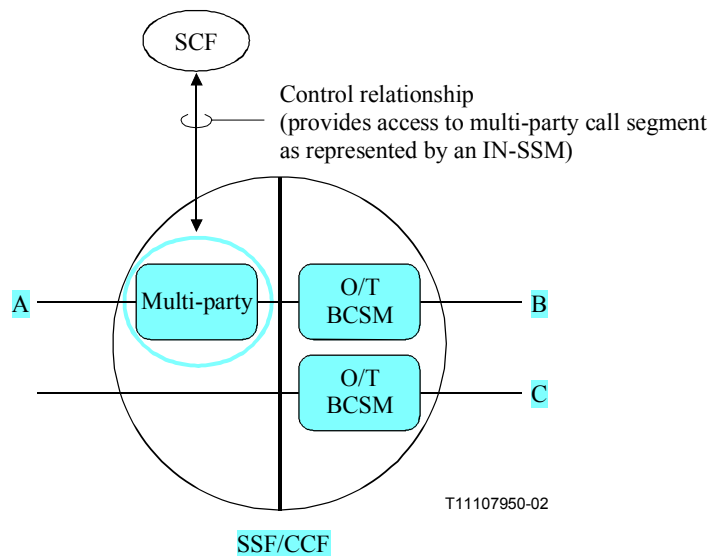


Figure 17/Q.1248.2 – Single-ended control of a multi-party call

6.4.2 The Connection View Model

This clause describes the Connection View model, defines an inventory of CSA and CS Connection View states, and lists the allowable transitions between CSCVs.

The Connection View approach is based on the Connection View (CV) model. CV processing provides an SCF with the ability to influence existing call and connection processing capabilities. It does so by providing a generic representation of call and connection processing resources that support the processing capabilities of interest.

The CV processing within the CCF/SSF can be viewed as translating SCF instructions into operations that are understood by internal CCF/SSF call processing, as well as translating internal call processing events, and the state of internal call processing resources, into information which is understood by the SCF, as indicated in Figure 18.

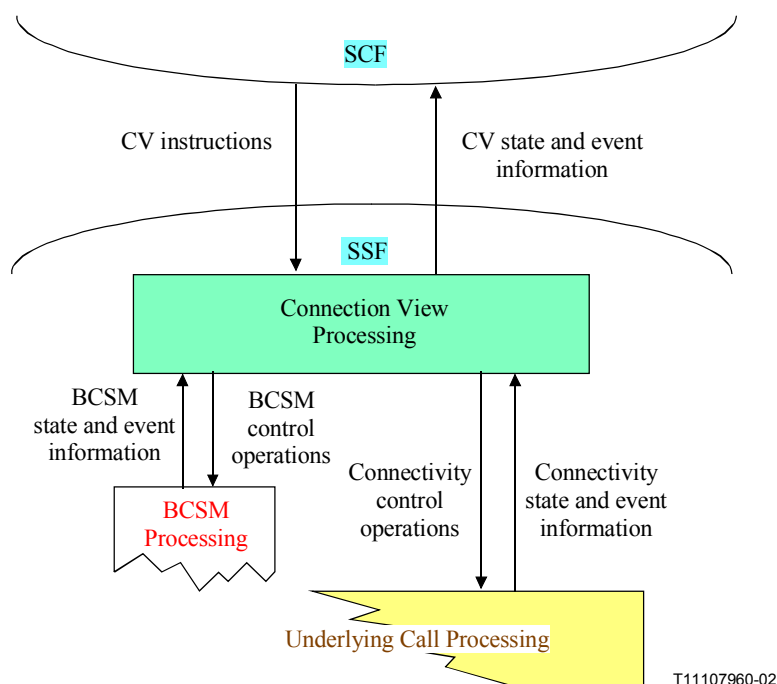


Figure 18/Q.1248.2 – Connection View (CV) processing model

6.4.2.1 Connection View Objects

The CCF/SSF call processing and CV processing resources are described by a set of CV objects which include the following:

- Call Segment Association (CSA);
- Call Segment (CS);
- Legs;
- Connection Point (CP);
- BCSM.

The CV provides a view or abstraction of call and connection processing resources that is independent of supplier implementation and that only represents the essential characteristics of the resources needed by service logic, hiding the physical details and technical complexity of these resources. The CV reflects these properties in terms of *connectivity* context and a *call processing* context. The objects in the *call processing* context manage the relationship of a party to a basic two-party call and the objects in the *connectivity* context manage multiple two-party calls.

The *call processing* context reflects the state of the basic call processing required to set up and maintain the legs in a CS. Only one type of object is explicitly defined in the call processing context and that is BCSM. The BCSM represents the basic call processing required to establish and maintain a communication path from the CS toward an originating party and from the CS toward a terminating party.

The *connectivity* context reflects the state of a CS, or a pair of associated CSs, and includes the set of legs in the CSs and the relationship of each leg to a connection point (CP). The types of objects in the connectivity context correspond to legs, CPs, Call Segments (CSs), and pairs of associated CSs (CSA). Figure 19 provides the graphical examples of the object relationship in the connectivity context.

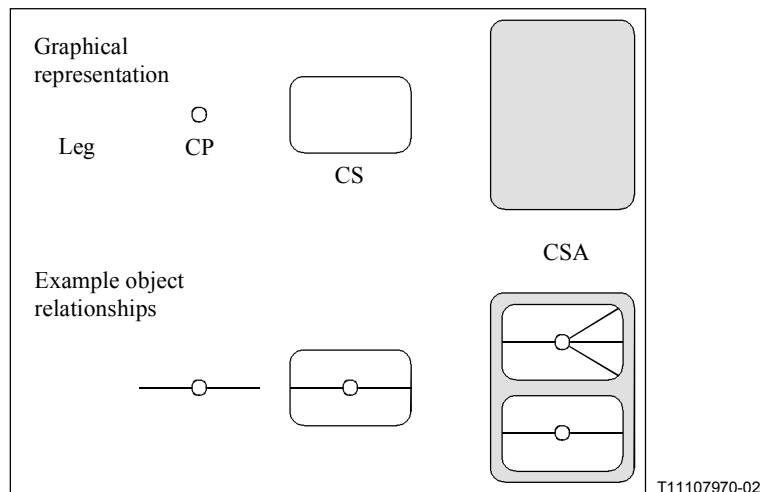


Figure 19/Q.1248.2 – Connection View Objects

As indicated by Figure 19, a Call Segment Association (CSA) object contains one or more Call Segments (CSs). Each CS object contains a single Connection Point (CP) object, which may be connected to leg objects. Depending on its status, each of these objects has implicit or explicit attributes. These objects are further described below:

6.4.2.1.1 Call Segment Association Object

The CSA object contains one or more CSs that are associated in the context of IN control.

A Call Segment Association (CSA) instance is created when an IN service logic instance is invoked that requires IN connection control. It is either created as a result of encountering a TDP in a BCSM that satisfies DP criteria, or is initiated by the SCF independent of encountering TDPs.

A CSA instance is destroyed when the SCF informs the SSF that the IN service logic instance is completed or the CSA should be destroyed. The SSF can also initiate CSA destruction when an IN service relationship to the SCF is to be terminated (e.g. no pending reports or when an error or abnormal condition occurs).

6.4.2.1.2 Call Segment Object

The Call Segment (CS) object contains a Connection Point and any attached legs.

IN service logic may request the manipulation of CSs via operations that act on the Connection View Objects. IN Service Triggering occurs in the "Initial Call Segment".

6.4.2.1.3 Leg Object

The leg object represents a communication path towards a real or virtual end user.

leg: A representation within a call processing state model representing a telecommunication path towards some addressable entity e.g. a path toward a user, intelligent peripheral unit etc. (ITU-T Q.1290).

A **leg** can be designated as a controlling leg or as a passive leg.

- The **controlling leg** is the leg that represents the local access interface at local exchange or the remote access interface at transit exchange (e.g. the incoming line or trunk in an originating call segment, or the outgoing line or trunk in a terminating call Segment). It is the leg for which IN service logic program instances are invoked, either as a result of end user signalling (e.g. a mid-call event) or on behalf of an end user (e.g. individual-based triggering on a line or SCF initiated call). There is no more than one joined or pending

controlling leg in a CSA. Transfer of ownership from an end user supported by a controlling leg to an end user supported by a passive leg is not feasible (see Note).

- The **passive leg(s)** are directed towards the other half-call, i.e. any other leg in a Call Segment Association than the controlling leg.

NOTE – The controlling leg represents line or trunk interfaces. The example of Connection View state transitions given presents limitations on manipulation of these two types of controlling legs. Especially, the "third-party" control aspect, which has the capability of initiating call/connection set-up between two parties from the "third-party" side, is supported partially by defining controlling leg status as "surrogate".

The following three **status values** are specified for the leg:

- **Joined**, indicating that the leg is joined to the connection point and exists.
- **Pending**, indicating that the leg is in a call set-up state (i.e. the call is not yet stable) or a call clearing phase.
- **Surrogate** (for a controlling leg), indicating that the leg supports a communication path towards a virtual party (e.g. the network, or a party that forwarded the call), rather than an external end party. The "surrogate" leg status signifies participation of a "third-party" in the call, but not in the connection. It may represent a relationship with the controlling user, where the remote party has been put on hold and a possible charging ownership.

Legs are uniquely identifiable for the SLPI in an IN Call Segment Association.

For IN it is possible:

- a) To influence the flow of basic call processing associated with a leg (e.g. generate a signalling event and continue basic call processing as appropriate for that event).
- b) To add a passive leg to a Call Segment Association by originating a call or terminating a call; to drop legs (one or more) by clearing calls.
- c) To make or break connections between legs (e.g. join or split).
- d) To move legs from one connection point to another within the same Call Segment Association (e.g. split a leg from one connection point then joining it to another). It is possible to move a leg from one call segment to another call segment within the same CSA or to move a leg from one CSA to another.

6.4.2.1.4 Connection Point Object

A *connection point* represents a joint function between two legs, a conference function between three or more legs, replication function, merging function, or an information distribution function between two or more legs that specifies the directionality of information flow through the connection point (e.g. the connection point could receive information from multiple legs and/or from a SRF resource and distribute it to another leg). It interconnects legs supported by equivalent bearer services, and supports interworking between circuit mode/speech and circuit mode/3.1 kHz audio bearer services.

There can be 1 to n connection points in a Call Segment Association. Each connection point is related to a Call Segment that is represented by the CSA. In a Call Segment Association, it is possible to merge two connection points into a single connection point, thereby merging the corresponding call segments. Finally, it is possible to release a connection point and all of its legs all at once, thereby clearing the corresponding Call Segment.

6.4.2.2 Relationship of BCSM to Connection View CS States

Figure 20 shows, at a high level, the relationship of the Connectivity Objects that make up the Connection View CS States to the Call Processing Objects (BCSMs).

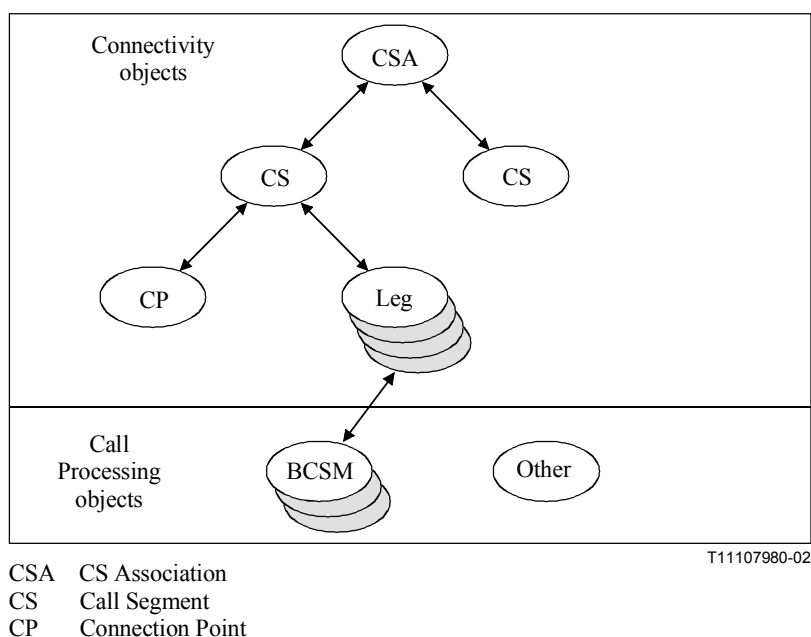


Figure 20/Q.1248.2 – Connection View CS State objects and their relationship with BCSM

A Connection View CS State shows the connectivity between a Controlling Leg and one or more Passive Legs (see below for definitions of terms).

A single BCSM models the Originating or Terminating call processing to set up and maintain a two-party call.

There is a single instance of a BCSM for each Passive Leg of a Connection View CS State (CSCV).

In case no passive leg is present within a CS in the CSCV, the BCSM instance belongs to the controlling leg in that CS.

A Call Segment with only a controlling leg (status "joined") left is assigned a BCSM instance (Call processing Context) to supervise the leg. The "BCSM type" attribute set on the controlling leg is used to reflect the type of BCSM (originating or terminating) to be assigned. The corresponding FSM for the Call Segment is put into the "Waiting_For_Instruction" state when a leg is left in a CS on request from the SCF due to a CPH operation (e.g. DisconnectLeg). The call processing is suspended (e.g. at the O/T_MidCall DP) in the associated BCSM instance.

When the controlling leg becomes connected again to another passive leg (e.g. MergeCallSegments, Connect or AnalyseInformation operation) and the BCSM instances are the same type no BCSM instance will be connected to the controlling leg anymore. The DPs armed and other pending reports for the controlling BCSM instance will be transferred to the passive leg BCSM instance.

6.4.3 Connection View State Transitions

6.4.3.1 Introduction

From a protocol perspective the Connection View objects have been combined to create a set of three standardized CSA Connection Views (CSACVs) and a set of 11 standardized CS Connection Views (CSCVs). The protocol controls the CSAs and the CSs respectively represented by the CSA and CS Connection View states. The CSs representing the same or different CSCV states may be associated in a CSA object described by appropriate CSACV states. Note that the number of passive legs in a CS and the number of CSs into a CSA are not limited to two, allowing all stable multiparty calls with three or more parties, to be modelled.

The SSME-Control functionality is extended in order to control the creation of the CSA instances and to provide an overall coordination function for managing the procedures which have an impact on more than one CSA.

Next, the SSME-Control procedures and transitions are described followed by a description of the CSA procedures and transitions. Hereafter, the CS procedures and transitions are detailed.

6.4.3.2 SSME-Control

6.4.3.2.1 Introduction

The SSME-Control functionality is extended in order to control the creation of the CSA instances and to provide an overall coordination function for managing the procedures which have an impact on more than one CSA (e.g. MoveCallSegments).

A CSA instance is created:

- on receipt of a CreateCallSegmentAssociation operation;
- on receipt of an InitiateCallAttempt operation as the first operation of an SCF-initiated dialogue;
- on determining that a service triggering will occur.

The SSME-Control does not appear as a separate process in the SDL descriptions since it is assumed to be embedded into the Interface Handler (IH) process type behaviour description.

6.4.3.2.2 Transition table for SSME-Control

The SSME-Control transitions related to the CPH Handling are described in Table 13.

The signals used in the transition table for SSME-Control are as follows:

- The ExportCSReq signal shall request the addressed CSA process instance to export the indicated call segment.
- The ExportCSResp signal shall be returned by the addressed CSA process instance to indicate to the SSME-Control that the export of the call segment was successfully executed.
- The ImportCSReq signal shall request the addressed CSA process instance to import the indicated call segment, since unique numbering of the legs are required on a CSA basis renumbering of the legs for all remaining CSs are required.
- The ImportCSResp signal shall be returned by the addressed CSA process instance to indicate to the SSME-Control that the import of the call segment was successfully executed.

Table 13/Q.1248.2 – SSME-Control Transition Table

Event	Action
Triggering received from FIM	Create an instance of a CSA process (csa1 signal), allocate a CSAID, update the routing table and associate the signalling controller CallRef with the newly created CSA.
InitiateCallAttempt	Create an instance of a CSA process (csa1 signal), allocate a CSAID, update the routing table. NOTE – For CS-1, the InitiateCallAttempt always creates a new CSA process instance. For CS-2 onwards, the ICA may be preceded by a CreateCallSegmentAssociation, in which case the InitiateCallAttempt shall not create a new CSA.

