INTERNATIONAL STANDARD

ISO/IEC 13818-10

First edition
1999-07-15

Information technology — Generic coding of moving pictures and associated audio information —

Part 10:
Conformance extensions for Digital Storage Media Command and Control (DSM-CC)

Technologies de l'information — Codage générique des images animées et des informations sonores associées —

Partie 10: Extensions de conformité pour commande et contrôle de supports de mémoire numérique (DSM-CC)

Reference number
ISO/IEC 13818-10:1999(E)

Contents

1 Scope 1
2 Normative References 1
3 Abbreviations 1
4 The PICS 2
4.1 ISO/IEC Protocol Versions Implemented 3
4.2 Global Statement of Conformance 3
4.3 DSM-CC User-to-Network Functional Unit 4
4.3.1 Roles of DSM-CC User-to-Network 4
4.3.2 Major Capabilities of DSM-CC User-to-Network 4
4.3.3 PDU Support for DSM-CC User-to-Network 4
4.3.4 Parameter support for DSM-CC User-to-Network Message Header 7
4.3.5 Parameter support for DSM-CC Compatibility Descriptors 8
4.3.6 Parameter support for DSM-CC User-to-Network Configuration messages 8
4.3.7 Parameter support for DSM-CC U-N Session messages 9
4.3.8 Parameter Support for DSM-CC User-to-Network Download Messages 19
4.3.9 Parameter support for DSM-CC User-to-Network SDB CCP Messages 21
4.3.10 Parameter Support for DSM-CC U-N Pass-Thru Messages 23
4.4 DSM-CC User-to-User Functional Unit 24
4.4.1 Roles of DSM-CC User-to-User 24
4.4.2 Major capabilities of DSM-CC User-to-User 24
4.4.3 Subsidiary Capabilities Related to DSM-CC User-to-User Procedures 27
4.5 Conformance 29
5 The Conformance ATS 29
5.1 Test Method 29
5.1.1 Test Environment of DSM-CC U-N Functions 30
5.1.2 Test Environment of DSM-CC U-U Functions 31
5.2 Test Cases 31
5.2.1 DSM-CC User-to-Network 31
5.2.2 Test Coverage 96
5.2.3 DSM-CC User-to-User 101

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 13818-10 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology,* Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information.*

ISO/IEC 13818 consists of the following parts, under the general title *Information technology — Generic coding of moving pictures and associated audio information*:

— *Part 1: Systems*
— *Part 2: Video*
— *Part 3: Audio*
— *Part 4: Conformance testing*
— *Part 5: Software simulation*
— *Part 6: Extensions for DSM-CC*
— *Part 7: Advanced Audio Coding (AAC)*
— *Part 9: Extension for real time interface for systems decoders*
— *Part 10: Conformance extensions for Digital Storage Media Command and Control (DSM-CC)*

This page intentionally left blank

Information technology – Generic coding of moving pictures and associated audio information – Part 10: Conformance extensions for Digital Storage Media Command and Control (DSM-CC)

1 Scope

This part of ISO/IEC 13818 defines compliance to Data Storage Media Command and Control (DSMCC) standard in 2 steps: the static review and the dynamic review as defined in ISO/IEC 9646 Conformance Testing standard [1, 2, 3]. The static review requirements are specified in clause 4 of this part of ISO/IEC 13818 in the form of Protocol Implementation Conformance Statement (PICS) proforma. The ATS used for dynamic review is described in clause 5.

This part of ISO/IEC 13818 does not specify all the requirements with which terminal equipment intended for use in conjunction with multimedia information retrieval services has to comply. In particular, this part of ISO/IEC 13818 does not specify (lower layer) protocols to be used to deliver/transport DSM-CC protocol data units. Neither does it specify requirements related to safety, protection, and electromagnetic compatibility (EMC) of the equipment, or regulatory requirements with which such equipment may be required to comply.

2 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 13818. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 13818 are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

[1] ISO/IEC 9646-1: Information technology - Open Systems Interconnection - Conformance testing methodology and framework – Part 1: General concepts.

[2] ISO/IEC 9646-2: Information technology - Open Systems Interconnection - Conformance testing methodology and framework – Part 2: Abstract Test Suite Specification.

[3] ISO/IEC 9646-7: Information technology - Open Systems Interconnection - Conformance testing methodology and framework – Part 7: Implementation Conformance Statements.

[4] ISO/IEC 13818-6: Information technology - Generic coding of moving pictures and associated audio information – Part 6: Extensions for DSM-CC.

3 Abbreviations

API : Application Portability Interface

ATS : Abstract Test Suite

DSM-CC : Digital Storage Media-Command and Control

DSM-CC UN : Digital Storage Media-Command and Control User to Network

DSM-CC UU : Digital Storage Media-Command and Control User to User

ETS : Executable Test Suite

LT : Lower Tester

IUT : Implementation Under Test

MPEG-2 : Moving Picture Experts Group-2

PCO : Point of Control and Observation

PDU : Protocol Data Unit

SUT : System Under Test

TCP : Test Coordination Procedures

UT : Upper Tester

VOD : Video on Demand

Additional abbreviations are defined in ISO/IEC 13818-6 and ISO/IEC 9646 parts 1-7.

4 The PICS

This part of ISO/IEC 13818 defines a Protocol Implementation Conformance Statement (PICS) proforma for the detailed expression of the conformance requirements of ISO/IEC 13818-6. The PICS proforma is in compliance with the relevant requirements, and in accordance with the relevant guidance for a PICS proforma, given in ISO 9646-2. Detail of the use of this proforma is provided in this part of ISO/IEC 13818. Implementations claiming conformance to ISO/IEC 13818-6 shall complete the proforma as part of the conformance requirements.

The DSM-CC specification is a collection of protocols and interfaces that accomplish the control of bit streams. DSM-CC can be viewed as consisting of two broad categories of functionality; User-to-User and User-to-Network. Each of these categories is further divided into functional groups.

The following table defines the functional groups of DSM-CC User-to-Network and provides a brief description of each group:

Table 1 – DSM-CC User-to-Network Functional Groups

Functional Group	Description
U-N Configuration	A set of messages which are used to configure a User device for operation in a DSM-CC Network environment.
U-N Session Messages	A set of messages which are used to set-up maintain, and tear-down DSM-CC sessions.
U-N Download	A set of messages which are used to download information to a DSM-CC User.
U-N Switched Digital Broadcast Channel Change Protocol	A set of messages which are used by the User to request stream switching by the Network.
U-N Pass Thru Messages	A set of messages which are used to communicate between Users through the Network and outside of a session.
DSM-CC Message Header	All DSM-CC UN messages begin with the DSM-CC Message Header, which contains information about the type of message being passed

The following table defines the functional groups of DSM-CC User-to-User and provides a brief description of each group:

Table 2 – DSM-CC User-to-User Functional Groups

Functional Group	Description
U-U Application Portability Interfaces (API)	Describes the U-U interfaces for application software programmer
U-U Service Interoperability Interfaces (SII)	Describes the U-U interfaces for network software programmer
U-U Application Boot Process	Describes a complete boot process including User-to-Network Session messages
U-U Object Carousel	U-U API for broadcast networks
Stream Descriptors	Descriptors that are inserted into the MPEG-2 transport stream to control or signal a change in the stream.

Transport in MPEG-2 Systems	Defines the method by which the messages and data shall be transported when these messages or data are transported in an MPEG-2 Systems Transport. This clause does not specify transport methods other than MPEG-2 Systems Transport.

An implementation may be compliant with any or all of the functional groups as required by that implementation.

4.1 ISO/IEC Protocol Versions Implemented

ISO/IEC 13818-6 identifies the protocol version through the use of the dsmccType field in the message header of User to Network messages. If the protocol is modified, the dsmccType for that class of messages shall be assigned a new value. The following table (from ISO/IEC 13818-6 clause 2, table 2-2) identifies the current versions of the DSM-CC protocol:

Table 3 – DSM-CC Protocol Versions

dsmccType	Description
0x00	ISO/IEC 13818-6 Reserved
0x01	Identifies the message as an ISO/IEC 13818-6 User-to-Network configuration message.
0x02	Identifies the message as an ISO/IEC 13818-6 User-to-Network session message.
0x03	Identifies the message as an ISO/IEC 13818-6 Download message.
0x04	Identifies the message as an ISO/IEC 13818-6 SDB Channel Change Protocol message.
0x05	Identifies the message as an ISO/IEC 13818-6 User-to- Network pass-thru message.
0x06-0x7F	ISO/IEC 13818-6 Reserved.
0x80-0xFF	User Defined message type.

For DSM-CC User-to-User, the protocol version is identified by the object version in the message header.

An implementation shall be described by completing a separate PICS proforma for each supported protocol version. The PICS for all versions of the protocol for which conformance is claimed, should be attached to each other, and used together.

4.2 Global Statement of Conformance

Table 4 – DSM-CC Functional Units

Item Number	Does the implementation support ...	Condition for status	Status	ISO/IEC 13818-6 reference	Implemented? Y = Yes, N = No, n/a = not applicable
Fu1	**User-to-Network (U-N)**		o.1	4	
Fu2	**User-to-User (U-U)**		o.1	5	
Fu3	**Stream Descriptors**	Fu2	o.1	8	
Fu4	**Transport in MPEG-2 Systems**	Fu1 or Fu2	o.1	9	
o.1 It is mandatory to support at least one of these options Note: DSM-CC may be used in an interactive network and/or a broadcast network. Therefore some functional units are exclusive. In a broadcast environment the U-U Object Carousel will be implemented while in an interactive network the U-U Service Interoperability Interface will be implemented.					

4.3 DSM-CC User-to-Network Functional Unit

4.3.1 Roles of DSM-CC User-to-Network

Table 5 – User-to-Network Role PICS

Item	Role Does the implementation support...	Conditions for status	Status	ISO/IEC 13818-6 reference
R1	the client role		o.1	0.6
R2	the session & resource manager (SRM) role		o.1	0.6
R3	the server role		o.1	0.6
o.1	Support of one, and only one, of these options is required.			

4.3.2 Major Capabilities of DSM-CC User-to-Network

Table 6 – DSM-CC User-to-Network Major Capability PICS

Item	Does the implementation support...	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM		Implemented? Y = Yes, N = No, n/a = not applicable		
			Conditions for status	Status	Conditions for status	Status	Conditions for status	Status	Client	Server	SRM
Mc1	**User-to-Network (U-N)**	3.1	Mc2 Not Mc2	O M	Mc2 Not Mc2	o		m			
Mc11	Configuration Procedures	3		c:o		c:o		c:o			
Mc12	Session Control Procedures	4		c:o1		c:m		c:m			
Mc13	Data Download Procedures	7		c:o		c:o		c:o			
Mc14	Switched Digital Broadcast / Channel Change Procedures	10		c:o1		c:o2		n/a			
Mc15	Pass-Thru Procedures	12		c:o		c:o		c:o			
o.1: It is mandatory to support at least one of these options o.2: It is mandatory to support at least one of these options											

4.3.3 PDU Support for DSM-CC User-to-Network

Table 7 – PICS for DSM-CC User-to-Network PDU Support

Item	Does the implementation support...	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM		Implemented? Y = Yes, N = No, n/a = not applicable		
			Cond. for status	Status	Cond. for status	Status	Cond. for status	Status	Client	Server	SRM
Mc11	**User-to-Network Configuration Group**										
Pdu1	UNConfigRequest	3.3.1		m		m		m			
Pdu2	UNConfigIndication	3.3.3		m		m		m			
Pdu3	UNConfigResponse	3.3.4		m		m		m			
Pdu4	UNConfigConfirm	3.3.2		m		m		m			
Mc12	**User-to-Network Session Control Group**										
Pdu5	ClientSessionSetUpReqest	4.2.4.1		m		n/a		m			
Pdu6	ClientSessionSetUpConfirm	4.2.4.2		m		n/a		m			
Pdu7	ServerSessionSetUpIndication	4.2.4.3		n/a		m		m			
Pdu8	ServerSessionSetUpResponse	4.2.4.4		n/a		m		m			
Pdu9	ServerContinuousFeedSession Request	4.3.8.1		n/a		o		o			
Pdu10	ServerContinuousFeedSession Confirm	4.2.8.2		n/a		o		o			
Pdu11	ClientReleaseRequest	4.2.5.1		m		n/a		m			
Pdu12	ClientReleaseIndication	4.2.5.3		m		n/a		m			
Pdu13	ClientReleaseResponse	4.2.5.4		m		n/a		m			
Pdu14	ClientReleaseConfirm	4.2.5.2		m		n/a		m			
Pdu15	ServerReleaseRequest	4.2.5.5		n/a		m		m			
Pdu16	ServerReleaseIndication	4.2.5.7		n/a		m		m			

Pdu17	ServerReleaseResponse	4.2.5.8		n/a		m		m			
Pdu18	ServerReleaseConfirm	4.2.5.6		n/a		m		m			
Pdu19	ClientAddResourceIndication	4.2.6.1	If Pdu21 used	m		n/a		m			
Pdu20	ClientAddResourceResponse	4.2.6.2	If Pdu21 used	m		n/a		m			
Pdu21	ServerAddResourceRequest	4.2.6.3		n/a		o		m			
Pdu22	ServerAddResourceConfirm	4.2.6.4		n/a	If Pdu21 used	m		m			
Pdu23	ClientDeleteResourceIndication	4.2.7.1	If Pdu25 used	m		n/a		m			
Pdu24	ClientDeleteResourceResponse	4.2.7.2	If Pdu25 used	m		n/a		m			
Pdu25	ServerDeleteResourceRequest	4.2.7.3		n/a		o		m			
Pdu26	ServerDeleteResourceConfirm	4.2.7.4		n/a	If Pdu25 used	m		m			
Pdu27	ClientStatusRequest	4.2.9.1		m		n/a		m			
Pdu28	ClientStatusIndication	4.2.9.3		m		n/a		m			
Pdu29	ClientStatusResponse	4.2.9.4		m		n/a		m			
Pdu30	ClientStatusConfirm	4.2.9.2		m		n/a		m			
Pdu31	ServerStatusRequest	4.2.9.5		n/a		m		m			
Pdu32	ServerStatusIndication	4.2.9.7		n/a		m		m			
Pdu33	ServerStatusResponse	4.2.9.8		n/a		m		m			
Pdu34	ServerStatusConfirm	4.2.9.6		n/a		m		m			
Pdu35	ClientResetRequest	4.2.10.1		m		n/a		m			
Pdu36	ClientResetIndication	4.2.10.3		m		n/a		m			
Pdu37	ClientResetResponse	4.2.10.4		m		n/a		m			
Pdu38	ClientResetConfirm	4.2.10.2		m		n/a		m			
Pdu39	ClientSessionProceedingIndication	4.2.11.1		o		n/a		m			
Pdu40	ServerSessionProceedingIndication	4.2.11.2		n/a		m		m			
Pdu41	ClientSessionInProgress	4.2.14.1		o		n/a		o			
Pdu42	ServerSessionInProgress	4.2.14.2		n/a		o		o			
Pdu43	ServerResetRequest	4.2.10.5		n/a		m		m			
Pdu44	ServerResetIndication	4.2.10.7		n/a		m		m			
Pdu45	ServerResetResponse	4.2.10.8		n/a		m		m			
Pdu46	ServerResetConfirm	4.2.10.6		n/a		m		m			
Pdu47	ClientResetRequest	4.2.10.1		m		m		n/a			
Pdu48	ClientResetConfirm	4.2.10.2		m		m		n/a			
Pdu49	ClientResetIndication	4.2.10.3		m		m		n/a			
Pdu50	ClientResetResponse	4.2.10.4		m		m		n/a			
Pdu51	ClientConnectRequest	4.2.12.1		o		n/a		o			
Pdu52	ServerConnectIndication	4.2.12.2		n/a		o		o			
Pdu53	ServerSessionTransferRequest	4.2.13.3		n/a		o		o			
Pdu54	ServerSessionTransferIndication	4.2.13.5		n/a		o		o			
Pdu55	ServerSessionTransferResponse	4.2.13.6		n/a		o		o			
Pdu56	ServerSessionTransferConfirm	4.2.13.4		n/a		o		o			
Pdu57	ClientSessionTransferIndication	4.2.13.1		o		n/a		o			
Pdu58	ClientSessionTransferResponse	4.2.13.2		o		n/a		o			
Mc13	**User-to-Network Download**										
Pdu59	DownloadInfoRequest	7.3.1									
Pdu60	DownloadInfoResponse	7.3.2									
Pdu61	DownloadInfoIndication	7.3.2									
Pdu62	DownloadDataBlock	7.3.3									
Pdu63	DownloadDataRequest	7.3.4									
Pdu64	DownloadCancel	7.3.5									
Pdu65	DownloadServerInitiate	7.3.6									
Mc14	**User-to-Network Switched Digital Broadcast Channel Change protocol**										
Pdu66	SDBProgramSelectRequest	10.2.3.1									
Pdu67	SDBProgramSelectConfirm	10.2.3.2									

Pdu68	SDBProgramSelectIndication	10.2.3.3									
Pdu69	SDBProgramSelectResponse	10.2.3.4									
Mc15	**User-to-Network Pass-Through messages**										
Pdu70	PassThruRequest	12.2.2.1		o		o		o			
Pdu71	PassThruIndication	12.2.2.2		o		o		o			
Pdu73	PassThruReceiptRequest	12.2.2.3		o		o		o			
Pdu74	PassThruReceiptConfirm	12.2.2.4		o		o		o			
Pdu75	PassThruReceiptIndication	12.2.2.5		o		o		o			
Pdu76	PassThruReceiptResponse	12.2.2.6		o		o		o			