Table 13/Q.1248.2 – SSME-Control Transition Table

Event	Action
CreateCallSegmentAssociation	Create an instance of a CSA process (csa1 signal), allocate a CSAID, update the routing table, sent a CreateCallSegmentAssociationResult to SCF.
MoveCallSegments	<p>Send an ExportCSReq event in order to request the CSA to export the source CS and perform the leg numbering by the CSA prior to exporting the source CS and wait for ExportCSResp.</p> <p>After receiving the ExportCSResp send an ImportCSReq in order to import the previously exported CS into the target CSA with the identifier of the newCallSegment and wait for ImportCSResp.</p> <p>After receiving the ImportCsResp send a MoveCallSegmentsResult to the SCF.</p>

6.4.3.3 Call Segment Association Connection View (CSACV) Transitions

6.4.3.3.1 Introduction

This clause describes the Call Segment Association Connection View (CSACV) processing for each state from a protocol perspective, and gives examples of the events, or the SCF operations that result in creating a new CSACV state, destroying a CSACV state, and transitioning from one CSACV state into another CSACV state. The transitions between CSACV states occur due to end user actions (e.g. off-hook or disconnect), switch processing (e.g. switch-based features), or as a result of processing SCF operations.

6.4.3.3.2 Functional Procedures for Call Segment Association (CSA)

In the case when a RequestReportBCSMEvent (RRBE) operation is received by the CSACV with legID for which no corresponding passive leg yet exists, the "to be armed" BCSM events shall be queued at the CSA level.

Only RRBE arming requests for one new LegID ("leg to be created") should be accepted and queued.

The EDP arming requests may be provided in one or more RRBE operations from the SCF. It is the task of the SSF to accumulate all RRBE arming requests to be queued for the same LegID.

When an operation (e.g. Connect) subsequently creates a new passive leg with a legID equal to a legID for which BCSM events have been queued, the queued BCSM events shall be retrieved and transferred to the exiting CS instance.

Receiving a RRBE for arming BCSM events on an unknown legID, while RRBE arming requests are already queued for a different legID, should be processed as an error "Unknown legId".

A RRBE for disarming at least one DP on an unknown legID is processed as an error "Unknown legId".

The queued RRBE arming requests are maintained at the CSA level. They disappear when the CSA is deleted, or when the new leg is created by an operation referencing the corresponding legID.

6.4.3.3.3 Transition diagram for Call Segment Association (CSA)

The transition diagram for the CSA is given in Figure 21.

NOTE – Figure 21 shows examples and does *not* show all possible transitions.

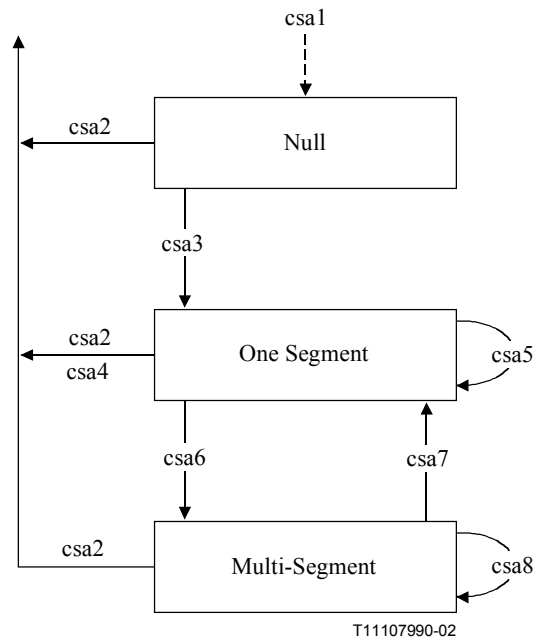


Figure 21/Q.1248.2 – CSACV transition diagram

The states identified for the CSACV transition diagram are:

Null: The state Null represents the condition when a call segment association is created without any call segments.

One Segment: This state represents a call segment association containing one call segment.

Multi-Segment: This state represents a call segment association containing multiple calls.

The CSACV transition diagram contains the following transitions:

csa1: Create Call Segment Association;

csa2: Delete Call segment Association (ReleaseCall);

csa3: Triggering event from FIM (examples are: SetupInd, SetupReqInd), InitiateCallAttempt;

csa4: ReleaseCall;

csa5: DisconnectLeg;

csa6: SplitLeg, InitiateCallAttempt;

csa7: ReleaseCall, MergeCallSegments;

csa8: SplitLeg, InitiateCallAttempt, MergeCallSegments, DisconnectLeg, MoveLeg.

6.4.3.3.4 Transition table for Call Segment Association (CSA)

The Call Segment Association (CSA) process type transitions related to the CPH Handling are described in Table 14/Q.1238.2.

The signals used in transition table for Call Segment Association (CSA) are as follows:

The ExportCSReq signal shall request the addressed CSA process instance to export the indicated call segment.

The ExportCSResp signal shall be returned by the addressed CSA process instance to indicate to the SSME-Control that the export of the call segment was successfully executed.

The ImportCSReq signal shall request the addressed CSA process instance to import the indicated call segment, since unique numbering of the legs is required on a CSA basis renumbering of the legs for all remaining CSs is required.

The ImportCSResp signal shall be returned by the addressed CSA process instance to indicate to the SSME-Control that the import of the call segment was successfully executed.

The SetLegLocation signal shall set the location of a leg, i.e. to which call segment a given leg identifier given in the signal belongs to.

The GetLegLocation signal shall request the location of a leg, i.e. the call segment identifier to which the leg identifier given in the signal belongs to.

The ImportLeg signal shall invoke to move a communication path in a CS (in the process of moving the path from another CS to this CS). This operation shall add the existing leg specified by the leg identifier to the CS. For a passive leg, the number of legs within a CS are incremented by one and the leg vector is appended. For the controlling leg, the leg status shall change from "shared" to "joined".

The ExportLeg signal shall remove the leg specified by the leg identifier to the specified CS, but shall not destroy the leg. For a passive leg, the number of legs shall be decremented by one, and the state vector is removed. For the controlling leg, the leg status shall change to "shared".

The RemoveLeg signal shall remove the leg from the CS that the identifier specifies. The number of legs in a CS is decremented by one, and the associated state vector is removed. The path corresponding to the removed leg is released as a result of this operation.

Table 14/Q.1248.2 – Call Segment Association (CSACV) Transition Table

CSACV State ⇒ Event ↓	Null	One Segment	Multi-Segment
SplitLeg operation	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Check SplitLeg Argument – ExportLeg from currently residing segment – Create Call Segment process instance – ImportLeg into new call segment – Pass: Splitleg(c) to the new call segment or SplitLeg(p) to the segment where the leg is exported from – Go to Multi-Segment 	<ul style="list-style-type: none"> – Check SplitLeg Argument – ExportLeg from currently residing segment – Create Call Segment process instance – ImportLeg into new call segment – Pass: Splitleg(c) to the new call segment or SplitLeg(p) to the segment where the leg is exported from – Remain in Multi-Segment

Table 14/Q.1248.2 – Call Segment Association (CSACV) Transition Table

CSACV State ⇒ Event ↓	Null	One Segment	Multi-Segment
Initiate-Call-Attempt operation	<ul style="list-style-type: none"> – Check InitiateCallAttempt Argument – Create Initial Call Segment process instance – SetLegLocation for the passive and controlling legs to the newly created CS – Pass InitiateCallAttempt to the newly created CS – Go to One Segment 	<ul style="list-style-type: none"> – Check InitiateCallAttempt Argument – Create Call Segment process instance – SetLegLocation for controlling and passive legs to the newly created CS – Pass InitiateCallAttempt to the newly created CS – Go to Multi-Segment 	<ul style="list-style-type: none"> – Check InitiateCallAttempt Argument – Create Call Segment process instance – SetLegLocation for controlling and passive legs to the newly created CS – Pass InitiateCallAttempt to the newly created CS – Remain in Multi-Segment
Merge-Call-Segments operation	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Check MergeCallSegments argument – Move all legs from the source CS to the target CS – Delete the source CS – Remain in Multi-Segment, if number of CSs is > 1, otherwise go to One Segment
Disconnect-Leg operation	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Check DisconnectLeg Argument – GetLegLocation to locate the CS where the leg shall be removed – RemoveLeg in CS – Pass DisconnectLeg to the CS where disconnection shall occur – Remain in same state if number of legs left in call segment is greater than zero else move to Null 	<ul style="list-style-type: none"> – Check DisconnectLeg Argument – GetLegLocation to locate the CS where the leg shall be removed – RemoveLeg in CS – Pass DisconnectLeg to the CS where disconnection shall occur – Remain in same state if number of legs left in call segment is greater than zero or number of call segments left is greater than 1, else move to One Segment

Table 14/Q.1248.2 – Call Segment Association (CSACV) Transition Table

CSACV State ⇒ Event ↓	Null	One Segment	Multi-Segment
MoveLeg operation	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Check MoveLeg Argument – ExportLeg from the CS where it is currently residing – ImportLeg into the target CS – Remain in same state or move to the One Segment state if it is the ExportLeg of the last passive leg of a CS
Import-CSReq	<ul style="list-style-type: none"> – Check ImportCSReq – Import CS and assign the CS identifier – Send ImportCSResp to IH – Go to One Segment 	<ul style="list-style-type: none"> – Check ImportCSReq – Import CS and assign the CS identifier – Send ImportCSResp to IH – Go to Multi-Segment 	<ul style="list-style-type: none"> – Check ImportCSReq – Import CS and assign the CS identifier – Send ImportCSResp to IH – Remain in same state
Export-CSReq	<ul style="list-style-type: none"> – Discard operation and return error (Task Refused) – Remain in same state 	<ul style="list-style-type: none"> – Check ExportCSReq – Export CS the given source CS – Send ExportCSResp to IH – Go to Null 	<ul style="list-style-type: none"> – Check ExportCSReq – Export CS the given source CS. When the CS with the controlling leg joined is exported, the status of the controlling leg in all the other CSs are set to surrogate. (This is done by the ExportControllingLeg procedure.) – Send ExportCSResp to IH – Remain in same state if number of call segments left is greater than 1, else move to One Segment

6.4.3.4 Call Segment Connection View (CSCV) Transitions

6.4.3.4.1 Introduction

This clause describes the Call Segment Connection View (CSCV) processing for each state from a protocol perspective, and gives examples of the events or the SCF operations that result in creating a new CSCV state, destroying a CSCV state, and transitioning from one CSCV state into another CSCV state. The transitions between CSCV states occur due to end user actions (e.g. off-hook or disconnect), switch processing (e.g. switch-based features), or as a result of processing SCF operations. Note that the transition events related to user interactions (e.g. ConnectToResource) are outside the scope of this capability set.

Table 15 showing "Transition of DP events to the CSCV states", followed by CSCV state transition figures, provides a concise summary and overview of these transition events.

The transition figures are not intended to cover all possible transitions, but merely to provide an overview.

IN-initiated transitions may be categorized according to the manner in which they affect the CSCV state:

- 1) *BCSM-only transitions*: Occur when the BCSM changes state (or continues normal processing) with no explicit change to the CSCV state. For example, if SSF sends an Initial DP from the "Originating_Setup" CSCV state, and SCF responds with Connect, the BCSM changes state, but no explicit change to the initial CSCV state occurs. Note that, in processing the BCSM request, there *may* be an implicit change to the CSCV state through non-IN processing (e.g. a switch-based screening feature).
- 2) *SCF-requested connectivity changes*: Occur when SCF explicitly requests a change in connectivity. This request may or may not cause a CSCV state transition.

Within this clause, each CSCV state is described with respect to the following:

- a) ***Relationship with the BCSM***: a list of the Points in Call (PICs) and DPs that are associated with this CSCV state. Note that the PICs are associated with the passive legs (i.e. *Leg p1* or *Leg p2*) only, to avoid the ambiguity that would otherwise result for the "Stable_Multi_Party", "Stable_Passive_Party", and "Forward" CSCV states. Additional detail on the mapping of DPs and events to the CSCV states is provided in this clause.
- b) ***Entry Events***: the events that cause the call processing and CV processing associated with this CSCV state to be initiated.
- c) ***Exit Events***: the events that signify normal completion of the call processing and CV processing associated with this CSCV state. This item begins with a summary of the SSF events (i.e. TDPs, EDPs, and BCSM events) relevant to the CSCV state approach to Call Party Handling. A list of SCF responses that cause CSCV state transitions follow.

Figure 22 depicts the CSCV states to be supported by the CV processing. The On_Hold state defined in Q.1238.2 appears on the figure but is no longer used in the context of IN-CS-4. The configuration initially modelled using the On-Hold state can be described using the Stable 1-Party and Stable Multi-Passive Party states.

In subsequent subclauses detailed descriptions of these Connection View States (CSCV states), including their relationship with the BCSM and the transition events between CSCV states are provided.

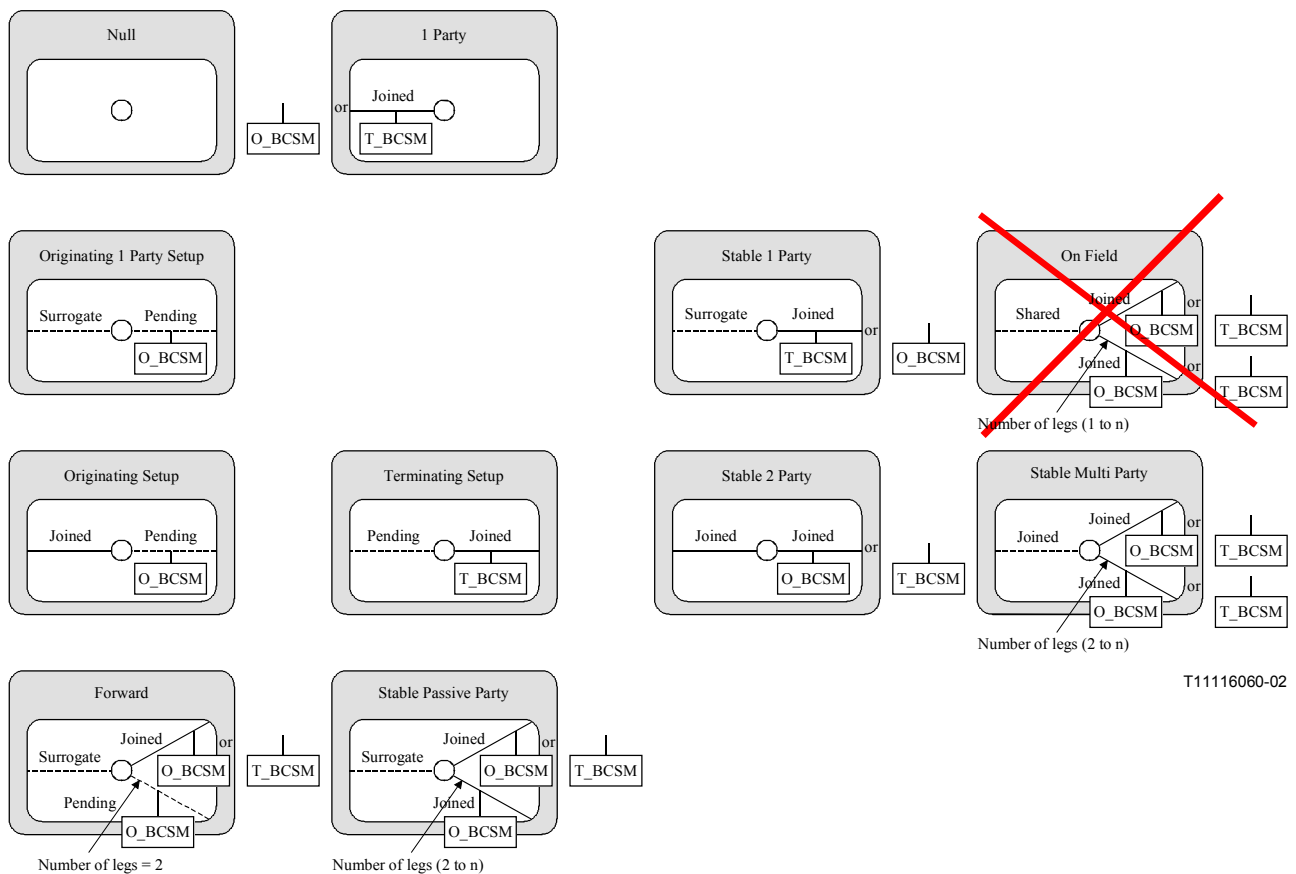


Figure 22/Q.1248.2 – Call Segment Connection View (CSCV) States

CSCV State Definitions

Null: This state represents the condition where no call processing is active and there is no controlling leg or passive leg connected to the connection point.