4.3.4 Parameter support for DSM-CC User-to-Network Message Header

Table 8 – DSM-CC User-to-Network Message Header Parameters

Item	Parameter	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM	
			Status	Values allowed	Status	Values allowed	Status	Values allowed
Par1	protocolDiscriminator	2	m	0x11	m	0x11	m	0x11
Par2	dsmccType	2	m o	0x01-0x05 0x80-0xFF	m o	0x01-0x05 0x80-0xFF	m o	0x01-0x05 0x80-0xFF
Par3	messageId	3.3	m o	0x0001-0x0004 0x8000-0xFFFF	m o	0x0001-0x0004 0x8000-0xFFFF	m o	0x0001-0x0004 0x8000-0xFFFF
		4.2	m m m m o	0x4010-0x4011 0x4020-0x4023 0x4032-0x4033 0x4042-0x4043 0x4060-0x4063 0x4070-0x4073 0x4082 0x4090 0x40A2-0x40A3 0x40B0 0x6000-0x7FFF	n/a		m	0x4010-0x4011 0x4020-0x4023 0x4032-0x4033 0x4042-0x4043 0x4060-0x4063 0x4070-0x4073 0x4082 0x4090 0x40A2-0x40A3 0x40B0 0x6000-0x7FFF
			n/a		m o	0x8012-0x8013 0x8020-0x8023 0x8030-0x8031 0x8040-0x8041 0x8050-0x8051 0x8060-0x8063 0x8070-0x8073 0x8082 0x8092 0x80A0-0x80A3 0x80B0 0xA000-0xFFFF	m o	0x8012-0x8013 0x8020-0x8023 0x8030-0x8031 0x8040-0x8041 0x8050-0x8051 0x8060-0x8063 0x8070-0x8073 0x8082 0x8092 0x80A0-0x80A3 0x80B0 0xA000-0xFFFF
		7.3	m	0x1001-0x1006	m	0x1001-0x1006	n/a	
		10.2	m o	0x0001-0x0004 0x8000-0xFFFF	n/a		m o	0x0001-0x0004 0x8000-0xFFFF
		12.2	m o	0x0001-0x0006 0x8000-0xFFFF	m o	0x0001-0x0006 0x8000-0xFFFF	m o	0x0001-0x0006 0x8000-0xFFFF
Par4	transactionId	2	m	0x00 - 0x03	m	0x00 - 0x03	m	0x00 - 0x03
Par5	adaptationLength	2	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par6	messageLength	2	m	0x0000-0xFFFF	m	0x0000-0xFFFF	m	0x0000-0xFFFF
Par7	dsmccAdaptationHeader	2.1	o		o		m	
Par7.1	AdaptationType	2.1	c:m c:o	0x01, 0x02 0x80-0xFF	c:m c:o	0x01, 0x02 0x80-0xFF	c:m c: o	0x01, 0x02 0x80-0xFF
Par7.1.2	Reserved	2.1.1	c:m	0xFF	c:m	0xFF	c:m	0xFF
Par7.1.3	CaSystemId	2.1.1	c:m	(CA_system_ID in ISO/IEC 13818-1)	c:m	(CA_system_ID in ISO/IEC 13818-1)	c:m	(CA_system_ID in ISO/IEC 13818-1)
Par7.1.4	ConditionalAccessLength	2.1.1	c:m	TBA	c:m	TBA	c:m	TBA
Par7.1.5	conditionalAccesDataByte	2.1.1	c:m	TBA	c:m	TBA	c:m	TBA
Par7.2	dsmccUserId	2.1.2	o		o		o	
Par7.2.1	reserved	2.1.2	c:m	0xFF	c:m	0xFF	c:m	0xFF
Par7.2.2	userId	2.1.2	c:m	(values of clientId or serverId in U-N session messages)	c:m	(values of clientId or serverId in U-N session messages)	c:m	(values of clientId or serverId in U-N session messages)

c:m If the dsmccAdaptationHeader is supported, this parameter is mandatory.
The support of the dsmccAdaptationHeader is optional.
c:o If the dsmccAdaptationHeader is supported, this parameter is optional.
The support of the dsmccAdaptationHeader is optional.

4.3.5 Parameter support for DSM-CC Compatibility Descriptors

Table 9 – DSM-CC Compatibility Descriptor Parameters

Item	Parameter	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM	
			Status	values allowed	Status	Values allowed	Status	Values allowed
Par1	compatibiltyDescriptorLength	6.1	m	0x0000-0xFFFF	m	0x0000-0xFFFF	m	0x0000-0xFFFF
Par2	descriptorCount	6.1	m	0x0000-0xFFFF	m	0x0000-0xFFFF	m	0x0000-0xFFFF
Par3	descriptorType	6.1	m o	0x00 - 0x02 0x40 - 0xFF	m o	0x00 - 0x02 0x40 - 0xFF	m o	0x00 - 0x02 0x40 - 0xFF
Par4	descriptorLength	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par5	specifierType	6.1	m o	0x01 0x80 - 0xFF	m o	0x01 0x80 - 0xFF	m o	0x01 0x80 - 0xFF
Par6	specifierData	6.1	m	0x000000 - 0xFFFFFF	m	0x000000 - 0xFFFFFF	m	0x000000 - 0xFFFFFF
Par7	model	6.1	m	0x0000- 0xFFFF	m	0x0000- 0xFFFF	m	0x0000- 0xFFFF
Par8	version	6.1	m	0x0000- 0xFFFF	m	0x0000- 0xFFFF	m	0x0000- 0xFFFF
Par9	subDescriptorCount	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par10	SubDescriptor	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par10.1	subDescriptorType	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par10.2	subDescriptorLength	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par10.3	additionalInformation	6.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF

4.3.6 Parameter support for DSM-CC User-to-Network Configuration messages

Table 10 – DSM-CC User-to-Network Configuration Message Parameters

Item	Parameter	ISO/IEC 13818-6 Reference	Role: Client		Role: Server		Role: SRM	
			Status	Values Allowed	Status	Values Allowed	Status	Values Allowed
User-to-Network Configuration Parameters								
Par1	**dsmccConfigurationParameters**							
Par1.1	MessageTimer	3.2.1	m	0x00000000 - 0xFFFFFFFF	m	0x00000000 - 0xFFFFFFFF	m	0x00000000 - 0xFFFFFFFF
Par1.2	SessionInProgressTimer	3.2.1	m	0x00000000 - 0xFFFFFFFF	m	0x00000000 - 0xFFFFFFFF	m	0x00000000 - 0xFFFFFFFF
Par1.3	MessageRetryCount	3.2.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par1.4	SessionIdAssignor	3.2.1	m	0x00 - 0x01	m	0x00 - 0x01	m	0x00 - 0x01
Par1.5	ResourceIdAssignor	3.2.1	m	0x00 - 0x01	m	0x00 - 0x01	m	0x00 - 0x01
Par1.6	MaximumForwardCount	3.2.1	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par2	**NetworkConfigurationParameters**							
Par2.1	UserId	3.2.2	m	(OSI NSAP address)	m	(OSI NSAP address)	m	(OSI NSAP address)
Par2.2	PrimaryServerId	3.2.2	m	(OSI NSAP address)	m	(OSI NSAP address)	m	(OSI NSAP address)
Par2.3	NetworkParameterLength	3.2.2	m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par2.4	NetworkParameterDataByte	3.2.2	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par3	**UserDefinedConfigurationParameters**							
Par3.1	UserDefinedParameterLength	3.2.2	o	0x0000 - 0xFFFF	o	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par3.2	UserDefinedParameterDataByte	3.2.2	o	0x00 - 0xFF	o	0x00 - 0xFF	m	0x00 - 0xFF
UNConfigRequest								
Par4	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par5	DeviceId	3.3.1	m	0x000000000-0xFFFFFFFFFFFF	m	0x000000000-0xFFFFFFFFFFFF	m	0x000000000-0xFFFFFFFFFFFF
Par6	Reserved	3.3.1	m	0xFFFF	m	0xFFFF	m	0xFFFF

Par7	CompatibiltyDescriptor	6	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)
UNConfigConfirm								
Par8	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)		(as specified by ISO/IEC 13818-6 subclause 4.3.4)		(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par9	DeviceId	3.3.2	m	0x000000000-0xFFFFFFFFFFF		0x000000000-0xFFFFFFFFFFF		0x000000000-0xFFFFFFFFFFF
Par10	Response	3.3.2	m o	0x0000 - 0x0003 0x8000 - 0xFFFF		0x0000 - 0x0003 0x8000 - 0xFFFF		0x0000 - 0x0003 0x8000 - 0xFFFF
Par11	DsmccConfigurationParameters	3.2.1	m	(see Par1)	m	(see Par1)	m	(see Par1)
Par12	NetworkConfigurationParameters	3.2.2	m	(see Par2)	m	(see Par2)	m	(see Par2)
Par13	UserDefinedConfigurationParameters	3.3.2	o	(see Par3)	o	(see Par3)	o	(see Par3)
UNConfigIndication								
Par11	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par12	DeviceId	3.3.3	m	0x000000000-0xFFFFFFFFFFF	m	0x000000000-0xFFFFFFFFFFF	m	0x000000000-0xFFFFFFFFFFF
Par13	Reason	3.3.3	m o	0x0000 - 0x0001 0x8000 - 0xFFFF	m o	0x0000 - 0x0001 0x8000 - 0xFFFF	m o	0x0000 - 0x0001 0x8000 - 0xFFFF
Par14	CompatibiltyDescriptor	3.3.3	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	M	(as specified by ISO/IEC 13818-6 subclause 4.3.5)
Par15	DsmccConfigurationParameters	3.2.1	m	(see Par1)	m	(see Par1)	M	(see Par1)
Par16	NetworkConfigurationParameters	3.2.2	m	(see Par2)	m	(see Par2)	M	(see Par2)
Par17	UserDefinedConfigurationParameters	3.3.2	o	(see Par3)	o	(see Par3)	O	(see Par3)
UNConfigResponse								
Par18	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par19	UserId	3.3.4	m	(see OSI NSAP)	m	(see OSI NSAP)	m	(see OSI NSAP)
Par20	Response	3.3.4	m o	0x0000 - 0x0003 0x8000 - 0xFFFF	m o	0x0000 - 0x0003 0x8000 - 0xFFFF	m o	0x0000 - 0x0003 0x8000 - 0xFFFF
Par21	Reserved	3.3.4	m	0x00 - 0xFF	m	0x00 - 0xFF	m	0x00 - 0xFF
Par22	CompatibiltyDescriptor	6	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.5)

4.3.7 Parameter support for DSM-CC U-N Session messages

Table 11 – DSM-CC User-to-Network Session Message Parameters

Item	Parameter	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM	
			Status	Values allowed	Status	Values allowed	Status	Values allowed
ClientSessionSetUpRequest								
Par1	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par2	sessionId	4.2.4.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par3	Reserved	4.2.4.1	m	0xFFFF	n/a		m	0xFFFF
Par4	clientId	4.2.4.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par5	serverId	4.2.4.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)

Par6	UserData	4.2.4.1	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientSessionSetUpConfirm								
Par7	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par8	sessionId	4.2.4.2	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par9	serverId	4.2.4.2	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par10	Response	4.2.4.2	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par11	Resources	4.2.4.2	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par12	UserData	4.2.4.2	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6subclause 4.2.2)
ClientReleaseRequest								
Par13	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par14	sessionId	4.2.5.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par15	Reason	4.2.5.1	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m o	0x00000x001B 0x8000 - 0xFFFF reason
Par16	UserData	4.2.5.1	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientReleaseIndication								
Par17	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par18	sessionId	4.2.5.3	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par19	reason	4.2.5.3	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m	0x00000x001B 0x8000 - 0xFFFF
Par20	UserData	4.2.5.3	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientReleaseIndication								
Par21	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par22	sessionId	4.2.5.4	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par23	Response	4.2.5.4	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par24	UserData	4.2.5.4	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientReleaseConfirm								
Par25	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par26	sessionId	4.2.5.2	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)

Par27	response	4.2.5.2	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par28	UserData	4.2.5.2	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientAddResourceIndication								
Par29	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par30	sessionId	4.2.6.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par31	Resources	4.2.6.1	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par32	UserData()	4.2.6.1	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientAddResourceResponse								
Par33	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par34	sessionId	4.2.6.2	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par35	response	4.2.6.2	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par36	Resources	4.2.6.2	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par37	UserData()	4.2.6.2	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientDeleteResourceIndication								
Par38	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par39	sessionId	4.2.7.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par40	reason	4.2.7.1	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m o	0x00000x001B 0x8000 - 0xFFFF
Par41	Resources	4.2.7.1	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Pa42	UserData()	4.2.7.1	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ClientDeleteResourceResponse								
Par43	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par44	sessionId	4.2.7.2	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par45	response	4.2.7.2	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par46	UserData()	4.2.7.2	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)

ClientStatusRequest								
Par43	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par44	reason	4.2.9.1	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m o	0x00000x001B 0x8000 - 0xFFFF
Par45	statusType	4.2.9.1	m o	0x0001 - 0x0005 0x8000-0xFFFF	n/a		m o	0x0001 - 0x0005 0x8000-0xFFFF
Par46	statusCount	4.2.9.1	m	0x0000 - 0xFFFF	n/a		m	0x0000 - 0xFFFF
Par47	statusByte	4.2.9.1	m	(as specified by ISO/IEC 13818-6 subclause 4.6)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.6)
ClientStatusIndication								
Par48	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par49	reason	4.2.9.3	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m	0x00000x001B 0x8000 - 0xFFFF
Par50	statusType	4.2.9.3	m o	0x0001 - 0x0005 0x8000-0xFFFF	n/a		m o	0x0001 - 0x0005 0x8000-0xFFFF
Par51	statusCount	4.2.9.3	m	0x0000 - 0xFFFF	n/a		m	0x0000 - 0xFFFF
Par52	statusByte	4.2.9.3	m	(as specified by ISO/IEC 13818-6 subclause 4.6)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.6)
ClientStatusResponse								
Par53	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par54	response	4.2.9.4	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par55	statusType	4.2.9.4	m o	0x0001 - 0x0005 0x8000-0xFFFF	n/a		m o	0x0001 - 0x0005 0x8000-0xFFFF
Par56	statusCount	4.2.9.4	m	0x0000 - 0xFFFF	n/a		m	0x0000 - 0xFFFF
Par57	statusByte	4.2.9.4	m	(as specified by ISO/IEC 13818-6 subclause 4.6)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.6)
ClientStatusConfirm								
Par58	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par59	response	4.2.9.2	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par60	statusType	4.2.9.2	m o	0x0001 - 0x0005 0x8000-0xFFFF	n/a		m o	0x0001 - 0x0005 0x8000-0xFFFF
Par61	statusCount	4.2.9.2	m	0x0000 - 0xFFFF	n/a		m	0x0000 - 0xFFFF
Par62	statusByte	4.2.9.2	m	(as specified by ISO/IEC 13818-6 subclause 4.6)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.6)
ClientSessionProceedingIndication								
Par63	dsmccMessageHeader()	2	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par64	sessionId	4.2.11.1	m	(as specified by ISO/IEC 13818-6 subclause 4.3)	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par65	reason	4.2.11.1	m o	0x00000x001B 0x8000 - 0xFFFF	n/a		m o	0x00000x001B 0x8000 - 0xFFFF
ServerSessionSetUpIndication								
Par66	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par67	sessionId	4.2.4.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)

Par68	clientId	4.2.4.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par69	serverId	4.2.4.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par70	forwardCount	4.2.4.3	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par71	forwardServerId	4.2.4.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par72	UserData	4.2.4.3	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerSessionSetUpResponse								
Par73	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par74	sessionId	4.2.4.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par75	serverId	4.2.4.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par76	response	4.2.4.4	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par77	nextServerId	4.2.4.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par78	Resources	4.2.4.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par79	UserData	4.2.4.4	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerReleaseRequest								
Par80	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par81	sessionId	4.2.5.5	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par82	reason	4.2.5.5	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
Par83	UserData	4.2.5.5	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerReleaseIndication								
Par84	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par85	sessionId	4.2.5.7	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par86	reason	4.2.5.7	n/a		m	0x00000x001B 0x8000 - 0xFFFF	m	0x00000x001B 0x8000 - 0xFFFF
Par87	UserData	4.2.5.7	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerReleaseResponse								
Par88	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par89	sessionId	4.2.5.8	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3) INTEGER	m	(as specified by ISO/IEC 13818-6 subclause 4.3) INTEGER

Par90	response	4.2.5.8	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par91	UserData	4.2.5.8	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerReleaseConfirm								
Par92	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par93	sessionId	4.2.5.6	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par94	response	4.2.5.6	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par95	UserData	4.2.5.6	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerAddResourceRequest								
Par96	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par97	sessionId	4.2.6.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par98	Resources	4.2.6.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par99	UserData()	4.2.6.3	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerAddResourceConfirm								
Par100	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par101	sessionId	4.2.6.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par102	response	4.2.6.4	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par103	Resources	4.2.6.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par104	UserData()	4.2.6.4	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerDeleteResourceRequest								
Par105	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par106	sessionId	4.2.7.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par107	reason	4.2.7.3	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
Par108	Resources	4.2.7.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
Par109	UserData()	4.2.7.3	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)

ServerDeleteResourceConfirm								
Par110	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par111	sessionId	4.2.7.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par112	response	4.2.7.4	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par113	UserData()	4.2.7.4	n/a		o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	o	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerStatusRequest								
Par114	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par115	serverId	4.2.9.5	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par116	reason	4.2.9.5	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
Par117	statusType	4.2.9.5	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par118	statusCount	4.2.9.5	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par119	statusByte	4.2.9.5	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerStatusIndication								
Par120	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par121	reason	4.2.9.7	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
Par122	statusType	4.2.9.7	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par123	statusCount	4.2.9.7	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par124	statusByte	4.2.9.7	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerStatusResponse								
Par125	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par126	response	4.2.9.8	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par127	statusType	4.2.9.8	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par128	statusCount	4.2.9.8	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par129	statusByte	4.2.9.8	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)
ServerStatusConfirm								
Par130	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par131	response	4.2.9.6	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par132	statusType	4.2.9.6	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par133	statusCount	4.2.9.6	n/a		m	0x0000 - 0xFFFF	m	0x0000 - 0xFFFF
Par134	statusByte	4.2.9.6	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.2)

ClientResetRequest								
Par135	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par136	clientId	4.2.10.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par137	reason	4.2.10.1	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
ClientResetConfirm								
Par138	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par139	clientId	4.2.10.2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par140	response	4.2.10.2	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
ClientResetIndication								
Par141	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par142	clientId	4.2.10.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par143	reason	4.2.10.3	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
ClientResetResponse								
Par144	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par145	clientId	4.2.10.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par146	response	4.2.10.4	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
ServerResetRequest								
Par147	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par148	serverId	4.2.10.5	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par149	reason	4.2.10.5	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
ServerResetConfirm								
Par150	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par151	clientId	4.2.10.6	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par152	response	4.2.10.6	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
ServerResetIndication								
Par153	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par154	clientId	4.2.10.7	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)