Originating Setup: This state represents a call segment instance with a joined controlling leg and a pending passive leg with an associated O_BCSM (e.g. an originating two-party call in the setup phase).

Terminating Setup: This state represents a call segment instance with a pending controlling leg and a joined passive leg with an associated T_BCSM (e.g. a terminating two-party call in the setup phase).

Stable 2 Party: This state represents a stable two-party call, and is either an originating or a terminating call from the perspective of the controlling user with a joined controlling leg and a "joined" passive leg with either an O_BCSM or a T_BCSM associated with it.

1-Party: This state represents a 1-party call with a joined controlling leg status and no passive legs. The O_BCSM or T_BCSM is associated with the controlling leg since no passive leg is connected to the CP.

Originating 1 Party Setup: This state represents a 1-party call with a surrogate controlling leg and a pending passive leg with which an O_BCSM is associated.

Stable 1 Party: This state represents a 1-party call with a surrogate controlling leg and a joined passive leg with either an O_BCSM or T_BCSM associated with it, that is in a stable phase.

Forward: This state represents a forwarded call with a surrogate controlling leg, a joined passive leg with which an O_BCSM or a T_BCSM is associated, and a pending passive leg with which an O_BCSM is associated.

Stable_Passive_Party: This state represents a transferred call with a surrogate controlling leg and two to N joined passive legs with each of which an O_BCSM or a T_BCSM is associated. The call between the involved passive legs is in the stable phase.

Stable_Multi_Party: This state represents a multi-party call with a joined controlling leg and two to N joined passive legs with each of which an O_BCSM or a T_BCSM is associated.

6.4.3.4.2 Functional Procedures for Call Segment (CS)

6.4.3.4.2.1 BCSM type indication

A "BCSM type" attribute shall be allocated with the controlling leg at the moment this leg is created, indicating if the controlling leg belongs to an Originating BCSM or a T-BCSM, i.e. an indication for the type of BCSM in which call triggering occurred.

When a controlling leg is left in a CS, the "BCSM type" attribute shall determine which BCSM shall apply. The conditions of the leg i.e. the armed EDPs, the ApplyChargingReport pending, the EventNotificationCharging pending, and the CallInformationReport pending are also applied to the same leg after creation of an BCSM instance (e.g. after a SplitLeg). The latter entails that if no EDPs and/or reports were armed for the leg in relation to the created BCSM type, then an explicit arming is to be made using e.g. the RequestReportBCSMEvent operation, if needed by the SLP.

In order to obtain full alignment with the SLP design in the SCF, no mapping of events from O_BCSM to T_BCSM events, and vice versa, shall occur. This means that the DP, at which call processing is to be suspended, shall be selected in the associated BCSM instance based on the actual state of the call as seen from the legs viewpoint; e.g. if in an "Active PIC" then the BCSM instance shall be put in MidCall DP of the active BCSM state when the leg is left in a CS after e.g. a DisconnectLeg or SplitLeg operation.

The following is noted for the O_BCSM and T_BCSM instances:

- When the O_BCSM instance is put in the O_Mid_Call DP then:
 - a transition to the Collect_Information, Analyse_Information or Select_Route is possible by means of sending the appropriate operations from the SCF. This allows to support follow-on calls in accordance with the extended BCSM transitions as described herein.
- When the T_BCSM instance is put in the T_Mid_Call DP then:
 - in order to allow to the setting up of an originating call connection to another terminating called party, the ContinueWithArgument followed by the InitiateCallAttempt and the MergeCallSegments operations may be sent from the SCF in order to perform a "transfer call" (see figure below). It is noted that the transfer of a call in this case where the controlling leg has the T-BCSM type attribute, is only possible in a serving node exchange (not in a transit exchange).

If in the above-mentioned cases, the controlling joined leg is the only leg left in a CS due to a SCF operation (DisconnectLeg, SplitLeg, MoveLeg etc.), the CS shall be put in the 1-Party state with the associated SSF_FSM state in "Waiting For Instructions". In order to resume from the O_Mid_Call and T_Mid_Call DPs of the associated BCSM states, a ContinueWithArgument operation can be sent by the SLPI.

It should be noted that the problem of stranding a "surrogate" controlling leg does not exist as it does for a "joined" controlling leg, because it is assumed that a stranded surrogate leg will be destroyed due to the fact that no voice path and no signalling is associated with this controlling leg.

6.4.3.4.2.2 Rules for inter-BCSM precedence Event handling rules

For Call Party Handling, it is possible to detect the same event (e.g. mid-call or user abandon) on multiple BCSMs. However, it is necessary to provide only one report of the given event to the SLPI in the SCF.

6.4.3.4.2.2.1 Event Detection Rules

The inter-BCSM precedence rules for the **detection of events** in CCF/SSF signalled on the **controlling leg** follow.

Two cases apply:

- 1) The event signalled on the controlling leg does *not* indicate the particular passive leg to which it applies (e.g. analogue hook-flash), then the following event detection rules apply:
 - a) If the CSA contains multiple call segments (CSs), the event signalled on the controlling leg should be detected only on the CS with a controlling *legStatus* of "joined" or "pending".
 - b) If the CS to which the event applies is a multi-party CS, then the event should be detected on all BCSMs within this multi-party CS.
- 2) The event signalled on the controlling leg indicates the particular passive leg to which it applies (e.g. a party identifier indication), then the mid-call event should be detected on the BCSM associated with the indicated passive leg. Any support for this feature is outside the scope of this capability set.

However, in all cases, the event is reported only once to the SCF according to the Event Reporting Rules defined below.

6.4.3.4.2.2.2 Event Reporting Rules

The following inter-BCSM precedence rules for the **reporting to the SCF of events** to be signalled for the controlling leg shall apply:

- 1) An event shall only be reported once to the SCF irrelevant of the number of passive legs per CS connected to the controlling leg (via CP).
- 2) In case more than one event is possible due to different BCSM states prevailing for the different passive legs, then the reported event on the controlling leg shall be the event reflecting the most advanced call state of the BCSMs. For example, in case of a CSCV Multi-Party call with two passive legs, one passive leg (O_BCSM) in alerting (before answer) or release state (after answer) and the other passive leg (O_BCSM) in Active Call PIC. When the calling party goes on-hook and a release message is received from the calling party, two events are possible to report on the controlling leg: either O_Abandon or O_Disconnect. In this case, O_Disconnect is reported according to the most advanced call state of the BCSMs, which is PIC Active call state.
- 3) A "BCSM type" attribute on the controlling leg indicates if the controlling leg belongs to an Originating or a Terminating-BCSM. The "BCSM type" attribute is set to reflect the type of BCSM (originating or terminating) in which the trigger, or event, occurred at the moment the CS (i.e. initial CS in the CSA) was created:
 - If the controlling leg belongs to an O_BCSM it is reported as an originating event.
 - If the controlling leg belongs to a T_BCSM it is reported as a terminating event.

6.4.3.4.2.3 Creation of O/T_BCSM based on "BCSM type" attribute

A CS with only a controlling leg (leg status "joined") left is assigned a BCSM instance to supervise the leg. The "BCSM type" attribute set on the controlling leg is used to reflect the type of BCSM (originating or terminating) to be assigned. The corresponding SSF_FSM for the CS shall be put in

"Waiting_For_Instructions" state where a leg is left in a CS due to a CPH (e.g. DisconnectLeg) operation and call processing is suspended at the Mid_Call DP in the associated BCSM.

When the controlling leg becomes connected to the passive leg, no BCSM instance shall be connected to the controlling leg provided the same type of BCSM applies for the controlling and passive leg. The BCSM instance which was connected to the controlling leg disappears in case a passive leg is moved (imported) to the Call Segment (e.g. MergeCallSegments operation). If a new passive leg is created (e.g. Connect operation) within the CS, then the existing BCSM instance becomes connected to the passive leg.

6.4.3.4.2.4 Release of a Call/Connection

Release Event Processing General Rules:

- a) When a DP related to a release event (e.g. *O/T_Disconnect*, *O_CalledPartyBusy*, *T_Busy*, *O/T_NoAnswer*, *Route_Select_Failure*) on a leg is reported to the SCF the LegId value and its pointer to a BCSM instance are retained until call processing at the current DP is resumed.
- b) If the release occurs on a joined leg in the CS, the leg state is changed as follows: c-leg: "joined" to "surrogate"; p-leg: "joined" to "pending".
- c) If the release occurs on the last joined leg in the CS, the leg state is changed without changing the CSCV state and the CS is deleted upon resumption of the call processing.
NOTE – This implies *transient existence* of e.g. CSCV state "Originating_Setup" and "1_Party" with a surrogate controlling leg and "Terminating_Setup" with two pending legs.
- d) Resumption (using Continue, or ContinueWithArgument) following a release event reported for a *Controlling Leg*, causes propagation of the release event, the release of passive legs in the Call Segment, and deletion of the Call Segment. Exception: Resumption when two, or more than two, joined passive legs are connected in the CS before resumption does not result in the propagation of the release event, all passive legs are retained and the CS continues.
- e) Resumption (using Continue, or ContinueWithArgument) following a release event reported for a *Passive Leg*, causes removal of the passive leg from the Call Segment and the propagation of the release event. Exception: When only two joined passive legs with the same type of BCSM (i.e. either O_BCSM or T_BCSM) or more than two joined legs are connected in the CS, the resumption causes removal of the passive leg but does not result in the propagation of the release event.

6.4.3.4.2.5 DisconnectLeg Operation

A DisconnectLeg operation for a controlling and passive leg shall physically release the specified leg from the connection point towards the remote user.

The following general DisconnectLeg and BCSM Transition Rules apply:

- a) When a DisconnectLeg(p) is performed on the last joined passive leg in a CS, for which an association with the joined controlling leg exists, the BCSM is suspended according to the rule b).
- b) If call processing is already suspended at a DP reported for the c-leg, it remains suspended at that DP; otherwise, a transition to O_/T_Mid_Call DP for leg c is performed (due to the applied CPH operation).
- c) When a disconnectLeg (c) is performed on the joined controlling leg in a CS, then the BCSM instance(s) attached to the passive leg(s) will be suspended at MidCall DP, if not already suspended at a DP on the passive leg.

- d) On receipt of a valid DisconnectLeg, the leg is deleted immediately. When the last joined leg in the CS is released by SCF due to DisconnectLeg, the CS (including its associated BCSM instances) is deleted immediately (i.e. no call process suspension in this case).

An example, depicting the release sequences as a result of a DisconnectLeg operation for the controlling (c) and passive legs of a Call/Connection for a normal call scenario where both call segments are in the "Stable_2_Party" state (with the associated O_BCSM and T_BCSM in the active state), is given in Figure 22.

The information flows for the receipt of the following operations are illustrated:

- The following actions occur when a DisconnectLeg (c) operation (for the controlling leg) is received in CS1 CSCV state "Stable_2_Party" from the SLP in the SCF:
 - a ReleaseReq signal is sent toward the A-side with a cause value "disconnect controlling leg"; and
 - the O_BCSM goes to the appropriate O_MidCall DP according to rule c) above.
- The following actions occur when a subsequent DisconnectLeg (p1) operation (for the passive leg) is received in CS1 CSCV state "Stable_1_Party" from the SLP in the SCF:
 - a ReleaseReqInd signal is sent to remote T_BCSM with a cause value "disconnect passive leg"; and
 - the O_BCSM state transition occurs according to rule d) above as the last joined leg in the CS is deleted.

6.4.3.4.2.6 User interactions during the SSF-FSM "Monitoring" state

During the "Monitoring" state of the SSF-FSM, it shall be possible to perform user interactions in order to send tones, announcements and display information. For details refer to clause 8.

6.4.3.4.2.7 Call Segment and associated BCSM states for CPH operations

The following principles apply for the Call Segment and associated BCSM states for CPH operations:

- 1) If a CPH operation (SplitLeg, DisconnectLeg, MergeCallSegments, MoveCallSegment or MoveLeg) is received in the Monitoring state, the FSMs for the involved Call Segments shall first go to the "Waiting for Instruction" state while the associated BCSM instances within the involved Call Segments shall move from the O/T PIC to the corresponding O/T_Mid_Call DP in order to handle subsequent EDP rearming. It shall be noted that when the BCSM instance is in a DP, it shall stay in this DP where appropriate after processing of the CPH operation. This means that only some PIC transitions to MidCall DPs are possible as described in the templates of the operations.
- 2) The receipt of a CPH operation received and/or processed in the state Waiting for Instructions, shall not cause the change of the state Waiting for Instructions.
- 3) For these CPH operations, which create a new BCSM for the controlling leg, only this BCSM shall be put into the DP-O/T_Mid_Call, the BCSM states of the other involved CS should transit from the PIC into the corresponding DP state, or, if already suspended, remain in the DP wait state. The associated CS SSF_FSM for the newly created BCSM and the other involved CS shall be put into the "Waiting for Instructions" state so that the EDPs can be rearmed.
- 4) All CPH operation sequences shall be finalized by an operation which changes the state into Monitoring (e.g. ContinueWithArgument, Connect).

6.4.3.4.2.8 "Forward" and "Stable_Passive_Party" connection view behaviour principles

The connection view for Forward and Stable_Passive_Party state behaves differently for the case where an operation is received from Terminating Setup (e.g. Call forward service) or Stable_1_Party (e.g. meet me conference service). Depending on the situation, the signalling events received from one party (one leg) may have to be relayed to the other party.

The Stable_Passive_Party CSCV state can be entered via:

- a) Originating_1_Party_Setup -> Stable_1_Party -> Forward-> Stable_Passive_Party, e.g. SCF initiated call:

A SCF initiated call (due to an InitiateCallAttempt and Connect operation). In this CSCV state the passive legs shall both have O_BCSMs connected to them. The receipt of a SetupRespConf, CallProgressReqInd or DataReqInd from the connected-to leg is **not relayed** to the initial created leg ("outgoing" leg). The ReleaseReqInd will not be transmitted to the initial party.

- b) Terminating_Setup -> Forward -> Stable_Passive_Party, e.g. User initiated forwarded call:

A call diversion at the terminating party (instantiated by a T_BCSM). In this CSCV state the passive leg connected to the initial party is associated with a T_BCSM, while the passive leg connected to the diverted-to party is associated with an O_BCSM. The receipt of a SetupRespConf, CallProgressReqInd or DataReqInd from the diverted-to leg is **relayed** to the initial created leg ("incoming" leg). The ReleaseReqInd will be transmitted to the initial party.

NOTE – From the point of signalling interaction, only the view of one SLP is considered, which is assumed to have the full visibility of the call, and any signalling interworking aspects, in case of multiple SLPIs with different visibility of the call, is not handled.

6.4.3.4.2.9 "Stable_2_Party" and "Stable_1_Party" connection view behaviour principles

Since the Stable_1_Party CSCV state can be entered from the "Stable_2_Party" state as a result of the DisconnectLeg c operation, the receipt of a SetupConf, or a DataInd (from B-party) in the T_BCSM may not cause the sending of a corresponding signal (a SetupRespConf, or a DataReqInd) to the remote half call O_BCSM. A SetupConf, or DataInd can be received, for example due to glare.

However, in the Stable_1_Party CSCV state when an O_BCSM is connected to the passive leg, the receipt of a SetupRespConf, or a ServiceFeatureIndicationReqInd (from B-party) in the O_BCSM causes the sending of a corresponding DP, if armed.

6.4.3.5 Overview of Transitions of DP Events to the CSCV states

Table 15 and CSCV state diagrams provide an overview of the transitions of DP events to the CSCV states.

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ _Party	Stable_ Multi_ _Party
Origination_ Attempt DP for leg c	Originating_Setup (Note 5)	NA (Note 6)	NA	NA	NA	NA	NA	NA	NA
Origination_ Attempt_ Denied DP for leg c and leg p (due to InitiateCall Attempt or Connect for call diversion)	Originating_Setup	NA (Note 6)	NA	NA	NA	NA	Forward for leg p2	NA	NA
Origination_ Attempt_ Authorized DP for leg c and leg p (due to InitiateCall Attempt or Connect for call diversion)	Originating_Setup	Originating_1_ Party_Setup	NA	NA	NA	NA	Forward for leg p2	NA	NA
Collected_ Information DP for leg c and leg p (due to InitiateCall Attempt or Connect for call diversion)	Originating_Setup	Originating_1_ Party_Setup	NA	NA	NA	NA	Forward for leg p2	NA	NA

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ _Party	Stable_ Multi_ _Party
Analysed_Information DP for leg c and leg p (due to InitiateCall Attempt or Connect and SelectFacility (with destination addr) for call diversion)	Originating_Setup for leg c	Originating_1_Party_Setup for leg p	NA	NA	NA	NA	Forward for leg p2	NA	NA
O_Term_Seized DP for leg p	Stable_2_Party	Stable_1_Party	NA	NA	NA	NA	Stable_Passive_Party for leg p2	Stable_Passive_Party for leg p2	NA
Route_Select_Failure DP for leg p	Originating_Setup	Originating_1_Party_Setup	NA	NA	NA	Originating_Setup	Forward for leg p2	NA	NA
Authorized_Route_Failure DP for leg p	Originating_Setup	Originating_1_Party_Setup	NA	NA	NA	NA	Forward for leg p2	NA	NA
O_Called_Party_Busy DP for leg p	Originating_Setup	Originating_1_Party_Setup	Originating_1_Party_Setup	NA	NA	Originating_Setup	Forward for leg p2	If passive legs = 2 before release Forward else if > 2 then Stable_Passive_Party	If remaining legs = 2 Stable_2_Party else Stable_Multi_Party

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ Party	Stable_ Multi_Party
O_No_Answer DP for leg p	Originating_Setup	Originating_1 _Party_Setup	Originating_1 _Party_Setup	NA	NA	Originating _Setup	Forward for leg p2	If passive legs = 2 before release Forward else if more than 2 legs then Stable_ Passive_ Party	If remaining legs = 1 Stable_2_ Party else Stable_ Multi_Party
O_Answer DP for leg p	Stable_2_Party	Stable_1_Party	Stable_1_ Party	NA	NA	Stable_2_ Party	NA	Stable_ Passive_ Party	Stable Multi_Party
O_Suspend DP for leg p	NA	NA	Stable_1_ Party	NA	NA	Stable_2_ Party	NA	Stable_ Passive_ Party	Stable Multi_Party
O_Re-answer DP for leg p	NA	NA	Stable_1_ Party	NA	NA	Stable_2_ Party	NA	Stable_ Passive_ Party	Stable Multi_Party
O_Mid_Call DP (Send_Call PIC) for leg c	Originating_Setup	NA	NA	NA	NA	NA	NA	NA	NA
O_Mid_Call DP (O_Alerting PIC) for leg c	NA	NA	Stable_1_ Party	NA	1_Party	Stable_2_ Party	NA	NA	Stable Multi_Party
O_Mid_Call DP (O_Active PIC) for leg c and leg p	NA	NA	Stable_1_ Party	NA	NA	Stable_2_ Party	NA	Stable_ Passive_ Party	Stable Multi_Party

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ Party	Stable_ Multi_ Party
O_Mid_Call DP (O_Suspended PIC) for leg c and leg p	NA	NA	Stable_1_ Party	NA	NA	Stable_2_ Party	NA	Stable_ Passive_ Party	Stable Multi_Party
O_Disconnect DP for leg c and leg p	NA	NA	For leg p: Originating_1 _Party_ Setup For leg c: NA	NA	CS Instance deleted (Note 4)	For leg p: Originating_S etup For leg c: Stable_1_ Party	For leg p1: CS instance deleted else NA	For leg p: If passive legs = 2 before release Forward else if more than 2 legs Stable_ Passive_ Party For leg c: NA	For leg p: If remaining legs = 1 Stable_2_ Party else Stable_ Multi_Party For leg c: Stable_ Passive_ Party
O_Abandon DP for leg c	CS instance deleted	NA	NA	NA	CS instance deleted	Stable_1_ Party	NA	NA	Stable_ Passive_ Party
Terminating_ Attempt DP for leg p	NA	NA	NA	Terminating_ Setup (Note 5)	NA	NA	NA	NA	NA
Terminating_ Attempt_ Authorized DP for leg p	NA	NA	NA	Terminating_ Setup	NA	NA	NA	NA	NA

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ Party	Stable_ Multi_ Party
Terminating_ Attempt Denied DP for leg p	NA	NA	NA	Terminating_ Setup	NA	NA	NA	NA	NA
Facility_ Selected_and_ Available DP for leg c	NA	NA	NA	Terminating_ Setup	NA	NA	NA	NA	NA
Call_Accepted DP for leg c	NA	NA	NA	Stable_2_ Party	NA	NA	NA	NA	NA
T_Busy DP for leg c	NA	NA	NA	Terminating_ Setup	CS Instance deleted (Note 4)	Terminating_ Setup	NA	NA	Stable_ Passive_ Party
T_No_Answer DP for leg c	NA	NA	NA	Terminating_ Setup	CS Instance deleted (Note 4)	Terminating_ Setup	NA	NA	Stable_ Passive_Party
T_Answer DP for leg c	NA	NA	NA	Stable_2_ Party	1-Party	Stable_2_ Party	NA	NA	Stable_ Multi_Party
T_Suspend DP for leg c	NA	NA	NA	NA	1-Party	Stable_2_ Party	NA	NA	Stable_ Multi_Party
T_Re-answerDP for leg c	NA	NA	NA	NA	1_Party	Stable_2_ Party	NA	NA	Stable_ Multi_Party

Table 15/Q.1248.2 – Transition of DP events to CSCV states

CSCV State → Detection of DP event ↓	Originating_Setup	Originating_1 _Party_Setup	Stable_1 _Party	Terminating _Setup	1_Party	Stable_2_ _Party	Forward	Stable_ Passive_ Party	Stable_ Multi_ Party
T_Midcall DP (T_Active) for leg c and leg p	NA	NA	For leg p: Stable_1_ Party For leg c: NA	NA	For leg c: 1_Party For leg p NA	Stable_2_ Party	For leg p: Forward For leg c: NA	For leg p: Stable_ Passive_ Party For leg c: NA	Stable_ Multi_ Party
T_Disconnect DP for leg c and leg p	NA	NA	CS Instance deleted (Note 4)	NA	CS Instance deleted (Note 4)	For leg p: 1_Party For leg c: Stable_1_ Party	For leg p1: CS instance deleted else NA	For leg p: If passive legs = 2 before release Stable_1_ Party else if > 2 then Stable_ Passive_ Party For leg c: NA	For leg p: If remaining passive legs = 1 Stable_2_ Party else Stable_ Multi_ Party For leg c: Stable_ Passive_ Party
T_Abandon DP for leg p	NA	NA	CS Instance deleted (Note 4)	CS instance deleted	NA	1_Party	For leg p1: CS instance deleted else NA	NA	If remaining passive legs = 1 Stable_2_ Party else Stable_ Multi_ Party

Table 15/Q.1248.2 – Transition of DP events to CSCV states

NA Not Applicable

NOTE 1 – The SLPI in the SCF becomes aware of the CS transition into Stable_2_Party/Stable_1_Party by the reporting of the O_Term_Seized DP/O_Answer DP in case of an O_BCSM. When the SCF sends a CPH operation before having received one of these DP notifications, the operation will be processed as an error.

NOTE 2 – The SLPI in the SCF becomes aware of the CS transition into Stable_2_Party by the reporting of the Call_Accepted DP/T_Answer DP in case of a T_BCSM.

NOTE 3 – In order to secure a correct connection view for service execution, the SLPI in the SCF becomes updated about the CSCV state evolution in the SSF via DP arming and DP reporting.

NOTE 4 – If the DP is armed as an EDP, it is reported to the SCF together with any other pending reports (ApplyChargingReport, CallInformationReport). The CS instance will subsequently be deleted. This occurs immediately in case call processing was not suspended otherwise (DP armed as EDP-R) upon resumption of the suspended call.

NOTE 5 – DP cannot occur as an EDP only as a TDP.

NOTE 6 – With InitiateCallAttempt, the O_BCSM becomes suspended at DP Origination_Attempt_Authorized.

NOTE 7 – Detection of a Route_Select_Failure DP after a Release event which generated a "Route_Failure" event causing a transition from PIC Alerting to PIC Select_Route.

NOTE 8 – The O/T_Modify_Request and O/T_Modify_Result DP do not cause any CSCV state transition.

6.5 Out-Channel Call-Related User Interaction (OCCRUI)

Clause 6.6/Q.1238.2 section is applicable.

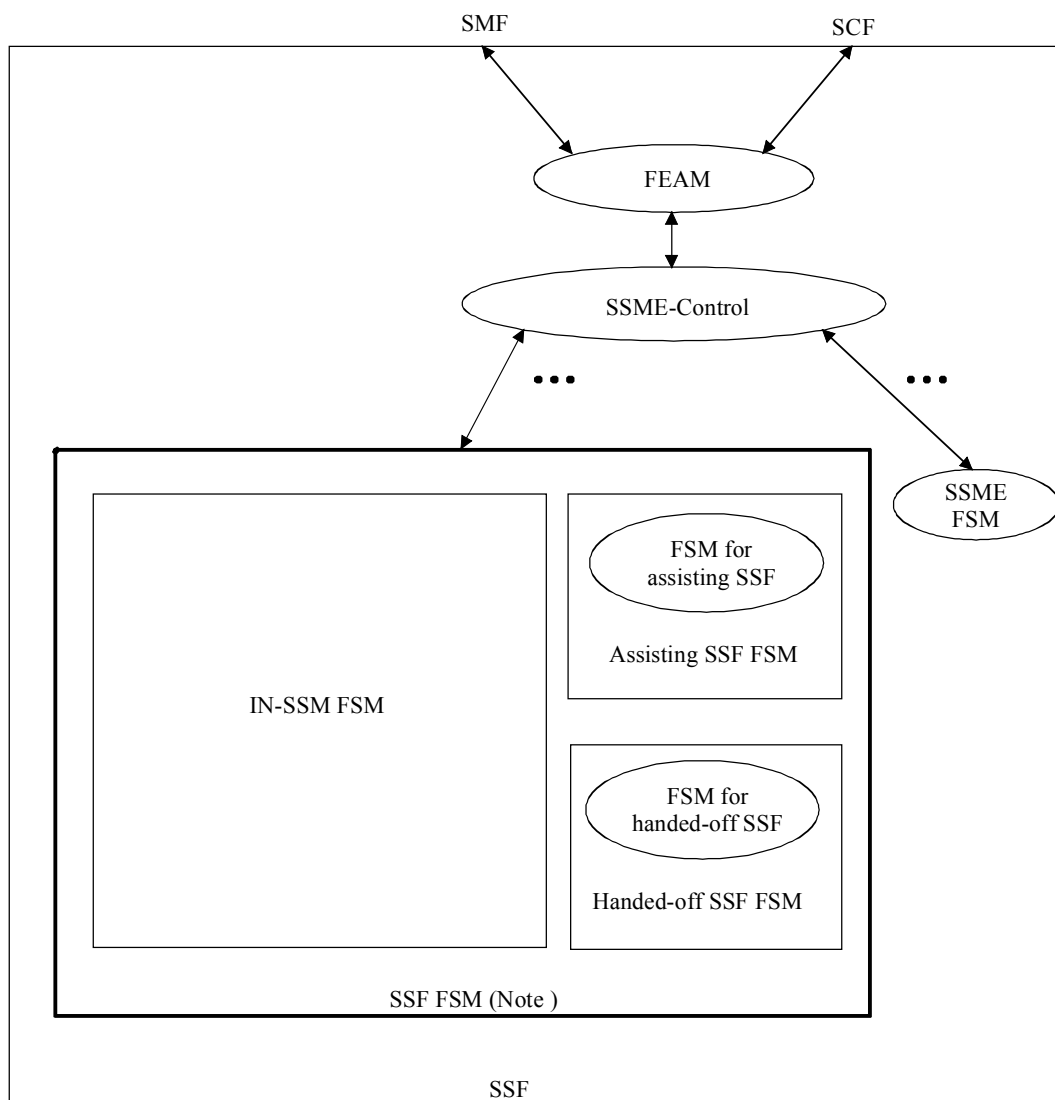
7 SCF Model

Clause 7/Q.1238.2 is applicable.

8 FSM for SSF

Clause 8/Q.1238.2 is applicable with the following exceptions:

- Clause 8.1/Q.1238.2 is replaced by 8.1 of this Recommendation;
- Figure 37/Q.1238.2 is replaced with Figure 23.



NOTE – One of the three possible FSMs (either IN-SSM, assisting SSF or handed-off SSF) is selected.

T11107200-02

Figure 23/Q.1248.2 – SSF FSM structure

8.1 SSF management finite state model (SSME-FSM)

The SSME-FSM state diagram is described in Figure 24.

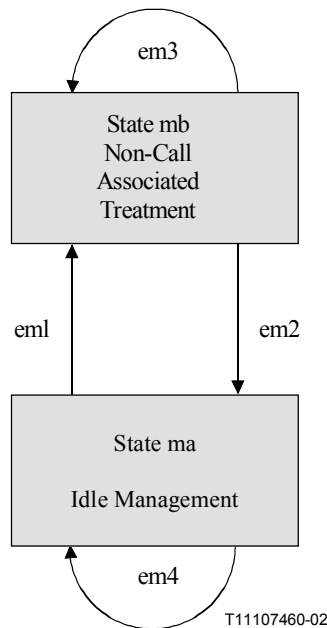


Figure 24/Q.1248.2 – SSME-FSM state diagram

The SSME-FSM is independent of the individual FSMs for CS.

In the Idle Management state, the following operations may be received from the SCF and processed by the SSME-FSM with no resulting transition to a different state (transition em4):

- **RequestCurrentStatusReport;**
- **ManageTriggerData;**
- **CreateOrRemoveTriggerData;**
- **CallFiltering.**

The ManageTriggerData, CreateOrRemoveTriggerData and CallFiltering operations can only be received outside a call context transaction.

The Non-Call Associated Treatment state is entered from the Idle Management state when one of the following non-call associated operations is received (transition em1):

- **RequestEveryStatusChangeReport;**
- **RequestFirstStatusMatchReport;**
- **ActivateServiceFiltering;**
- **CallGap;**
- **MonitorRouteRequest.**

The CallGap operation can be received inside, as well as outside, a call context transaction. The ActivateServiceFiltering, MonitorRouteRequest and CallFiltering operations can be received outside a call context transaction only.

During this state the following events can occur:

- given that service filtering or route monitoring is active, the SSF shall send a service filtering response to the SCF: the SSME-FSM remains in this state (transition em3);

- given that service filtering or route monitoring is active, the SSF shall increment a counter; the SSME-FSM remains in this state (transition em3);
- given that service filtering is active and the service filtering duration expires: the SSME shall send a **ServiceFilteringResponse** operation to the SCF; the SSME-FSM moves to the Idle Management state (transition em2);
- given that route monitoring is active and the service filtering duration expires: the SSME shall send a **MonitorRouteReport** operation to the SCF; the SSME-FSM moves to the Idle Management state (transition em2);
- given that status report is active, as previously requested by a RequestEveryStatusChangeReport operation, the SSF sends a **StatusReport** operation with reportCondition (optional) when the resource status changes; the SSME-FSM remains in this state (transition em3);
- given that status report is active, as previously requested by a RequestFirstStatusMatchReport operation, the SSF sends a **StatusReport** operation with reportCondition (optional) when the resource status matches; the SSME-FSM transits to Idle Management state (transition em2);
- given that status report is active, as previously requested by a RequestFirstStatusMatchReport or a RequestEveryStatusChangeReport operation, the SSF sends a StatusReport operation with reportCondition set to "cancelled" when the monitor is cancelled by receiving a CancelStatusReportRequest operation from the SCF; the SSME-FSM moves to the Idle Management state (transition em2);
- given that status report is active, as previously requested by a RequestFirstStatusMatchReport or a RequestEveryStatusChangeReport operation, the SSF sends a StatusReport operation with reportCondition set to "timerExpired" when the monitor duration expires; the SSME-FSM moves to the Idle Management state (transition em2);
- if call gap related duration timer expires, the SSME-FSM moves to the Idle Management state (transition em2);
- given that call gap/service filtering is active, another CallGap/ActivateServiceFiltering operation having the same gapping/filtering criteria can be received by the SSF: the second "filter" or "gap" replaces the first one (transition em3) unless the duration timer value is equal to zero, in which case the SSME-FSM moves to the Idle Management state (transition em2).

All other operations have no effect on the SSME-FSMs; the operations are passed by the SSME-Control to the relevant FSM for CSA.

9 FSM for SCF

Clause 9/Q.1238.2 is applicable.

10 FSM for USI

Clause 10/Q.1238.2 is applicable.

11 Operation procedures

Clause 11/Q.1238.2 is applicable with the enhancements specified in this Recommendation.

11.1 Modified operations

The following operations defined in ITU-T Q.1238.2 are modified by this Recommendation.