Par155	reason	4.2.10.7	n/a		m o	0x00000x001B 0x8000 - 0xFFFF	m o	0x00000x001B 0x8000 - 0xFFFF
ServerResetResponse								
Par156	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par156	clientId	4.2.10.8	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par157	response	4.2.10.8	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
ServerContinuousFeedSessionRequest								
Par158	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par159	sessionId	4.2.8.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par160	serverId	4.2.8.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par161	Resources	4.2.8.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
ServerContinuousFeedSessionConfirm								
Par162	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par163	sessionId	4.2.8.2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par164	response	4.2.8.2	n/a		m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF	m m m m m o	0x0000 - 0x000B 0x0010 - 0x0019 0x0020 - 0x002A 0x0030 - 0x0038 0x0041 - 0x0042 0x8000 - 0xFFFF
Par165	Resources	4.2.8.2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.2.3)
ClientSessionInProgress								
Par166	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par167	sessionCount	4.2.14.1	n/a		m			
Par168	sessionId	4.2.14.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
ServerSessionInProgress								
Par169	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par170	sessionCount	4.2.14.2	n/a		m			
Par171	sessionId	4.2.14.2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
ClientSessionTransferIndication								
Par172	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par173	sessionId	4.2.13.1	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par174	clientId	4.2.13.1						
Par175	oldServerId	4.2.13.1						
Par176	newServerId	4.2.13.1						
Par177	Resources()	4.2.13.1						

ClientSessionTransferConfirm								
Par187	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par188	sessionId	4.2.13.2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par189	response	4.2.13.2						
Par190	UserData()	4.2.13.2						
ServerSessionTransferRequest								
Par182	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par183	sessionId	4.2.13.3	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par184	destServerId	4.2.13.3						
Par185	baseServerId	4.2.13.3						
Par186	UserData()							
ServerSessionTransferConfirm								
Par187	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par188	sessionId	4.2.13.4	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par189	response	4.2.13.4						
Par190	UserData()							
ServerSessionTransferIndication								
Par191	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par192	sessionId	4.2.13.5	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par193	clientId	4.2.13.5						
Par193	srcServerId	4.2.13.5						
Par194	baseServerId	4.2.13.5						
Par195	Resources()							
Par195	UserData()							
ServerSessionTransferResponse								
Par196	dsmccMessageHeader()	2	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)	m	(as specified by ISO/IEC 13818-6 subclause 4.3.4)
Par197	sessionId	4.2.13.6	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)	m	(as specified by ISO/IEC 13818-6 subclause 4.3)
Par198	response	4.2.13.6	n/a		m	(as specified by ISO/IEC 13818-6 subclause 4.3)		
Par199	Resources()	4.2.7.3						
Par200	UserData()	4.2						

4.3.8 Parameter Support for DSM-CC User-to-Network Download Messages

Table 12 – DSM-CC User-to-Network Download Message Parameters

Item	Parameter.	ISO/IEC 13818-6 reference	Role: Client		Role: Server		Role: SRM	
			Status	values allowed	Status	Values allowed	Status	Values allowed
dsmccDownloadDataHeader								
Par1	protocolDiscriminator	7.2.2.1	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par2	dsmccType	7.2.2.1	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par3	messageId	7.3	m	(see ISO/IEC 13818-6 subclause 7.3)	m	(see ISO/IEC 13818-6 subclause 7.3)	n/a	
Par4	downloadId	7.2.2.1	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	n/a	
Par5	reserved	7.2.2.1	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	n/a	
Par6	adaptationLength	7.2.2.1	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par7	messageLength	7.2.2.1	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par8	dsmccAdaptationHeader()	2.1	o	(see ISO/IEC 13818-6 subclause 2.1)	o	(see ISO/IEC 13818-6 subclause 2.1)	n/a	
DownloadInfoRequest								
Par9	dsmccMessageHeader()	2	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par10	bufferSize	7.3.1	m	(see ISO/IEC 13818-6 subclause 7.3.1)	m	(see ISO/IEC 13818-6 subclause 7.3.1)	n/a	
Par11	maximumBlockSize	7.3.1	m	(see ISO/IEC 13818-6 subclause 7.3.1)	m	(see ISO/IEC 13818-6 subclause 7.3.1)	n/a	
Par12	CompatibiltyDescriptor()	6.1	m	(see ISO/IEC 13818-6 subclause 6.1)	m	(see ISO/IEC 13818-6 subclause 6.1)	n/a	
Par13	privateDataLength	7.3.1	m	(see ISO/IEC 13818-6 subclause 7.3.1)	m	(see ISO/IEC 13818-6 subclause 7.3.1)	n/a	
Par14	privateDataByte	7.3.1	m	(see ISO/IEC 13818-6 subclause 7.3.1)	m	(see ISO/IEC 13818-6 subclause 7.3.1)	n/a	
DownloadInfoResponse DownloadInfoIndication								
Par15	dsmccMessageHeader()	2	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par16	downloadId	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par17	blockSize	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par18	windowSize	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par19	ackPeriod	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par20	tCDownloadWindow	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par21	tCDownloadScenario	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par22	CompatibiltyDescriptor()	6.1	m	(see ISO/IEC 13818-6 subclause 6.1)	m	(see ISO/IEC 13818-6 subclause 6.1)	n/a	

Par23	numberOfModules	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)	n/a	
Par24	moduleId	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par25	moduleSize	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par26	moduleVersion	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par27	moduleInfoLength	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par28	moduleInfoByte	7.3.2	o	(see ISO/IEC 13818-6 subclause 7.3.2)	o	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par29	privateDataLength	7.3.2	m	(see ISO/IEC 13818-6 subclause 7.3.2)	m	(see ISO/IEC 13818-6 subclause 7.3.2)		
Par30	privateDataByte	7.3.2	o	(see ISO/IEC 13818-6 subclause 7.3.2)	o	(see ISO/IEC 13818-6 subclause 7.3.2)		
DownloadDataBlock								
Par31	dsmccDownloadDataHeader()	7.2.2.1	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	n/a	
Par32	moduleId	7.3.3	m	(see ISO/IEC 13818-6 subclause 7.3.3)	m	(see ISO/IEC 13818-6 subclause 7.3.3)	n/a	
Par33	moduleVersion	7.3.3	m	(see ISO/IEC 13818-6 subclause 7.3.3)	m	(see ISO/IEC 13818-6 subclause 7.3.3)	n/a	
Par34	blockNumber	7.3.3	m	(see ISO/IEC 13818-6 subclause 7.3.3)	m	(see ISO/IEC 13818-6 subclause 7.3.3)	n/a	
Par35	blockDataByte	7.3.3	m	(see ISO/IEC 13818-6 subclause 7.3.3)	m	(see ISO/IEC 13818-6 subclause 7.3.3)	n/a	
DownloadDataRequest								
Par36	dsmccDownloadDataHeader()	7.2.2.1	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	m	(see ISO/IEC 13818-6 subclause 7.2.2.1)	n/a	
Par37	moduleId	7.3.4	m	(see ISO/IEC 13818-6 subclause 7.3.4)	m	(see ISO/IEC 13818-6 subclause 7.3.4)	n/a	
Par38	blockNumber	7.3.4	m	(see ISO/IEC 13818-6 subclause 7.3.4)	m	(see ISO/IEC 13818-6 subclause 7.3.4)	n/a	
Par39	downloadReason	7.3.4	m	(see ISO/IEC 13818-6 subclause 7.3.4)	m	(see ISO/IEC 13818-6 subclause 7.3.4)	n/a	
DownloadCancel								
Par40	dsmccMessageHeader()	2	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par41	downloadId	7.3.5	m	(see ISO/IEC 13818-6 subclause 7.3.5)	m	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	
Par42	moduleId	7.3.5	m	(see ISO/IEC 13818-6 subclause 7.3.5)	m	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	
Par43	blockNumber	7.3.5	m	(see ISO/IEC 13818-6 subclause 7.3.5)	m	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	
Par44	downloadCancelReason	7.3.5	m	(see ISO/IEC 13818-6 subclause 7.3.5)	m	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	
Par45	privateDataLength	7.3.5	m	(see ISO/IEC 13818-6 subclause 7.3.5)	m	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	

Par46	privateDataByte	7.3.5	o	(see ISO/IEC 13818-6 subclause 7.3.5)	o	(see ISO/IEC 13818-6 subclause 7.3.5)	n/a	
DownloadServerInitiate								
Par47	dsmccMessageHeader()	2	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	n/a	
Par48	serverId	7.3.6	m	(see ISO/IEC 13818-6 subclause 7.3.6)	m	(see ISO/IEC 13818-6 subclause 7.3.6)	n/a	
Par49	compatibilitiesLength	7.3.6	m	(see ISO/IEC 13818-6 subclause 7.3.6)	m	(see ISO/IEC 13818-6 subclause 7.3.6)	n/a	
Par50	privateDataLength	7.3.6	m	(see ISO/IEC 13818-6 subclause 7.3.6)	m	(see ISO/IEC 13818-6 subclause 7.3.6)	n/a	
Par51	privateDataByte	7.3.6	o	(see ISO/IEC 13818-6 subclause 7.3.6)	o	(see ISO/IEC 13818-6 subclause 7.3.6)	n/a	