11.1.1 AnalyzeInformation

Table 20/Q.1238.2 is replaced with Table 16:

Table 16/Q.1248.2 – Transition for CS

Operation ⇒		Analyse-Information
CSCV State (original state)	Originating Setup	Stable 2-Party (Notes 1 and 3)
	Originating 1-Party Setup	Error (Originating 1-Party Setup) or Stable 1-Party (Notes 1 and 2)
	Stable 1-Party	Forward (Stable Passive Party (Note 1))
	Terminating Setup	Error (Terminating Setup)
	1-Party	Originating Setup (Stable 2-Party (Note 1)) or Error (1-Party) (Note 5)
	Stable 2-Party	Error (Stable 2-Party)
	Forward	Forward (Note 2) (Stable Passive Party (Note 1))
	Stable Passive Party	Error (Stable Passive Party)
	Stable Multi-Party	Error (Stable Multi-Party)
<p>NOTE 1 – State transition occurs not until DP O_Term_Seized (i.e. when the CallProgress.req.ind (bptyAlerted) is received). If no such DP is encountered, the transition is detected at DP O_Answer (i.e. as Setup.resp.conf (answer) is received from the called destination (automatic answer case)).</p> <p>NOTE 2 – In case of an unsuccessful call setup, the AnalyseInformation operation is allowed when call processing is suspended at the following DPs for leg p: O_Called_Party_Busy, Authorize_Route_Failure, Route_Select_Failure, O_No_Answer, O_Disconnect.</p> <p>NOTE 3 – Not allowed in O_Mid_Call DP (i.e. after entering Send_Call PIC)</p> <p>NOTE 5 – The AnalyseInformation operation is only allowed in the 1_Party CSCV state when the controlling leg is connected to an O_BCSM: when the controlling leg is connected to a T_BCSM an error procedure is invoked.</p>		

11.1.2 CallGap

11.1.2.1 General description

This operation is used to request the SSF to reduce the rate at which specific service requests are sent to the SCF.

The operation applies to a request for filtering of service requests (i.e. as opposed to filtering of calls).

11.1.2.2 Parameters

11.1.2.2.1 Argument Parameters

The operation argument consists of the following parameters. These parameters are defined in clause 12.

- gapCriteria;
- gapIndicators;
- controlType;
- gapTreatment;
- extensions.

11.1.2.3 Invoking entity (SCF)

11.1.2.3.1 Normal procedure

SCF Precondition:

- 1) The SCF detects an overload condition persists and IN service request gapping has to be initiated at the SSF; or
the SCF receives a manually initiated service gapping request.

SCF Postcondition:

- 1) The **SCME-FSM** remains in the same state upon issuing the "CallGap" operation.

A congestion detection and control algorithm monitors the load of SCP resources. After detection of a congestion situation, the parameters for the "CallGap" operation are provided.

If the congestion level changes, new "CallGap" operations may be sent for active gap criteria but with new gap interval. If no congestion is detected, gapping may be removed.

A manual initiated call gap will prevail over an automatic initiated call gap.

11.1.2.3.2 Error handling

Operation related error handling is not applicable, due to class 4 operation.

11.1.2.4 Responding entity (SSF)

11.1.2.4.1 Normal procedure

SSF Precondition:

- 1) Service request gapping for gapCriteria is not active, or; Service request gapping for gapCriteria is active.

SSF Postconditions:

- 1) The **SSME-FSM** is in the state "Non-call associated treatment".
- 2) Service request gapping for gapCriteria is activated; or Service request gapping for gapCriteria is renewed; or Call gapping for gapCriteria is removed.

If there is no already existing **SSME-FSM** for the gap criteria provided, a new **SSME-FSM** is created. This **SSME-FSM** enters the state "Non-call associated treatment" and initializes service request gapping for the specified IN calls. The parameters "gapIndicators", "controlType" and "gapTreatment" for the indicated gap criteria will be set as provided by the "CallGap" operation.

If the gapTreatment parameter is not present, the SSF will use a default treatment depending on network operator implementation. The default treatment can, e.g. either be to release call or continue call processing.

In case both manually initiated and automatically initiated service request gapping are active for the same "gapCriteria", the manuallyInitiated call gapping will prevail over automatically initiated ("sCPOverloaded"). More specifically, the following rules will be applied in the SSF to manage the priority of different control Types associated with the same "gapCriteria":

- If an **SSME-FSM** already exists for the "gapCriteria" provided, then:
 - 1) If the (new) "controlType" equals an existing "controlType", then the new parameters (i.e. "gapIndicators" and "gapTreatment") will overwrite the existing parameter values.
 - 2) If the (new) "controlType" is different than the existing "controlType", then the new parameters (i.e. "controlType", "gapIndicators", and "gapTreatment") will be appended to the appropriate **SSME-FSM** (in addition to the existing parameters). The **SSME-FSM** remains in the state "Non-Call Associated Treatment".

If the SSF meets a TDP, it will check if service request gapping was initiated, either for the "serviceKey", or for the "calledAddressValue" assigned to this TDP. If not, an "InitialDP" or a DP-specific operation can be sent. In case service request gapping was initiated for "calledAddressAndService", or "callingAddressAndService", and the "serviceKey" matches, a check on the "calledAddressValue" and "callingAddressValue" and optionally "locationNumber" for active service request gapping, is performed. If not, an "InitialDP" or a DP-specific operation can be sent.

In case of gapping on "callingAddressAndService" and the parameter "locationNumber" is present, gapping will be performed on "locationNumber" instead of "callingAddressValue".

If a call to a controlled number matches only one "gapCriteria", then the corresponding control is applied.

If both "manuallyInitiated" and "sCPOverload" controls are active, then only the manually initiated control will be applied.

If a call matches several active "gapCriteria", then the treatment as specified in the CallGap associated with the gapCriteria with the highest priority should be applied, with the priority being from high to low:

- 1) calledAddressAndService/calledAddressValue;
- 2) callingAddressAndService;
- 3) gapOnService;
- 4) gapAllINTraffic.

For example, a call with called number 123456 and ServiceKey = NP matches two CallGaps, one with gapCriteria "CalledAddressValue = 123" and another with "gapOnService = NP". Then the call is subject to the control of the service request CallGap with "CalledAddressValue = 123".

In case multiple call gapping procedures are active with the same gap criteria, the "manuallyInitiated" service request gapping shall prevail over automatically initiated service request gapping (sCPOverloaded).

If a call to a controlled called number or from a controlled calling number matches several active "gapCriteria" of the same type (in this context "calledAddressAndService" and "calledAddressValue" are seen as one type), then only the "gapCriteria" associated with the longest called respective calling party number should be used, and the corresponding control should be applied.

For example, the codes 1234 and 12345 are under control. Then the call with 123456 is subject to the control on 12345.

If a call to a controlled called number matches calledAddressAndService and calledAddressValue with the same number length, then calledAddressAndService has priority. Furthermore, if both "manuallyInitiated" and "sCPOverloaded" "controlTypes" are active for this "gapCriteria", then the "manuallyInitiated" control will be applied.

If service request gapping is performed on a call for a particular service and triggering of this service is allowed, apart from the case that "gapAllINTraffic" is active, no other gap criteria should be applied to the same service.

If "gapAllINTraffic" is active, then the checks for other criteria will be applied as described above. After these checks, control according to "gapAllINTraffic" will be applied for every IN call not blocked by other active criteria.

Active GapCriteria with assigned scfID will have higher priority than the others. In case an entry with scfID matching the current call exist all other criteria without scfID are not evaluated.

The matching entries with scfID are evaluated in accordance with the priority rules for the basic criteria listed above.

If service request gapping shall be applied and there is no service request gap interval active, an "InitialDP" or a DP-specific operation can be sent including the "cGEncountered" parameter according to the specified controlType. A new possible service request interval will be initiated as indicated by "gapInterval".

If a service request interval is active, no "InitialDP" or a DP-specific operation is sent and the call is treated as indicated by "gapTreatment".

The service request gap process is stopped if the indicated duration equals ZERO.

If the call gapping proceeds then the **SSME-FSM** remains in the state "Non-call associated treatment". Otherwise, the **SSME-FSM** moves to state "idle management".

11.1.2.4.2 Error handling

Operation-related error handling is not applicable, due to class 4 operation.

11.1.3 Connect

The following parameters defined in clause 12 are added to the operation argument:

- calledPartySubAddress
- connectionIdentifier
- genericIdentifier
- qoSParameter
- bISDNParameters
- cug-Interlock
- cug-OutgoingAccess
- ipRelatedInformation

Table 21/Q.1238.2 is replaced with Table 17:

Table 17/Q.1248.2 – Transition for CS

	Operation ⇒	Connect
CSCV State (original state)	Originating Setup	Stable 2-Party (Notes 1 and 3)
	Originating 1-Party Setup	Error (Originating 1-Party Setup) (Note 7) or Stable 1-Party (Notes 1 and 2)
	Stable 1-Party	Forward (Stable Passive Party (Note 1))
	Terminating Setup	Forward (Stable Passive Party (Note 1)) or Error (Terminating Setup) (Note 5)
	1-Party	Originating Setup (Stable 2-Party Note 1) or Error (1-Party) (Note 6)
	Stable 2-Party	Error (Stable 2-Party)
	Forward	Forward (Note 2) (Stable Passive Party (Note 1))
	Stable Passive Party	Error (Stable Passive Party)
	Stable Multi-Party	Error (Stable Multi-Party)

Table 17/Q.1248.2 – Transition for CS

NOTE 1 – State transition occurs not until DP O_Term_Seized (i.e. when the CallProgress.req.ind (bptyAlerted) is received). If no such DP is encountered the transition is detected at DP O_Answer (i.e. as Setup.resp.conf (answer) is received from the called destination (automatic answer case)).

NOTE 2 – In case of an unsuccessful call setup, the Connect operation is allowed when call processing is suspended at the following DPs for leg p: O_Called_Party_Busy, Authorize_Route_Failure, Route_Select_Failure, O_No_Answer, O_Disconnect.

NOTE 3 – Not allowed in O_Mid_Call DP (i.e. after entering Send_Call PIC).

NOTE 5 – Only allowed if call processing is suspended at the following DPs: Facility_Selected_And_Available, Termination_Attempt, Termination_Attempt_Authorized, T_Busy, T_No_Answer.

NOTE 6 – The Connect operation is only allowed in the 1_Party CSCV state when the controlling leg is connected to an O_BCSM, when the controlling leg is connected to a T_BCSM an error procedure is invoked.

NOTE 7 – An error TaskRefused is provided if Connect (and not Continue/ContinueWithArgument) is received to resume call setup for a leg to be created following the InitiateCallAttempt.

11.1.4 ConnectToResource

The following clarification applies to the resourceAddress parameter:

For the Assist/Handed –Off SSF case, only the "ipRoutingAddress" and the "none" alternatives are used. Other alternatives are ignored.

11.1.5 ContinueWithArgument

The following parameters defined in clause 12 are added to the operation argument:

- connectionIdentifier
- cug-Interlock
- cug-OutgoingAccess
- ISDNAccessRelatedInformation
- originalCalledPartyID
- callingPartyNumber
- callingPartysCategory
- redirectingPartyID
- redirectionInformation
- forwardCallIndicators
- genericNumbers
- ipRelatedInformation

11.2 CreateOrRemoveTriggerData procedure

11.2.1 General description

The SCF uses this operation to create a new trigger at a particular DP by downloading to the SSF the required triggering information (triggering criteria, service Key). The operation also allows the removal of a previously created non-active trigger.

11.2.1.1 Parameters

11.2.1.1.1 Argument Parameters

The operation argument consists of the following parameters. These parameters are defined in clause 12.

- createOrRemove: If this parameter is omitted, the default is to "create" a new trigger;
- dPName;
- triggerDPType;
- serviceKey: This parameter is mandatory if the request is to "create" a new trigger;
- profile: This parameter is mandatory if the request is to "create" a new trigger;
- triggerData;
- defaultFaultHandling;
- tDPIIdentifier: This parameter is mandatory if the request is to "remove" a trigger;
- extensions.

11.2.1.1.2 Result Parameters

The operation result consists of the following parameters. These parameters are defined in clause 12.

- triggerStatus;
- tDPIIdentifier;
- registratorIdentifier;
- extensions.

11.2.2 Invoking entity (SCF)

11.2.2.1 Normal procedure

SCF precondition:

- 1) SLPI receives an indication (e.g. by the SMF) to create a new TDP and downloads to the SSF triggering information (triggering criteria, service Key, SCF address, etc.) associated with the TDP criteria. The SLPI may also receive an indication to remove an existing trigger and provides the SSF with the TriggerIdentifier for the trigger to be removed.

SCF postcondition:

- 1) SCME is in the state "Idle".

If createOrRemoveTriggerData has been successfully processed, the returnResult indicates the tDPIIdentifier assigned to the new created or removed trigger (TDP). The SCF indicates to the initiating entity the result of the procedure.

11.2.2.2 Error handling

Generic error handling for the operation-related error is described in clause 13 and the TCAP services used for reporting operation errors are described in clause 15.

11.2.3 Responding entity (SSF)

11.2.3.1 Normal procedure

SSF precondition:

- 1) None.

SSF postconditions:

- 1) **SSME-FSM** is in the state "Idle".
- 2) The result or an error indication of "createOrRemoveTriggerData" is sent as ReturnResult to the initiating SCF.

If a new trigger is to be created it is checked whether the profile addressed by the operation exists. If so, the trigger is created. The trigger status is "deactivated". Activation may be performed using the ManageTriggerData operation.

If an existing trigger is to be removed, it is checked if the triggerIdentifier is valid and if the trigger to be removed is not active (i.e. trigger status is "deactivated"). If so, the trigger is removed.

The request to create an already existing Trigger is not an error case. The trigger status result will be set to "trigger already exists".

If a request is received to remove a Trigger which is not known by the SSF, no error is returned. The trigger status result will be set to "unknown trigger".

11.2.3.2 Error handling

NOTE – In case of an invalid registratorIdentifier, a TaskRefused error is returned.

Generic error handling for the operation related error is described in subclause 13 and the TCAP services used for reporting operation errors are described in clause 15.

11.2.4 DisconnectLeg

Table 22/Q.1238.2 is replaced with Table 18.

Table 18/Q.1248.2 – Transition table for the CS

Operation ⇒		DisconnectLeg (c)	DisconnectLeg (p)
CSCV State (original state)	Originating Setup	CS instance deleted	Error (Originating Setup)
	Originating 1-Party Setup	Error (Originating 1-Party Setup)	Error (Originating 1-Party Setup)
	Stable 1-Party	Error (Stable 1-Party)	CS instance deleted (Note 4)
	Terminating Setup	Error (Terminating_Setup)	CS instance deleted
	1-Party	CS instance deleted	Error (1-Party)
	Stable 2-Party	Stable 1-Party	1-Party (Note 4)
	Forward	Error (Forward)	CS instance deleted/Error (Note 1)
	Stable Passive Party	Error (Stable Passive Party)	Stable 1-Party/Stable Passive Party (Note 3)
	Stable Multi-Party	Stable Passive Party	Stable 2-Party/Stable Multi-Party (Note 3)

NOTE 1 – Delete CS instance if passive leg status is joined. If passive leg 'pending' then Error case.

NOTE 3 – State depends on the number of remaining legs.

NOTE 4 – The transition to Stable_2_Party/Stable_1-Party in case of an O_BCSM occurs at call progress (alert)l and is reflected by an associated DP. The SCF becomes aware of this CSCV transition into Stable_2_Party/Stable_1-Party by a report of the O_Term_Seized/O_Answer DP.

The transition to Stable_2_Party/1-Party in case of an T_BCSM occurs at call progress (alert) and is reflected by an associated DP. The SCF becomes aware of this CSCV transition into Stable_2_Party/1_Party by a report of the Call_Accepted/T_Answer DP. If correct service execution is to be secured, the SCF shall not sent the CPH operation before it becomes aware of the present CSCV state in the SSF.

11.2.5 InitialDP

The following parameters defined in clause 13 are added to the operation argument:

- calledPartySubAddress
- connectionIdentifier
- genericIdentifier
- qoSParameter
- bISDNParameters
- GlobalCallReference
- cug-Index
- cug-Interlock
- cug-OutgoingAccess
- ipRelatedInformation

11.2.6 InitiateCallAttempt

The following parameters defined in clause 13 are added to the operation argument:

- bearerCapability
- calledPartySubAddress
- connectionIdentifier
- genericIdentifier
- qoSParameter
- bISDNParameters
- originalCalledPartyID
- callingPartysCategory
- redirectingPartyID
- redirectionInformation
- displayInformation
- forwardCallIndicators
- genericNumbers
- forwardGVNS
- cug-Interlock
- cug-OutgoingAccess
- incomingSignallingBufferCopy
- ipRelatedInformation

11.2.7 ManageTriggerData

The definition of the operation is modified as follows:

This operation is used to activate, deactivate or retrieve the status of one or several trigger detection points linked to a profile known at a switch, e.g. related to an access line. This operation is used for service logic controlled IN management purposes. The status or a success indication is sent to the SCF as Return Result of this operation.

11.2.8 MergeCallSegments

The following parameters defined in clause 12 are added to the operation argument:

- mergeSignallingPaths

Table 25/Q.1238.2 is replaced with Table 19:

Table 19/Q.1248.2 – Transition for Target CS

Operation (target CS original state) ⇒		MergeCall Segments Originating_Setup	MergeCall Segments Orig_1 Party Setup	MergeCall Segments Stable 1-Party	MergeCall Segments Terminating-Setup	MergeCall Segments 1_Party	MergeCall Segments Stable 2-Party	MergeCall Segments Forward	MergeCall Segments Stable Passive Party	MergeCall Segments Stable Multi-Party
CSCV State (source CS original state)	Originating Setup	Error (Originating Setup)	Error (Originating Setup)	Stable 2-Party	Error (Originating Setup)	Error (Originating Setup)	Error (Originating Setup)	Error (Originating Setup)	Stable Multi-Party	Error (Originating Setup)
	Originating 1 Party Setup	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)	Error (Originating 1 Party Setup)
	Stable 1-Party	Stable 2-Party	Error (Stable 1-Party)	Stable Passive Party	Error (Stable 1-Party)	Stable 2-Party	Stable Multi-Party	Error (Stable 1-Party)	Stable Passive Party	Stable Multi-party
	Terminating Setup	Error (Terminating Setup)	Error (Terminating Setup)	Error (Terminating Setup) or Stable Passive Party (Note 3)	Error (Terminating Setup)	Error (Terminating Setup) or Stable_2_Party (Notes 3 and 4)	Error (Terminating Setup)	Error (Terminating Setup)	Error (Terminating Setup)	Error (Terminating Setup)
	1-Party	Error (1-Party)	Error (1-Party)	Stable 2-Party	Error (1-Party)	Error (1-Party)	Error (1-Party)	Error (1-Party)	Stable Multi-party	Error (1-Party)

Table 19/Q.1248.2 – Transition for Target CS

Operation (target CS original state) ⇒		MergeCall Segments Originating_Setup	MergeCall Segments Orig_1 Party Setup	MergeCall Segments Stable 1-Party	MergeCall Segments Terminating_Setup	MergeCall Segments 1_Party	MergeCall Segments Stable 2-Party	MergeCall Segments Forward	MergeCall Segments Stable Passive Party	MergeCall Segments Stable Multi-Party
CSCV State (source CS original state)	Stable 2-Party	Error (Stable 2-Party)	Error (Stable 2-Party)	Stable Multi-Party	Error (Stable 2-Party)	Error (Stable 2-Party)	Error (Stable 2-Party)	Error (Stable 2-Party)	Stable Multi-party	Error (Stable 2-Party)
	Forward	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)	Error (Forward)
	Stable Passive Party	Stable M-Party	Error (Stable Passive Party)	Stable Passive Party	Error (Stable Passive Party)	Stable Multi-party	Stable Multi-party	Error (Stable Passive Party)	Stable Passive Party	Stable Multi-party
	Stable Multi-Party	Error (Stable Multi-Party)	Error (Stable Multi-Party)	Stable Multi-Party	Error (Stable Multi-Party)	Error (Stable Multi-Party)	Error (Stable Multi-Party)	Error (Stable Multi-Party)	Stable Multi-Party	Error (Stable Multi-Party)
NOTE 3 – For the source CSCV state Terminating_Setup the T_BCSM is suspended at any valid DP for this CSCV state, else an error case.										
NOTE 4 – The controlling leg becomes associated to a BCSM whose type is not necessary compatible with that indicated by the BCSM type attribute.										

11.2.9 MoveCallSegments

The SSF Postconditions are modified as follows:

- 1) The SSF performs the necessary actions to move the indicated Call Segment(s) into the target CSA. All legs with the status "joined" of the source CS are now connected to the Connection Point of the new CS in the target CSA.
- 2) The FSM **for Call Segment** of the moved Call Segment(s) in the target CSA remains in the same state
- 3) If the MoveCallSegments operation results in a source CSA with no remaining Call Segments, the source CSA is deleted.
- 4) The CSA CV is in state "Multi-Segment" or "One Segment" dependent on the number of CSs in the CSA.
- 5) A Return result is sent immediately after the successful change of the call segment(s) configuration is executed.

Table 27/Q.1238.2 is replaced with Table 20:

Table 20/Q.1248.2 – Target CSA: CSCV state transition for the new CS moved from the source CSA into the target CSA

Operation and precondition ⇒		MoveCallSegments: Target CSA contains a CS with a controlling leg	MoveCallSegments: Target CSA contains no CS with a controlling leg
Source CSA Moved CS: CSCV State (original state) Target CSA: New CS: CSCV State (matrix shows result)	Originating Setup	Error (Originating Setup)	Originating_Setup
	Originating 1-Party Setup	Originating 1 Party Setup	Originating 1 Party Setup
	Stable 1-Party	Stable 1-Party	Stable 1-Party
	Terminating Setup	Terminating Setup (Notes 4 and 5) or Error (Terminating Setup)	Terminating Setup (Note 4) or Error (Terminating Setup)
	1-Party	Error (1-Party)	1-Party
	Stable 2-Party	Not Applicable Error (Stable 2-Party)	Stable 2-Party
	Forward	Forward	Forward
	Stable Passive Party	Stable Passive Party	Stable Passive Party
	Stable Multi-Party	Not Applicable Error (Stable Multi-Party)	Stable Multi-Party
<p>NOTE 4 – The T_BCSM in the moved CS is suspended at any valid DP for this CSCV state, else an error case. However, call processing cannot be resumed (i.e. using a Continue, ContinueWithArgument operation is not allowed) for the moved CS, but the passive leg in the moved CS may become connected to a joined leg in another CS (MoveLeg, MergeCallSegments).</p> <p>NOTE 5 – This transition provides an exception case with more than one controlling leg in the CSA. It implies the transient existence in the CSA of the controlling leg in the moved CS together with a joined/pending controlling leg in another CS.</p>			

11.2.10 MoveLeg

The following parameters defined in clause 13 are added to the operation argument:

- detachSignallingPath
- exportSignallingPath

Table 30/Q.1238.2 is replaced with Table 21:

Table 21/Q.1248.2 – Transition for source CS

	Operation ⇒	MoveLeg (c)	MoveLeg (p)
CSCV State (original state)	Originating Setup	CS instance deleted or Error (Originating Setup) (Note 3)	Error (Originating Setup)
	Originating 1-Party Setup	Error (Originating 1-Party Setup)	Error (Originating 1-Party Setup)
	Stable 1-Party	Error (Stable 1-Party)	CS instance deleted
	Terminating Setup	Error (Terminating Setup)	Error (Terminating Setup) or CS instance deleted (Note 4)
	1-Party	CS instance deleted	Error (1-Party)
	Stable 2-Party	Stable 1 Party	1-Party
	Forward	Error (Forward)	CS instance deleted or Error (Forward)
	Stable Passive Party	Error (Stable Passive Party)	Stable 1-Party/Stable Passive Party (Note 2)
	Stable Multi-Party	Stable Passive Party	Stable 2-Party/Multi-Party (Note 2)

NOTE 2 – State depends on the number of remaining legs.
 NOTE 3 – The O_BCSM is suspended at any valid DP for this CSCV state before the O_BCSM has sent a SetupReqInd, else an error case.
 NOTE 4 – The T-BCSM is suspended at any valid DP for this CSCV state, else error case.

Table 31/Q.1238.2 is replaced with Table 22:

Table 22/Q.1248.2 – Transition for Target CS

	Operation ⇒	MoveLeg (c)	MoveLeg (p)
CSCV State (original state)	Originating Setup	Error (Originating Setup)	Stable 2-Party
	Originating 1-Party Setup	Error (Originating 1-Party Setup)	Error (Originating 1-Party Setup)
	Stable 1-Party	Stable 2-Party	Stable Passive Party
	Terminating Setup	Error (Terminating Setup)	Error (Terminating Setup)
	1-Party	Error (1-Party)	Stable 2-Party
	Stable 2-Party	Error (Stable 2-Party)	Stable Multi-Party
	Forward	Error (Forward)	Error (Forward)
	Stable Passive Party	Stable Multi-Party	Stable Passive Party
	Stable Multi-Party	Error (Stable Multi-Party)	Stable Multi-Party

11.2.11 RequestReportBCSMEvent

The following DPs are added to the arming tables for the B-ISDN modify capability.

Table 23/Q.1248.2 – DP Arming Table for O-BCSM

O_BCSM	Controlling leg	Passive leg	Default leg ID
O_Modify_Request DP	X	X	_ ^{o2)}
O_Modify_Result DP	X	X	_ ^{o2)}
^{o2)} The "legID" parameter shall be included. X Arming Applicable – Not Applicable			

Table 24/Q.1248.2 – DP Arming Table for T-BCSM

T_BCSM	Controlling leg	Passive leg	Default leg ID
T_Modify_Request DP	X	X	_ ^{t2)}
T_Modify_Result DP	X	X	_ ^{t2)}
^{t2)} The "legID " parameter shall be included. X Arming Applicable – Not Applicable			

11.2.12 SelectFacility

The following parameter defined in clause 12 is added to the operation argument:

- ipRelatedInformation

11.2.13 SelectRoute

The following parameter defined in clause 12 is added to the operation argument:

- ipRelatedInformation

11.2.14 SplitLeg

The following parameter defined in clause 12 is added to the operation argument:

- detachSignallingPath

Table 34/Q.1238.2 is replaced with Table 25:

Table 25/Q.1248.2 – Transition for source CS

Operation ⇒		SplitLeg (c)	SplitLeg (p)
CSCV State (original state)	Originating Setup	Error (Originating Setup)	Error (Originating Setup)
	Originating 1-Party Setup	Error (Originating 1-Party Setup)	Error (Originating 1-Party Setup)
	Stable 1-Party	Error (Stable 1-Party)	Error (Stable 1-Party)
	Terminating Setup	Error (Terminating_Setup)	Error (Terminating_Setup)
	1-Party	Error (1-Party)	Error (1-Party)
	Stable 2-Party	Stable 1 Party	1-Party
	Forward	Error (Forward)	Error (Forward)
	Stable Passive Party	Error (Stable Passive Party)	Stable 1-Party/Stable Passive Party (Note 2)
	Stable Multi-Party	Stable Passive Party	Stable 2-Party/Stable Multi-Party (Note 2)
NOTE 2 – The state depends on the number of remaining legs.			

Table 35/Q.1238.2 is replaced with Table 26:

Table 26/Q.1248.2 – Transition for New CS

Operation ⇒		SplitLeg (c)	SplitLeg (p)
CSCV State (original state)	Null	1-Party	Stable 1-Party (Note)
NOTE – The receipt of a SplitLeg (p) operation for the Stable 2-Party and Stable Multi-Party states shall create a new Call Segment in the CSCV Stable 1 Party state. The receipt of a SplitLeg (p) operation for the Stable Passive Party state shall create a new Call Segment in the CSCV Stable_1_Party state.			

11.3 New operations

The following new operations are defined by this Recommendation.

11.3.1 CallFiltering procedure

11.3.1.1 General description

The CallFiltering operation is used to allow the SCF to influence basic call gapping procedures based in the CCF, by sending information from the SCF to the SSF. The SSF relays the received information transparently to the CCF. This way, the SCF can influence the rate at which call attempts are allowed through. The operation thus influences the filtering of calls, as opposed to service requests as is done by the Callgap operation.

11.3.1.2 Argument Parameters

The operation argument consists of the following parameters. These parameters are defined in clause 13.

- destination Index: This index is a pointer to the Destination (see ITU-T E.410) to which calls are filtered;
- gapIndicators: The parameter contains the gapDuration and the gapInterval;
- registratorIdentifier: This parameter identifies the SCF and is to be used by the SSF to verify that the SCF is allowed to influence CCF-based call gapping procedures;
- extensions.

11.3.1.3 Invoking entity (SCF)

11.3.1.3.1 Normal procedure

SCF Preconditions:

- 1) The SCF receives an indication from the SMF an overload condition persists and callfiltering has to be initiated at the SSF.

SCF Postcondition:

- 1) The SCME is in the state "idle".

If the congestion level changes, new "CallFiltering" operations may be sent for active filter criteria, but with a new filter interval. If the congestion situation has ended, the filtering criteria may be removed.

11.3.1.3.2 Error handling

Generic error handling for the operation-related error is described in clause 5, and the TCAP services which are used for reporting operation errors are described in clause 8.

11.3.1.4 Responding entity (SSF)

11.3.1.4.1 Normal procedure

SSF precondition:

- 1) None.

SSF postcondition:

- 1) SSME-FSM is in the state "Idle".

The SSF relays the received information transparently to the CCF-based call filtering process. In case callfiltering to the specific destinations is already active at the CCF, then the new gapIndicator parameter overwrites the existing parameter values.

A manual initiated call filter will prevail over an automatic initiated call filter.

If a call matches several destinationIndexes, then the control corresponding with the most specific destinationIndex should be applied.

The service request gap process is stopped if the indicated duration equals zero.

11.3.1.4.2 Error handling

Generic error handling for the operation-related error is described in clause 6, and the TCAP services which are used for reporting operation errors are described in clause 9.

NOTE – In case of error (i.e. invalid registrator identifier,), a TaskRefused error is returned.

11.3.2 MonitorRouteRequest procedure

11.3.2.1 General description

When receiving this operation, the SSF monitors specified routes for DP4 hits (RouteSelectFailure). The SSF transfers the results of the operation, either when a certain threshold is reached (Monitoring Criteria), or when the stop time is reached.

11.3.2.2 Argument Parameters

The operation argument consists of the following parameters. These parameters are defined in clause 13.

- routeList: This parameter is used to specify the outgoing trunk group to monitor. A sequence of routes may also be provided
- correlationID: This parameter is used by the SCF to associate the MonitorRouteRequest with the MonitorRouteReport sent by the SSF.
- monitoringCriteria: This parameter specifies when the monitoring results are to be sent to the SCP.
- monitoringTimeout: This parameter indicates the duration, during which the monitoring should be active.
- startTime: This parameter defines when filtering is started. If "startTime" is not provided or was already met, the SSF starts monitoring immediately.
- Extensions.

11.3.2.3 Invoking entity (SCF)

11.3.2.3.1 Normal Procedure

SCF precondition:

- 1) SLPI decides to monitor traffic over specific routes.

SCF postconditions:

- 1) SLPI starts an application timer to monitor the expected end of monitoring routes
- 2) The SCME is in the state "Waiting for SSF MonitoringRouteReport".

11.3.2.3.2 Error handling

Generic error handling for the operation-related errors is described in clause 13 and the TCAP services used for reporting operation errors are described in clause 15.

11.3.2.4 Responding entity (SSF)

11.3.2.4.1 Normal procedure

SSF precondition:

- 1) None.

SSF postcondition:

- 1) The SSME-FSM is in the state "Non-call Associated Treatment".

11.3.2.4.2 Error handling

Generic error handling for the operation-related errors is described in clause 13 and the TCAP services used for reporting operation errors are described in clause 15.

11.3.3 MonitorRouteReport Procedure

11.3.3.1 General description

This operation is used to report the values of route list counters of a previous MonitorRouteRequest operation.

11.3.3.2 Argument Parameters

The operation argument consists of the following parameters. These parameters are defined in clause 13.

- routeCounters: The parameter contains the count of failed calls over selected routes, during the monitoring period. It is a list of route counter identifications and the related value. If a threshold was specified for a particular route, then the counter value holds the specified threshold value.
- correlationID: This parameter is used by the SCF to associate the MonitorRouteReport from a previously sent MonitorRouteRequest.
- extensions.

11.3.3.3 Invoking entity (SSF)

11.3.3.3.1 Normal Procedure

SSF preconditions:

- 1) Monitoring of routes is currently running and the interval timer has expired; or
- 2) Monitoring of routes is running and the threshold value has been reached; or
- 3) Monitoring has finished (duration time expired or stop time met); or
- 4) The operation MonitorRouteRequest is received and encounters an active route monitoring entity.