4.3.9 Parameter support for DSM-CC User-to-Network SDB CCP Messages

Table 13 – DSM-CC User-to-Network SDB Channel Change Protocol Parameters

Item	Parameter.	ISO/IEC 13818-6 reference	Role: Client		Role: Client		Role: SDB Server	
			Status	values allowed	Status	Values allowed	Status	Values allowed
PrivateData()								
Par1	privateDataLength	10.2.1	M	(see ISO/IEC 13818-6 subclause 10.2.1)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.1)
Par2	privateDataByte	10.2.1	O	(see ISO/IEC 13818-6 subclause 10.2)	n/a		o	(see ISO/IEC 13818-6 subclause 10.2)
SDBProgramSelectRequest								
Par3	sessionId	10.2.3.1	M	(see ISO/IEC 13818-6 subclause 10.2.3.1)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.1)
Par4	broadcastProgramId	10.2.3.1	M	(see ISO/IEC 13818-6 subclause 10.2.3.1)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.1)
Par5	PrivateData()	10.2.1	O	(see ISO/IEC 13818-6 subclause 10.2.1)	n/a		o	(see ISO/IEC 13818-6 subclause 10 2.1)
SDBProgramSelectConfirm								
Par6	sessionId	10.2.3.2	M	(see ISO/IEC 13818-6 subclause 10.2.3.2)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.2)
Par7	response	10.2.3.2	M	(see ISO/IEC 13818-6 subclause 10.2.3.2)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.2)
Par8	broadcastProgramId	10.2.3.2	M	(see ISO/IEC 13818-6 subclause 10.2.3.2)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.2)
Par9	PrivateData()	10.2.1	O	(see ISO/IEC 13818-6 subclause 10.2.1)	n/a		o	(see ISO/IEC 13818-6 subclause 10.2.1)
SDBProgramSelectIndication								
Par10	sessionId	10.2.3.3	M	(see ISO/IEC 13818-6 subclause 10.2.3.3)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.3)
Par11	response	10.2.3.3	M	(see ISO/IEC 13818-6 subclause 10.2.3.3)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.3)
Par12	broadcastProgramId	10.2.3.3	M	(see ISO/IEC 13818-6 subclause 10.2.3.3)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.3)
Par13	PrivateData()	10.2.1	O	(see ISO/IEC 13818-6 subclause 10.2.1)	n/a		o	(see ISO/IEC 13818-6 subclause 10.2.1)
SDBProgramSelectResponse								

Par14	sessionId	10.2.3.4	M	(see ISO/IEC 13818-6 subclause 10.2.3.4)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.4)
Par15	response	10.2.3.4	M	(see ISO/IEC 13818-6 subclause 10.2.3.4)	n/a		m	(see ISO/IEC 13818-6 subclause 10.2.4)
Par16	PrivateData()	10.2.1	O	(see ISO/IEC 13818-6 subclause 10.2.1)	n/a		o	(see ISO/IEC 13818-6 subclause 10.2.1)

4.3.10 Parameter Support for DSM-CC U-N Pass-Thru Messages

Table 14 – DSM-CC User-to-Network Pass-Thru Message Parameters

Item	Parameter	ISO/IEC 13818-6 reference	Role: Client		Role: Client		Role: SRM	
			Status	Values allowed	Status	Values allowed	Status	Values allowed
PassThruData								
Par1	passThruDataLength	12.2.1	M	(see ISO/IEC 13818-6 subclause 12.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.1)
Par2	passThruDataByte	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)
PassThruRequest								
Par3	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par4	userId	12.2.2.1	M	(see ISO/IEC 13818-6 subclause 12.2.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.2.1)
Par4	passThruType	12.2.2.1	M	(see ISO/IEC 13818-6 subclause 12.2.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.2.1)	m	(see ISO/IEC 13818-6 subclause 12.2.2.1)
Par5	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)
PassThruIndication								
Par6	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par7	userId	12.2.2.2	M	(see ISO/IEC 13818-6 subclause 12.2.2.2)	m	(see ISO/IEC 13818-6 subclause 12.2.2.2)	m	(see ISO/IEC 13818-6 subclause 12.2.2.2)
Par8	passThruType	12.2.2.2	M	(see ISO/IEC 13818-6 subclause 12.2.2.2)	m	(see ISO/IEC 13818-6 subclause 12.2.2.2)	m	(see ISO/IEC 13818-6 subclause 12.2.2.2)
Par9	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)
PassThruReceiptRequest								
Par10	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par11	sourceUserId	12.2.2.3	M	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)
Par12	destinationUserId	12.2.2.3	M	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)
Par13	passThruType	12.2.2.3	M	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)	m	(see ISO/IEC 13818-6 subclause 12.2.2.3)
Par14	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)
PassThruReceiptConfirm								
Par15	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par16	response	12.2.2.4	M	(see ISO/IEC 13818-6 subclause 12.2.2.4)	m	(see ISO/IEC 13818-6 subclause 12.2.2.4)	m	(see ISO/IEC 13818-6 subclause 12.2.2.4)
Par17	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)

PassThruReceiptIndication								
Par18	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par19	userId	12.2.2.5	M	(see ISO/IEC 13818-6 subclause 12.2.2.5)	m	(see ISO/IEC 13818-6 subclause 12.2.2.5)	m	(see ISO/IEC 13818-6 subclause 12.2.2.5)
Par20	passThruType	12.2.2.5	M	(see ISO/IEC 13818-6 subclause 12.2.2.5)	m	(see ISO/IEC 13818-6 subclause 12.2.2.5)	m	(see ISO/IEC 13818-6 subclause 12.2.2.5)
Par21	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)
PassThruReceiptResponse								
Par22	dsmccMessageHeader()	2	M	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)	m	(see ISO/IEC 13818-6 clause 2)
Par23	response	12.2.2.6	M	(see ISO/IEC 13818-6 subclause 12.2.2.6)	m	(see ISO/IEC 13818-6 subclause 12.2.2.6)	m	(see ISO/IEC 13818-6 subclause 12.2.2.6)
Par24	PassThruData()	12.2.1	O	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)	o	(see ISO/IEC 13818-6 subclause 12.2.1)

4.4 DSM-CC User-to-User Functional Unit

4.4.1 Roles of DSM-CC User-to-User

Table 15 – DSM-CC User-to-User Role PICS

Item	Role Does the implementation support...	Conditions for status	Status	ISO/IEC 13818-6 reference
R1	the client role		o.1	0.6
R2	the server role		o.1	0.6
o.1	Support of one, and only one, of these options is required.			

4.4.2 Major capabilities of DSM-CC User-to-User

Table 16 – DSM-CC User-to-User Major Capability PICS

Item	Major capabilities Does the implementation support...	ISO/IEC 13818-6 reference	Role: Client		Role: Server	
			Conditions for status	Status	Conditions for status	Status
mc 21	user-to-user procedures	5		o.1		o.2
mc 22	user-to-user session	8		o		o
mc 23	user-to-user download	8		o		o
mc 24	user-to-user object carousel	11		o.1		o.2
mc 25	user-to-user local objects	TBA				
o.1	Support of one, and only one, of these options is required					
o.2	Support of one, and only one, of these options is required					

4.4.2.1 Parameter Support for DSM-CC User-to-User Object Carousel

Table 17 – DSM-CC User-to-User Object Carousel Parameters

Item	Parameter	ISO/IEC 13818-6 reference	Role: Client		Role: Server	
			status	Values allowed	status	Values Allowed
	BIOP::GenericObjectMessage()					
Par1	magic	11.3.2.1	m	0x42494F50	m	0x42494F50

Par2	biop_version.major	11.3.2.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)
Par3	biop_version.minor	11.3.2.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)
Par4	byte_order	11.3.2.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)*	m	(See ISO/IEC 13818-6 subclause 1.3.2.1)*
Par5	message_type	11.3.2.1	m	0x00	m	0x00
Par6	message_size	11.3.2.1	m	0x00000000-0xFFFFFFFF	m	0x00000000-0xFFFFFFFF
Par7	objectKey_length	11.3.2.1	m	0x00000001-0xFFFFFFFF**	m	0x00000001-0xFFFFFFFF**
Par8	objectKey_data	11.3.2.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)
Par9	objectKind_length	11.3.2.1	m	0x00000004-0xFFFFFFFF**	m	0x00000004-0xFFFFFFFF**
Par10	objectKind_data	11.3.2	m	(See ISO/IEC 13818-6 subclause 11.3.2)	m	(See ISO/IEC 13818-6 subclause 11.3.2)
Par11	objectInfo_length	11.3.2.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)
Par12	objectInfo_data	11.3.2.1	o	(See ISO/IEC 13818-6 subclause 11.3.2.1)	o	(See ISO/IEC 13818-6 subclause 11.3.2.1)
Par13	serviceContextList_count	5.6.4	m	0x00-0xFF	m	0x00-0xFF
Par14	serviceContextList_data	5.6.4	o		o	
Par15	messageBody_length	11.3.2.1	m	0x00000000-0xFFFFFFFF	m	0x00000000-0xFFFFFFFF
*	Value 0x00 is advised.					
**	Value 0x00000004 is advised.					
	BIOP::FileMessage()					
Par1	BIOP::GenericObjectMessage	11.3.2.1/F.3.2	m		m	
Par2	content_length	11.3.2.3	m	(See ISO/IEC 13818-6 subclause 11.3.2.3)	m	0x00000000-0xFFFFFFFF
Par2-n	content_byte	11.3.2.3	o	0x00-0xFF	o	0x00-0xFF
	objectKind of GenericObjectMessage	11.3.2.3	o m	'DSM::File' 'fil'	o m	'DSM::File' 'fil'
	objectInfo of GenericObjectMessage	11.3.2.3	m	DSM::FileContentSize	m	DSM::FileContentSize
	DSM::FileContentSize					
Par1	content_size	5.6.5.4/5.5.1.4	m	0x0000000000000000-0xFFFFFFFFFFFFFFFF	m	0x0000000000000000-0xFFFFFFFFFFFFFFFF
	BIOP::DirectoryMessage()					
Par1	BIOP::GenericObjectMessage	11.3.2.1/F.3.2	m		m	
Par2	bindings_count	11.3.2.2	m	0x0000-0xFFFF	m	0x0000-xFFFF
Par3	BIOP::Name()	11.3.2.2	m		m	
Par4	bindingType	11.3.2.2	m	0x01-0x02*	m	0x01-0x02
Par5	IOP::IOR()	11.3.1	m		m	
Par6	objectInfo_length	11.3.2.2	m	0x0000-0xFFFF	m	0x0000-xFFFF
Par7	objectInfo_byte	11.3.2.2	o	0x00-0xFF	o	0x00-0xFF
	objectKind of GenericObjectMessage	11.3.2.2	o m	'DSM::Directory' 'dir'	o m	'DSM::Directory' 'dir'
*	0x01 = nobject (not bound to dir) 0x02 = ncontext (bound to dir)					
	BIOP::Name()					
Par1	nameComponents_count	11.3.2.2	m	0x00-0xFF	m	0x00-0xFF
Par2	id_length	11.3.2.2	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)
Par3	id_data	11.3.2.2	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)
Par4	kind_length	11.3.2.2	m	0x04-0xFF*	m	0x04-0xFF
Par5	kind_data	11.2.1	m	(See ISO/IEC 13818-6 subclause 11.2.1)	m	(See ISO/IEC 13818-6 subclause 11.2.1)
*	Value 0x04 is advised.					

	IOP::IOR()					
Par1	type_id_length	5.6.3/F.3.1	m	0x04-0xFF*	m	0x04-0xFF*
Par2	type_id_byte	5.6.3/F.3.1	m	0x00-0xFF	m	0x00-0xFF
Par3	alignment_gap	5.6.3	o	**	o	**
Par4	taggedProfiles_count	5.6.3	m	0x00000001-0xFFFFFFFF	m	0x00000001-0xFFFFFFFF
Par5	TaggedProfile()	5.6.3				
*	Value 0x04 is advised.					
**	Using short Id kinds does away with the necessity of alignment stuffing.					
	TaggedProfile()					
Par1	profileId_tag	5.6.3	o	0x49534F00-0x49534F05	o	0x49534F00-0x49534F05
			m	0x49534F06*	m	0x49534F06*
			o	0x49534F07-0x49534F0F (See ISO/IEC 13818-6 subclause 5.6.3)	o	0x49534F07-0x49534F0F (See ISO/IEC 13818-6 subclause 5.6.3)
Par2	profile_data_length	5.6.3.5/F.3.1	m	(See ISO/IEC 13818-6 subclause 5.6.3.5)	m	(See ISO/IEC 13818-6 subclause 5.6.3.5)
Par3	profile_data		**		**	
*	BIOPProfileBody.					
**	Dependant on profileId_tag.					
	BIOPProfileBody()					
Par3	byte_order	5.6.3.5/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)*	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)*
Par4	liteComponents _count	5.6.3.5/F.3.1	m	0x02-0xFF	m	0x02-0xFF
Par5	BIOP::ObjectLocation()	5.6.3.5/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)
Par6	BIOP::ConnBinder()	5.6.3.5/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.2)	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.2)
*	Value 0x00 is advised.					
	BIOP::ObjectLocation()					
Par1	componentId_tag	11.3.1.1.1/F.3.1	m	0x49534F50	m	0x49534F50
Par2	component_data_length	11.3.1.1.1/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)	m	(See ISO/IEC 13818-6 subclause11.3.1.1.1)
Par3	carouselId	11.3.1.1.1/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.2.2)	m	(See ISO/IEC 13818-6 subclause 11.2.2)
Par4	moduleId	11.3.1.1.1/F.3.1	m	0x0000-0xFFFF	m	0x0000-0xFFFF
Par5	version.major	11.3.1.1.1/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)
Par6	version.minor	11.3.1.1.1/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)	m	(See ISO/IEC 13818-6 subclause 11.3.1.1.1)
	objectKey_length	11.3.1.1.1/F.3.1	m	0x00000001-0xFFFFFFFF*	m	0x00000001-0xFFFFFFFF*
	objectKey_data	11.3.1.1.1/F.3.1	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)	m	(See ISO/IEC 13818-6 subclause 11.3.2.1)
*	Value 0x00000004 is advised.					
	BIOP::ConnBinder()					
Par1	componentId_tag	11.3.1.1.2/F.3.1	m	0x49534F40	m	0x49534F40
Par2	component_data_length	11.3.1.1.2/F.3.1	m	(See ISO/IEC 13818-6 subclause 5.6.3)	m	(See ISO/IEC 13818-6 subclause 5.6.3)
Par3	taps_count	11.3.1.1.2/F.3.1	m	0x01-0xFF	m	0x01-0xFF
Par4	BIOP::Tap()	5.6.1/11.2.6/11.3.1.1.2				
	BIOP::Tap()					
Par1	use	5.6.1	m	n/a	m	n/a

Par2	id	5.6.1	m	(See ISO/IEC 13818-6 subclause 5.6.1)	m	(See ISO/IEC 13818-6 subclause 5.6.1)
Par3	association_Tag	5.6.1	m	(See ISO/IEC 13818-6 subclause 5.6.1)	m	(See ISO/IEC 13818-6 subclause 5.6.1)
Par4	selector_length	5.6.1.1	m	(See ISO/IEC 13818-6 subclause 5.6.1.1)	m	(See ISO/IEC 13818-6 subclause 5.6.1.1)
Par5	selector_type	5.6.1.1	o	0x0001	o	0x0001
Par6	transactionId	5.6.1.1	o	(See ISO/IEC 13818-6 subclause 5.6.1.1)	o	(See ISO/IEC 13818-6 subclause 5.6.1.1)
Par7	timeout	5.6.1.1	o	(See ISO/IEC 13818-6 subclause 5.6.1.1)	o	(See ISO/IEC 13818-6 subclause 5.6.1.1)
	BIOP::ServiceGatewayMessage()	11.3.2.1/F.3.2				
Par1	BIOP::GenericObjectMessage		m		m	
Par2	*	11.3.2.2	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)	m	(See ISO/IEC 13818-6 subclause 11.3.2.2)
	objectKind of GenericObjectMessage	11.3.2.5	o m	'DSM::Service Gateway' 'srg'	o m	'DSM::Service Gateway' 'srg'
*	The body of the ServiceGatewayMessage is equal to that of the Directory message.					
	BIOP::StreamMessage()	11.3.2.1/F.3.2				
Par1	BIOP::GenericObjectMessage		m		m	
Par2	taps_count	11.3.1.1.2/F.3.1	m	0x00-0xFF	m	0x00-0xFF
Par3	BIOP::Tap()	5.6.1/11.2.6/11.3.1.1.2	o	(See ISO/IEC 13818-6 subclause 5.6.1)	o	(See ISO/IEC 13818-6 subclause 5.6.1)
	objectKind of GenericObjectMessage	11.3.2.4	o m	'DSM::Stream' 'str'	o m	'DSM::Stream' 'str'
	objectInfo of GenericObjectMessage	11.3.2.4	m	DSM::Stream::Info_T		
	DSM::Stream::Info_T					
Par1	aDescription_length	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause .5.1.3.1)
Par2	aDescription_bytes	5.5.1.3.1	o	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	o	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)
Par3	duration.aSeconds	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)
Par4	duration.aMicroSeconds	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)
Par5	audio	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)
Par6	Video	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)
Par7	Data	5.5.1.3.1	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)	m	(See ISO/IEC 13818-6 subclause 5.5.1.3.1)

4.4.3 Subsidiary Capabilities Related to DSM-CC User-to-User Procedures

Table 18 – DSM-CC User-to-User Subsidiary Capability PICS

Item	Major capabilities Does the implementation support...	ISO/IEC 13818-6 reference	Role: Client		Role: Server	
			Conditions for status	Status	Conditions for status	Status
User-to-user system environment & Common Definitions						
sc 1	related aspects of the user-to-user system environment including common types, exceptions, access control and library functions	5.2, 5.4		o		o
Overview of the Interface Definition Language & Service Interoperability Interfaces						
sc 2.1	remote procedure call mechanism based on UNO	5.3, 5.6.2		o.1		o.3
sc 2.2	other remote procedure call	5.3, 5.6.2		o.1		o.3

sc 3	recommended format for service context lists	5.6.4				
sc 4.1	Interoperable Object Reference format	5.6.3		m		m
sc 4.2	IIOP profile body	5.6.3		m		m
sc 4.3	tagged profiles for other RPC mechanisms	5.6.3	Sc 2.2 NOT Sc 2	O N/A	Sc 2.2 NOT Sc 2	O N/A
sc 5.1	processing of additional protocol profile body for composite resources	5.6.3		M		M
sc 5.2	resource to connection association (connection binder)	5.6.1		M		M
sc 5.3	inclusion of a preferred association tag in requests	5.6.1		O		O
Application Boot Process						
sc 6.1	client related aspects of the user-to-network assumptions and requirements	5.7		M		M
sc 6.3	DownloadInfoRequest in association with ClientSessionSetUpRequest	5.7.2.3		O		O
sc 6.4	session suspension/resumption	5.7.4 - Annex L		O		O
Core client-service interfaces						
sc 7.1	Base interface, including Base IsA	5.5.1.1		M		M
sc 7.2	DSM Base Close	5.5.1.1.2		M		M
sc 7.3	DSM Base Destroy	5.5.1.1.3		M		M
sc 8	Access interface	5.5.1.2, 5.5.1.2.1		M		M
sc 9.1	Directory interface	5.5.1.5, 5.5.1.5.1		M		M
sc 9.2	DSM Directory list	5.5.1.5.3		M		M
sc 9.3	DSM Directory resolve	5.5.1.5.4		M		M
sc 9.4	DSM Directory bind	5.5.1.5.5		O		M
sc 9.5	DSM Directory bind_context	5.5.1.5.6		O		M
sc 9.6	DSM Directory rebind	5.5.1.5.7		O		M
sc 9.7	DSM Directory rebind_context	5.5.1.5.8		O		M
sc 9.8	DSM Directory unbind	5.5.1.5.9		O		M
sc 9.9	DSM Directory new_context	5.5.1.5.10		O		M
sc 9.10	DSM Directory bind_new_context	5.5.1.5.11		O		M
sc 9.11	DSM Directory destroy	5.5.1.5.12		O		M
sc 9.12	DSM Directory open	5.5.1.5.13		O		M
sc 9.13	DSM Directory close	5.5.1.5.14		O		M
sc 9.14	DSM Directory get	5.5.1.5.15		O		M
sc 9.15	DSM Directory put	5.5.1.5.16		O		M
sc 10.1	Stream interface	5.5.1.3, 5.5.1.3.1, 5.5.1.3.2, 5.5.1.3.3		O		M
sc 10.2	complete stream interface state machine	5.5.1.3.4.3	Sc 10.1 NOT Sc 10.1	o.2 N/A	Sc 10.1 NOT Sc 10.1	o.4 N/A
sc 10.3	basic stream interface state machine	5.5.1.3.4.2	Sc 10.1 NOT Sc 10.1	o.2 N/A	Sc 10.1 NOT Sc 10.1	o.4 N/A
sc 10.4	DSM Stream Pause	5.5.1.3.5	Sc 10.1 NOT Sc 10.1	M N/A	Sc 10.1 NOT Sc 10.1	M N/A
sc 10.5	DSM Stream Resume	5.5.1.3.6	Sc 10.1 NOT Sc 10.1	M N/A	Sc 10.1 NOT Sc 10.1	M N/A
sc 10.6	DSM Stream Status	5.5.1.3.7	Sc 10.1 NOT Sc 10.1	O N/A	Sc 10.1 NOT Sc 10.1	O N/A
sc 10.7	DSM Stream Reset	5.5.1.3.8	Sc 10.1 NOT Sc 10.1	M N/A	Sc 10.1 NOT Sc 10.1	M N/A
sc 10.8	DSM Stream Jump	5.5.1.3.9	Sc 10.2 NOT Sc 10.2	M N/A	Sc 10.2 NOT Sc 10.2	M N/A
sc 10.9	DSM Stream Play	5.5.1.3.10	Sc 10.2 NOT Sc 10.2	M N/A	Sc 10.2 NOT Sc 10.2	M N/A
sc 11.1	Event interface	5.5.2.2	Sc 10.1 NOT Sc 10.1	O N/A	Sc 10.1 NOT Sc 10.1	O N/A
sc 11.2	DSM Event subscribe	5.5.2.2.3	Sc 11.1 NOT Sc 11.1	M N/A	Sc 11.1 NOT Sc 11.1	M N/A
sc 11.3	DSM Event unsubscribe	5.5.2.2.4	Sc 11.1 NOT Sc 11.1	M N/A	Sc 11.1 NOT Sc 11.1	M N/A
sc 12.1	File interface	5.5.1.4		M		M
sc 12.2	DSM File Read	5.5.1.4.3		M		M
sc 12.3	DSM File Write	5.5.1.4.4		O		M
sc 13.1	ServiceGateway	5.5.1.7	DSM-CC UN NOT DSM-CC UN[1)]	I M		M

sc 14.1	Session	5.5.1.6		M		M
Extended client-service interfaces						
sc 15	Extended interfaces	5.5.2		O		O
o.1	Support of at least one of these options is required.					
o.2	Support of at least one of these options is required.					
o.3	Support of at least one of these options is required.					
o.4	Support of at least one of these options is required.					
Note 1:	DSM-CC UN: If DSM-CC UN is supported as Session Protocol NOT DSM-CC UN: If DSM-CC UN is not supported as Session Protocol					

4.5 Conformance

A PICS which conforms to this specification shall be technically equivalent to the ISO published PICS proforma and shall preserve the numbering and ordering of the items in the ISO PICS proforma.

A PICS which conforms to this part of ISO/IEC 13818 shall:

a) Describe an implementation which conforms to Part 6 of ISO/IEC 13818,

b) Be a conforming PICS proforma, which has been implemented in accordance with the instruction for completion given in the appropriate clause,

c) Include the information necessary to uniquely identify both the supplier and the implementation.

5 The Conformance ATS

This clause contains a conformance abstract test suite (ATS) for the DSM-CC UU protocol as specified in part 6 of ISO/IEC 13818 [4]. It specifies a test method, the test coverage, and the test cases. Each test case contains the Test Purpose, a preamble and the test procedures.

The test procedures are described in text and use arrow graphs to specify the DSM-CC functional message exchanges. A complete set of each conformance abstract test suite for testing DSM-CC U-N functions and DSM-CC U-U functions includes:

- DSM-CC U-N Configuration
- DSM-CC U-N Session
- DSM-CC Download
- DSM-CC U-N SDB-CCP
- DSM-CC U-N Pass-Thru
- DSM-CC U-U Core interfaces
- DSM-CC U-U Extended interfaces

5.1 Test Method

For conformance testing of DSM-CC protocols, the Remote Test Method as defined in ISO/IEC 9646-1 [1] and ISO/IEC 9646-2 [2] is used. The characteristics of the Remote Test Method, shown in Figure 1, are as follows:

- It has only one PCO between the Lower Tester and the physical layer.
- There are no requirements for the Implementation Under Test (IUT).
- This method can be called a "Black Box" test.
- It has a limit in test coverage because it can observe the behaviors of IUT only through a lower interface.

The Lower Tester (LT) is connected to the System Under Test (SUT) through the service provider. It simulates the operation of the end client system. The Service Provider supports the protocol stacks and/or a core network (such as ATM network) for communication between the LT and the Service elements in the SUT. The Upper Tester (UT) is located on the IUT and emulates the operation of applications. The remote test method does not require an UT in the SUT. However, when there are functions available in the SUT that can control the upper interface of the IUT, they should be used. In these cases, the test coordination procedure (TCP) between the LT and the UT is necessary. The UT and the TCP, therefore, are indicated in dotted lines in Figure 1.

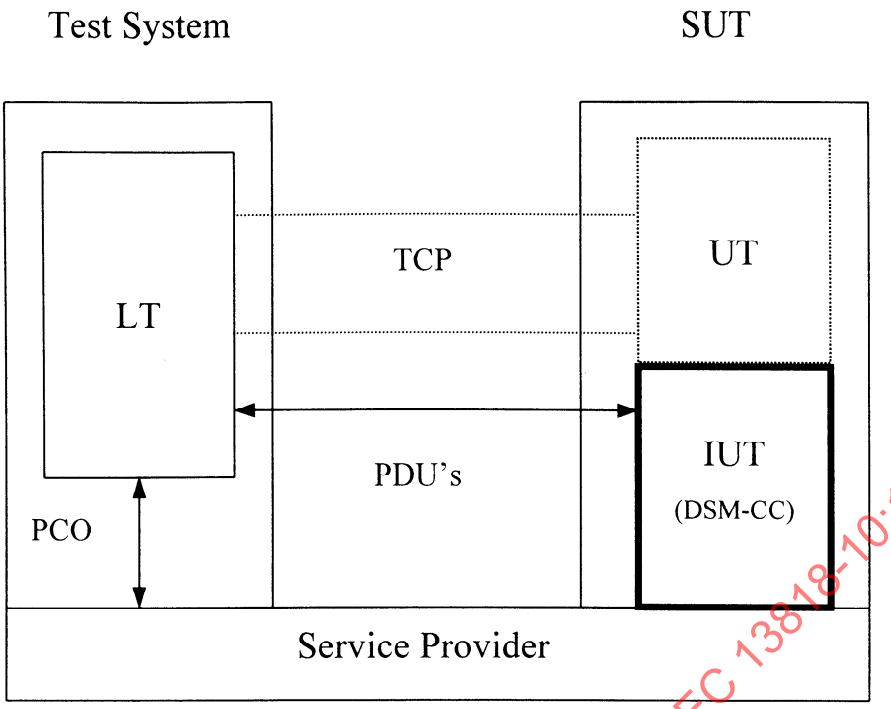


Figure 1 – The Remote Test Method

Since the Remote Test Method as defined in ISO/IEC 9646-1 [1] and ISO/IEC 9646-2 [2] is applied for testing DSM-CC protocols, the assumptions on the Test System including only the capabilities of the LT are considered and specified here.

5.1.1 Test Environment of DSM-CC U-N Functions

Figure 2 shows the function and role of the Test System and the SUT in the test environment for testing DSM-CC U-N functions.