SSF postcondition:

- 1) Monitoring of routes proceeds or is ended depending on the duration time.

11.3.3.3.2 Error handling

Operation-related error handling is not applicable, due to class 4 operation.

11.3.3.4 Responding entity (SCF)

11.3.3.4.1 Normal procedure

SCF preconditions:

- 1) Monitoring of routes is running.
- 2) The SCME is in the state "Waiting for SSF MonitorRouteRequest".

SCF postcondition:

- 1) The SCME forwards the received counter values to the SLPI.

11.3.3.4.2 Error handling

Operation-related error handling is not applicable, due to class 4 operation.

11.4 Operation Timers

Operation timers are defined in Table 27.

Table 27/Q.1248.2 – Operation timers and their value range

Operation Name	Timer	Value range
ActivateServiceFiltering	T _{asf}	Medium
ActivityTest	T _{at}	Short
AnalysedInformation	T _{adi}	Short
AnalyseInformation	T _{ai}	Short
ApplyCharging	T _{ac}	Short
ApplyChargingReport	T _{acr}	Short
AssistRequestInstructions	T _{ari}	Short
AuthorizeTermination	T _{atr}	Short
CallFiltering	T _{cf}	Short
CallGap	T _{cg}	Short
CallInformationReport	T _{cirp}	Short
CallInformationRequest	T _{cirq}	Short
Cancel	T _{can}	Short
CancelStatusReportRequest	T _{csr}	Short
CollectedInformation	T _{cdi}	Short
CollectInformation	T _{ci}	Medium
Connect	T _{con}	Short
ConnectToResource	T _{ctr}	Short
Continue	T _{cue}	Short
ContinueWithArgument	T _{cwa}	Short
CreateCallSegmentAssociation	T _{csa}	Short
CreateOrRemoveTriggerData	T _{crt}	Medium
DisconnectForwardConnection	T _{dfc}	Short
DisconnectForwardConnectionWithArgument	T _{dfcwa}	Short
DisconnectLeg	T _{dl}	Short
EntityRelease	T _{er}	Short
EstablishTemporaryConnection	T _{etc}	Medium
EventNotificationCharging	T _{enc}	Short
EventReportBCSM	T _{erb}	Short
EventReportFacility	T _{erf}	Short
FacilitySelectedAndAvailable	T _{fs}	Short
FurnishChargingInformation	T _{fci}	Short

Table 27/Q.1248.2 – Operation timers and their value range

Operation Name	Timer	Value range
HoldCallInNetwork	T _{hcn}	Short
InitialDP	T _{idp}	Short
InitiateCallAttempt	T _{ica}	Short
ManageTriggerData	T _{mtd}	Medium
MergeCallSegments	T _{mc}	Short
MonitorRouteReport	T _{mrp}	Short
MonitorRouteRequest	T _{mrr}	Medium
MoveCallSegments	T _{mcs}	Short
MoveLeg	T _{ml}	Short
OAbandon	T _{ob}	Short
OAnswer	T _{oa}	Short
OCalledPartyBusy	T _{ob}	Short
ODisconnect	T _{od}	Short
OMidCall	T _{omc}	Short
ONoAnswer	T _{ona}	Short
OriginationAttempt	T _{ora}	Short
OriginationAttemptAuthorized	T _{oaa}	Short
OSuspended	T _{os}	Short
Reconnect	T _{re}	Short
ReleaseCall	T _{rc}	Short
ReportUTSI	T _{ru}	Short
RequestCurrentStatusReport	T _{rcs}	Short
RequestEveryStatusChangeReport	T _{res}	Short
RequestFirstStatusMatchReport	T _{rfs}	Short
RequestNotificationChargingEvent	T _{mc}	Short
RequestReportBCSMEvent	T _{rrb}	Short
RequestReportFacilityEvent	T _{rrfe}	Short
RequestReportUTSI	T _{rru}	Short
ResetTimer	T _{rt}	Short
RouteSelectFailure	T _{rsf}	Short
SelectFacility	T _{sf}	Short
SelectRoute	T _{sr}	Short
SendChargingInformation	T _{sci}	Short

Table 27/Q.1248.2 – Operation timers and their value range

Operation Name	Timer	Value range
SendFacilityInformation	T _{sfi}	Short
SendSTUI	T _{ss}	Short
ServiceFilteringResponse	T _{sfr}	Short
SetServiceProfile	T _{sep}	Short
SplitLeg	T _{sl}	Short
StatusReport	T _{srp}	Short
TAnswer	T _{ta}	Short
TBusy	T _{tb}	Short
TDisconnect	T _{td}	Short
TermAttemptAuthorized	T _{taa}	Short
TerminationAttempt	T _{tara}	Short
TMidCall	T _{tmc}	Short
TNoAnswer	T _{tna}	Short
TSuspended	T _{ts}	Short

12 Parameter Descriptions

Clause 12/Q.1238.2 is applicable with the enhancements specified in the present Recommendation.

12.1 Modified parameters

The following parameters defined in ITU-T Q.1238.2 are modified by this Recommendation.

12.1.1 scfID

scfID is the means of identifying an SCF. The scfID is used in the context of a hand-off/assist procedure and only if the scfID is not embedded in the "destinationRoutingAddress". It may also be used alone (e.g. for mobile applications) to perform a "hand-off" from an initiating SSF (e.g. in a visiting mobile domain) to a requesting SSF (e.g. in home mobile domain). The scfID is used to provide the INAP address of the SCF to establish a connection between the requesting SSF and the specified SCF. The scfID is used to convey the necessary SCF address information (e.g. Global Title) in the network to the requesting SSF. See Q.713 "calling party address" parameter. The network operator has to decide about the actual mapping of this parameter on the used signalling system. This parameter may also indicate the address of the SCF which initiated the call gapping. When scfID is used in an operation, which may cross an internetwork boundary, its encoding must be understood in both networks; this requires bilateral agreement on the encoding.

12.1.2 BearerCapability

This parameter indicates the type of the bearer capability connection, or the transmission medium requirements to the user. It is a network option to select one of the two parameters to be used:

- bearerCap: This parameter contains the value of the DSS1 Bearer Capability parameter (ITU-T Q.931) in case the SSF is at local exchange level, or the value of the ISUP User Service Information parameter (ITU-T Q.763) in case the SSF is at transit exchange level.

The parameter "bearerCapability" shall only be included in the "InitialDP" operation in case the DSS1 Bearer Capability parameter or the ISUP User Service Information parameter is available at the SSP.

If two values for bearer capability are available at the SSF (or if User Service Information and User Service Information Prime are available at the SSF) the "bearerCap" shall contain the value of the preferred bearer capability (respectively the value of the User Service Information Prime parameter).

- tmr: The tmr is encoded as the Transmission Medium Requirement parameter of the ISUP according to ITU-T Q.763.

If two values for transmission medium requirement are available at the (SSF or if Transmission Medium Requirement and Transmission Medium Requirement Prime are available at the SSF) the "bearerCap" shall contain the value of the preferred transmission medium requirement (respectively the value of the Transmission Medium Requirement Prime parameter).

- BroadbandBearerCapability: This alternative sub-parameter indicates a requested broadband connection-oriented bearer service (see ITU-T F.811) to be provided by the network. It explicitly identifies the ATM transfer capability. It contains only information that may be used by the network. Refer to ITU-T Q.2931 for the definition of this parameter.

12.1.3 DpSpecificCommonParameters

The following sub-parameter defined in clause 13 is added to the DpSpecificCommonParameters:

- ipRelatedInformation.

12.1.4 EventSpecificInformationBCSM

This parameter indicates the call related information specific to the event. It comprises the following alternatives:

- collectedInfoSpecificInfo: This alternative specifies the collected information, it contains a number including all collected digits.

It comprises the following fields:

- calledPartyNumber: This field contains all available collected digits in the called number, it may be an incomplete number.

or

- analysedInfoSpecificInfo: This alternative contains the analysed information. It comprises the following fields:

- calledPartyNumber: This field contains the analysed called number, it may be an incomplete number.

or

- routeSelectFailureSpecificInfo

- failureCause: For a definition of this field, refer to the "FailureCause" subclause.

or

– oCalledPartyBusySpecificInfo

- busyCause: For a definition of this field, refer to the "BusyCause" subclause

or

– oNoAnswerSpecificInfo

- oNoAnswerCause: For a definition of this field, refer to the "NoAnswerCause" subclause

or

– oAnswerSpecificInfo

- backwardGVNS: For a definition of this field, refer to the "BackwardGVNS" subclause

or

– oMidCallSpecificInfo

- connectTime: For a definition of this field, refer to the "ConnectTime" subclause.
- oMidCallInfo: This field contains the digits representing IN service control code of the midcall event detected.

or

– oDisconnectSpecificInfo

- releaseCause: For a definition of this field, refer to the "ReleaseCause" subclause
- connectTime: For a definition of this field, refer to the "ConnectTime" subclause.

or

– tBusySpecificInfo

- busyCause: For a definition of this field, refer to the "BusyCause" subclause.

or

– tNoAnswerSpecificInfo

- tNoAnswerCause: For a definition of this field, refer to the "NoAnswerCause" subclause

or

– tAnswerSpecificInfo

- no specific info defined

or

– tMidCallSpecificInfo

- connectTime: For a definition of this field, refer to the "ConnectTime" subclause
- TMidCallInfo: This field contains the digits representing IN service control code of the midcall event detected.

or

– tDisconnectSpecificInfo

- releaseCause: For a definition of this field, refer to the "ReleaseCause" subclause.
- connectTime: For a definition of this field, refer to the "ConnectTime" subclause.

- or
 - oTermSeizedSpecificInfo
 - no specific info defined
- or
 - oSuspend
 - no specific info defined
- or
 - tSuspend
 - no specific info defined
- or
 - origAttemptAuthorized
 - no specific info defined
- or
 - oReAnswer
 - no specific info defined
- or
 - tReAnswer
 - no specific info defined
- or
 - facilitySelectedAndAvailable
 - no specific info defined
- or
 - callAccepted
 - no specific info defined
- or
 - oAbandon
 - abandonCause: For a definition of this field, refer to the "Cause" subclause.
- or
 - tAbandon
 - abandonCause: For a definition of this field, refer to the "Cause" subclause.
- or
 - authorizeRouteFailure
 - authorizeRouteFailureCause: For a definition of this field, refer to the "Cause" subclause.
- or
 - terminationAttemptAuthorized
 - no specific info defined
- or
 - originationAttemptDenied
 - originationDeniedCause: For a definition of this field, refer to the "Cause" subclause.

or

- terminationAttemptDenied
 - terminationDeniedCause: For a definition of this field, refer to the "Cause" subclause.

or

- oModifyRequestSpecificInfo
 - aTMCellRate
 - additionalATMCellRate

or

- oModifyResultSpecificInfo
 - modifyResultType
 - aTMCellRate
 - additionalATMCellRate

or

- tModifyRequestSpecificInfo
 - aTMCellRate
 - additionalATMCellRate

or

- tModifyResultSpecificInfo
 - modifyResultType
 - aTMCellRate
 - additionalATMCellRate

12.1.5 GapCriteria

This parameter identifies the criteria for a call to be subject to service request gapping. It comprises the following alternatives:

- basicGapCriteria

or

- compoundGapCriteria: This parameter comprises the following subparameters:
 - basicGapCriteria;
 - scfID.

This subparameter is restricted to include a fixed GT address string.

NOTE – In the case where the GT addresses more than one SCP (e.g. a mated pair) then if one of these physical SCPs enters overload conditions and issues CallGap, then it is applied to all of them. For a definition of this sub-parameter, refer to 12.1.1 'ScfID'.

12.1.6 Profile

This parameter identifies the (line, group of line, trunk, ...) that is to be managed. It comprises the following alternatives:

- access: This alternative identifies a subscriber access. It provides several addressing schemes to identify the line/subscriber profile linked to a TDP. It contains the number (e.g. call party directory number) needed to identify the subscriber profile.

or

- group: This alternative identifies a facility group. It comprises the following fields:
 - trunkGroupID: This field identifies a trunk group.
 - privateFacility: This field indicates the particular group of private facilities to route a call.
 - huntGroup: This field contains a link to a multiline hunt group. Its content is network operator specific.
 - routeIndex: This field contains a link to a specific trunk routing group. Its content is network operator specific.

12.1.7 ServiceInteractionIndicatorsTwo

This parameter contains a set of indicators which are exchanged between SSP and SCP to resolve interactions between IN-based services and network-based services, respectively between different IN based services. It comprises the following subparameters:

- forwardServiceInteractionInd: It is applicable to operations IDP, CON, ICA, CWA.
This field contains indicators applicable to be sent in forward direction to activate or deactivate suppression of switch-based services at the destination local exchange. Refer to ITU-T Q.1601.
It comprises the following fields:
 - conferenceTreatmentIndicator: This field indicates if a network switch-based conference request is to be accepted or rejected.
The network default is accept conference request.
 - callDiversionTreatmentIndicator: This field indicates if a network switch-based call diversion service request is to be allowed or not allowed, The network default is Call Diversion allowed.
 - callOfferingTreatmentIndicator: This field indicates if call offering is "allowed", "not allowed" or "no impact by IN".
The value "no impact by IN," has only local significance in SSF as a request to SSF not to modify the value of the call offering treatment indicator conveyed in signalling. The network default is Call Offering not allowed.
 - callWaitingTreatmentIndicator: This field indicates if a network switch-based call waiting service request is to be allowed or not allowed. The network default is Call Waiting allowed.
 - holdTreatmentIndicator: It is applicable to operations IDP, CON, CWA. This field indicates if a network switch based call hold service request is to be accepted or rejected. The network default is accept Call Hold request.
 - ectTreatmentIndicator: It is applicable to operations IDP, CON, CWA. This field indicates whether the call leg can become part or not of an ECT call, i.e. if the ECT service request is to be accepted or rejected. The network default is accept ECT request.
- backwardServiceInteractionInd: It is applicable to operations IDP, CON, CTR, ETC, CWA.
This field contains indicators applicable to be sent in backward direction to activate or deactivate suppression of switch-based services at the originating local exchange. Refer to ITU-T Q.1601.

It comprises the following fields:

- conferenceTreatmentIndicator: This field indicates if a network switch-based conference request is to be accepted or rejected.

The network default is accept conference request.

- callCompletionTreatmentIndicator: This field indicates if a network switch-based call completion service request is to be accepted or rejected, The network default is accept call completion service request.
- holdTreatmentIndicator: It is applicable to operations IDP, CON, CWA. This field indicates if a network switch based call hold service request is to be accepted or rejected. The network default is accept Call Hold request.
- ectTreatmentIndicator: It is applicable to operations IDP, CON, CWA. This field indicates whether the call leg can become part or not of an ECT call, i.e. if the ECT service request is to be accepted or rejected. The network default is accept ECT request.

- bothwayThroughConnectionInd: It is applicable to operations CTR, ETC.

This field allows the SCF to control bothway through connection in backward direction for user interaction, i.e. to indicate bothway through connection "required" or "not required". Default is "required". Refer to ITU-T Q.1601.

- suspendTimer: It is applicable to operations CON, ICA CWA.

This field allows the SCF to control the suspend timer to control the release of a called party (non-ISDN subscriber). A reduced time supervision, or an immediate release for the suspend timer as compared to the normal networked timer (T6), is possible to specify.

The suspend timer is to be started by SSF upon receipt of a suspend message generated by the network in response to a clear back indication from an interworking node or an on-hook condition from an analogue called party. Refer to ITU-T Q.1601.

- connectedNumberTreatmentInd: It is applicable to operations CON, CTR, ETC, CWA.

This field allows the SCF to control the number to be presented as "connected" number to calling party.

The following alternatives are possible:

- noINImpact

or

- presentationRestricted

or

- presentCalledINNumber

or

- presentCalledINNumberRestricted

- suppressCallDiversionNotification: It is applicable to CON, ICA, CWA. Refer to ITU-T Q.1601.

- suppressCallTransferNotification: It is applicable to CON, ICA, CWA. Refer to ITU-T Q.1601.

- allowCdINNoPresentationInd: It is applicable to CON, ICA CWA.

It indicates whether the Address Presentation restricted indicator of the ISUP "called IN number" shall be set to presentation allowed (TRUE) or presentation restricted (FALSE). Refer to ITU-T Q.1601.

- userDialogueDurationInd: It is applicable to operations CTR, ETC.
It is applicable when interaction with the user is required during call set-up. The interaction TRUE means the user interaction may last longer than 90 seconds. Otherwise the indicator should be set to FALSE. Used for delaying ISUP T9 timer (answer timer). The default is TRUE. Refer to ITU-T Q.1601.
- overrideLineRestrictions: It is only applicable to operations (e.g. Connect) which lead to a transition to a PIC before the AuthorizeCallSetup PIC. When set to TRUE, this parameter indicates that some facility restrictions should not be checked when the authority to place a call is verified in the Authorize_Call_Setup PIC.
Which restrictions are actually overridden is network specific.
The default is not to override line restrictions (i.e. FALSE).
- suppressVPNAPP: It is applicable to CWA, CON, ICA.
This field indicates whether to allow or stop (suppress) the forward transmission of the VPN PSS1 capability.
When set to TRUE, the exchange, on receipt of this parameter, will not transmit for this call any ISUP Application transport parameter with Application Context Identifier set to "PSS1 ASE (VPN)".
This indicator is populated by the SCF, where the SCF and SSF in conjunction have provided the outgoing gateway PINX functionality as required by PSS1. The default is "allow" (i.e. FALSE).
- calledINNumberOverriding: It is applicable to CON and CWA.
It indicates whether the generation/override of the ISUP "called IN number" is allowed (TRUE) or not allowed (FALSE) If set to FALSE, the ISUP shall not generate a "called IN number" or override an already existing "called IN number".
If absent, the default will be "generation/overriding allowed" (TRUE).
- redirectServiceTreatmentInd: It is applicable to operation CON.
This field indicates if a call redirection service is allowed or not. If absent, the default is "not allowed".
It comprises the following field:
 - redirectReason: The existence of this parameter requests Pivot Routing or Redirection supplementary service to be allowed for the new routing address specified in the Connect operation. If absent, whether Pivot Routing or Redirection service is actually invoked depends only on SSP conditions.
The redirectReason parameter indicates the reason for invoking Pivot Routing/Redirection service. Whether the service is actually invoked depends only on SSP conditions.
- nonCUGCall: It is applicable for Connect and CWA.
This parameter when present, indicates that no parameters for CUG shall be used for the call (i.e. the call shall be a non-CUG call). This parameter when not present, indicates one of two things:
 - continue with modified CUG information (when one or more of either CUG Interlock Code and Outgoing Access Indicator are present); or
 - continue with original CUG information (when neither CUG Interlock Code or Outgoing Access Indicator are present); or
 - continue with the original non-CUG call.

12.2 New parameters

The following new parameters are defined by this Recommendation.

12.2.1 AALParameters

This parameter specifies the ATM adaptation layer parameters information element which provides a means to convey information related to the ATM adaptation layer between endpoints. The information is of significance to both users and local exchanges. The ATM adaptation layer parameters information element is transferred transparently between ATM endpoints by the network. Refer to ITU-T Q.2931.

12.2.2 AdditionalATMCellRate

This parameter indicates information sent in the forward and backward direction to support the use of additional traffic parameters to the application process procedures. Refer to ITU-T Q.2762 for the actual definition of this parameter.

12.2.3 AESACalledParty

This parameter indicates information sent in the forward direction to carry the AESA received in the UNI Called party number IE across public network(s). Refer to ITU-T Q.2762 for the actual definition of this parameter.

12.2.4 AESACallingParty

This parameter indicates information sent in the forward direction to carry the AESA received in the UNI Calling party number IE across public network(s). Refer to ITU-T Q.2762 for the actual definition of this parameter.

12.2.5 AlternativeATMCellRate

This parameter indicates the Alternative ATM Cell Rate or the Alternative ATM traffic descriptor information element. It is used to negotiate alternative traffic parameters during call/connection setup. Refer to ITU-T Q.2763 for the actual definition of Alternative ATM Cell Rate and ITU-T Q.2931 for the actual definition of Alternative ATM traffic descriptor.

12.2.6 ATMCellRate

This parameter indicates information classified by the cell rate identifier indicating the number of cells per second that are required for the call. The cell rate value is unchanged as it traverses the B-ISDN network. Refer to ITU-T Q.2762 for the actual definition of this parameter.

12.2.7 BISDNParameters

This parameter includes the following sub-parameters:

- aAlParameters
- additionalATMCellRate
- aESACalledParty
- aESACallingParty
- alternativeATMTrafficDescriptor
- aTMCellRate
- cDVTDescriptor
- cumulativeTransitDelay
- endToEndTransitDelay
- minAcceptableATMTarfficDescriptor

12.2.8 CDVTDescriptor

This parameter indicates information sent in the forward and backward direction to determine the upper bound of the tolerance admitted for the time interval between cells pertaining to a given cell flow. The calling user may request CDVT values applicable for the forward direction and the maximum acceptable CDVT value for the backward direction. The received CDVT values are handled by the network and conveyed to the called user. The network(s) may use the signalled CDVT values for policing and decisions regarding resource allocation and shaping. The backward CDVT values shall be interpreted as maximum acceptable values for the cell flow in the backward direction. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.9 ConnectionIdentifier

This parameter indicates information sent to identify the ATM virtual connection. It includes the virtual path connection identifier and the virtual channel identifier. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.10 CumulativeTransitDelay

This parameter indicates the cumulative transit delay to be expected for a virtual channel connection. The support of the cumulative transit delay information element is mandatory for the network and optional for the user. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.11 Cug-Index

This parameter is used to select a CUG for an outgoing call at the user, or to indicate an incoming CUG call to the user.

12.2.12 Cug-Interlock

This parameter uniquely identifies a CUG within a network.

12.2.13 Cug-OutgoingAccess

This parameter indicates if the calling user has subscribed to the outgoing access inter-CUG accessibility subscription option.

12.2.14 DetachSignallingPath

This indicator is used in the argument of CPH operations. It tells the CCF whether the signalling path between the Signalling Termination represented by an exported leg and the Signalling Terminations represented by the other legs should be broken or not. When this parameter is absent, the behaviour of the CCF is implementation dependent.

12.2.15 DestinationIndex

This parameter contains a pointer to a call destination (see ITU-T E.410).

12.2.16 EndtoEndTransitDelay

This parameter indicates the End-to-end transit delay information element which indicates the maximum end-to-end transit delay acceptable for a connection. The support of the End-to-end transit delay information element is mandatory for the network and optional for the user. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.17 ExportSignallingPath

This indicator is used by the MoveLeg procedure. It tells the CCF whether the signalling path between the Signalling Termination represented by the exported leg and the Signalling Terminations represented by the other legs of the target CS should be impacted or not. If the parameter is absent, the behaviour of the CCF is implementation dependent. The detailed impact is

outside the scope of this Recommendation and should be specified in interworking Recommendations.

12.2.18 GenericIdentifier

This parameter indicates the DSS2 generic identifier transport capability. This capability allows the transport through the B-ISDN of identifiers required and used by various distributed applications. This signalling capability is a means of end-to-end signalling using identifiers having a specific purpose which the network can optionally access but not change. In some cases, the network may create an identifier on behalf of the user, or at its request. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.19 IncomingSignallingBufferCopy

This indicator is used in the InitiateCallAttempt procedure. When present, the parameters of the setup.ind primitive sent by the CCF should be populated with the information received in the operation argument and in the setup.req primitive from the Signalling Termination associated with the joined controlling leg (if any) of the call segment association. If this indicator is absent, the parameters of the setup.ind primitive sent by the CCF will be populated with the information received in the operation argument and some locally defined information as defined in ITU-T Q.1236.

12.2.20 IPRelatedInformation

This parameter contains a number of sub-parameters that are specific to the interworking with IP-based networks. Currently available sub-parameters are:

- alternativeCalledPartyIds: one or more identities representing a destination in the form of a valid URL. The mapping on call signalling parameters is protocol dependent.
NOTE 1 – In SIP environments, such identities are represented as SIP URLs, mapped to the "To:" field. In H.323 environments, such identities are represented as alias addresses, mapped to the destinationAddress field.
- alternativeOriginatingPartyIds: one or more identities representing an originating party in the form of a valid. The mapping on call signalling parameters is protocol dependent.
NOTE 2 – In SIP environments, such identities are represented as SIP URLs mapped to the "From:" field. In H.323 environments, such identities are represented as alias addresses, mapped to the sourceAddress field.
- alternativeOriginalCalledPartyIds: one or more identities representing the original destination of a forwarded call, in the form of a valid URL. The mapping on call signalling parameters is protocol dependent.
NOTE 3 – In SIP environments, such identities may be represented as SIP URLs, mapped to the "Record-route:" field. In H.323 environments, such identities are represented as alias addresses, mapped to the H.450.3 parameters.
- alternativeRedirectingPartyIds: one or more identities representing a redirecting party, in the form of a valid URL. The mapping on call signalling parameters is protocol dependent.
NOTE 4 – In SIP environments, such identities may be represented as SIP URLs mapped to the "Record-route:" field. In H.323 environments, such identities are represented as alias addresses, mapped to the H.450.3 parameters.

12.2.21 MergeSignallingPaths

This indicator is used by the MergeCallSegment procedure. It tells the CCF whether the signalling path between the Signalling Termination represented by the imported legs and the Signalling Terminations represented by the other legs of the target CS should be impacted or not. If the parameter is absent, the behaviour of the CCF is implementation dependent. The detailed impact is

outside the scope of this Recommendation and should be specified in interworking Recommendations.

12.2.22 MinimumATMCellRate

This parameter indicates the Minimum acceptable ATM Cell Rate or Minimum acceptable ATM traffic descriptor. It is used to negotiate traffic parameters during call/connection setup. The minimum acceptable ATM Cell Rate parameters are the lowest values that the user is willing to accept for the call/connection. Refer to ITU-T Q.2763 for the actual definition of Minimum ATM Cell Rate and to ITU-T Q.2931 for the actual definition of Minimum ATM Traffic Descriptor.

12.2.23 MonitoringCriteria

This parameter is used for route monitoring purposes and specifies when the monitoring results are to be sent to the SCP. It contains one of the following two sub-parameters:

- threshold: This alternative specifies that the report should be sent whenever the threshold is exceeded.
- interval: This alternative parameter indicates that the monitoring report should be sent at the expiration of the interval timer.

12.2.24 MonitoringTimeout

This parameter is used for route monitoring purposes and indicates the duration, during which monitoring should be active. It contains one of the following two sub-parameters:

- duration: Monitoring should stop when the indicated duration timer expires.
A duration of '0' indicates that monitoring should be removed
A duration of '-1' indicates an infinite duration.
- stopTime: Monitoring should stop at the indicated time.

12.2.25 NoAnswerCause

For a definition of this parameter, refer to the "Cause" subclause. It contains a cause value that specifies the reason why a call was not answered.

12.2.26 QoSParameter

This parameter enables the optional signalling of individual Quality of Service (QoS) parameters while establishing a new ATM connection (VP or VC). The calling user can optionally signal individual QoS parameters in addition to the mandatory signalling of the QoS. Refer to ITU-T Q.2931 for the actual definition of this parameter.

12.2.27 RouteCounters

The parameter contains a set of event counters (e.g. failure) each of which is associated with a route. Each set contains the following information:

- Route
- CounterID
- counterValue

12.2.28 StartTime

This parameter indicates when to execute the procedure requested by an operation.

13 Errors

Clause 13/Q.1238.2 is applicable with the enhancements specified in the present Recommendation.

Table 36/Q.1238.2 is replaced with Table 28:

Table 28/Q.1248.2 – Abbreviations for Operations on SSF-SCF interface

Operation abbreviation	Operations SCF → SSF Operation Name	Operation abbreviation	Operations SSF → SCF: Operation Name
ASF	ActivateServiceFiltering	AdI	AnalysedInformation
AI	AnalyseInformation	ACR	ApplyChargingReport
AC	ApplyCharging	ARI	AssistRequestInstructions
AuT	AuthorizeTermination	CdI	CollectedInformation
CF	CallFiltering	FS&A	FacilitySelectedAndAvailable
CIR	CallInformationRequest	IDP	InitialDP
Ca	Cancel	OAb	OAbandon
CaSRR	CancelStatusReportRequest	OAn	OAnswer
CI	CollectInformation	OCPB	OCalledPartyBusy
Co	Connect	OD	ODisconnect
CTR	ConnectToResource	OMC	OMidCall
CWA	ContinueWithArgument	ONoAn	ONoAnswer
CCSA	CreateCallSegmentAssociation	OAtt	OriginationAttempt
CRTD	CreateOrRemoveTriggerData	OAttAu	OriginationAttemptAuthorized
DFC	DisconnectForwardConnection	OS	OSuspended
DFCWA	DisconnectForwardConnectionWith	RSF	RouteSelectFailure
DL	DisconnectLeg	TAn	TAnswer
ETC	EstablishTemporaryConnection	TB	TBusy
CI	FurnishChargingInformation	TD	TDisconnect
HCIN	HoldCallInNetwork	TAttAu	TermAttemptAuthorized
ICA	InitiateCallAttempt	TAtt	TerminationAttempt
MTD	ManageTriggerData	TMC	TMidCall
MeCS	MergeCallSegments	TNoAn	TNoAnswer
MRR	MonitorRouteRequest	TS	TSuspended
MoCS	MoveCallSegments		
ML	MoveLeg		
Re	Reconnect		
RCSR	RequestCurrentStatusReport		
RESCR	RequestEveryStatusChangeReport		
RFSMR	RequestFirstStatusMatchReport		
RNCE	RequestNotificationChargingEvent		
RRBE	RequestReportBCSMEvent		
RRFE	RequestReportFacilityEvent		
RRUTSI	RequestReportUTSI		
RT	ResetTimer		
SF	SelectFacility		
SR	SelectRoute		
SCI	SendChargingInformation		
SFI	SendFacilityInformation		
SSTUI	SendSTUI		
SSP	SetServiceProfile		
SL	SplitLeg		

Table 37/Q.1238.2 is replaced with Table 29:

Table 29/Q.1248.2 – Available Errors for each SSF – SCF Operation

Errors	SSF→SCF																							
	A d I	A C R	A R I	C d I	F S & A	I D P	O A b	O A n	O C P B	O D	O M C	O N o A n	O A t t	O A t t A u	O S	R S F	T A n	T B	T D	T A t t A u	T A t t	T M C	T N o A n	T S
CancelFailed																								
ETCFailed																								
MissingCustomerRecord	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
MissingParameter	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ParameterOutOfRange	X	X				X		X	X	X	X		X		X	X	X	X	X	X	X	X	X	
RequestedInfoError																								
SystemFailure	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
TaskRefused	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
UnexpectedComponent-Sequence	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
UnexpectedDataValue	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
UnexpectedParameter	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
UnknownLegID							X								X									
UnknownResource																								

14 Services assumed from TCAP

Clause 15/Q.1238.2 is applicable with the following changes:

14.1 Handling of SSME-Related messages

14.1.1 SSF to SCF messages

- No dialogue shall be established when the MonitorRouteReport operation is sent. The operation is sent with a TC_BEGIN request primitive and the dialogue is ended by means of a TC_END request primitive with prearranged end.
- The dialogue is locally terminated by means of a TC_END request primitive with prearranged end, upon reception of a TC_BEGIN indication primitive with a CallFiltering operation.

14.1.2 SCF to SSF messages

- A dialogue shall be established when an MonitorRouteRequest operation is sent. The operation shall be transmitted with a TC_BEGIN request primitive.
- A dialogue shall not be established when a CallFiltering operation is sent. The operation is sent using a TC_BEGIN request primitive and the dialogue is terminated with a prearranged end.
- The dialogue is locally terminated upon reception of a MonitorRouteResponse operation using a TC_END request primitive with prearranged end.

14.1.3 Use of TC_BEGIN primitive parameters

The mapping of the parameters onto the TC_BEGIN primitive is defined in ITU-T Q.1248.1 "Mapping to TC dialogue primitives" with the following qualifications:

- The Application Context Name parameter shall take the value of the application-context-name field of the **ssf-scfGenericAC**, **ssf-scfDPSpecificAC**, **ssf-scfAssistHandoffAC** or **ssf-scfServiceManagementAC**, **ssf-scfRouteMonitoringAC** object if the initiating AE is a SSF or the **scf-ssfGenericAC**, **scf-ssfDPSpecificAC**, **scf-ssfINTrafficManagementAC**, **scf-ssfTrafficManagementAC**, **scf-ssfServiceManagementAC** or **scf-ssfStatusReportingAC**, **scf-ssfRouteMonitoringAC** object if the originating AE is a SCF.

Annex A

ASN.1 Definitions

This annex contains the ASN.1 description of the protocol defined by this Recommendation. The content of this annex is available in electronic form only.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems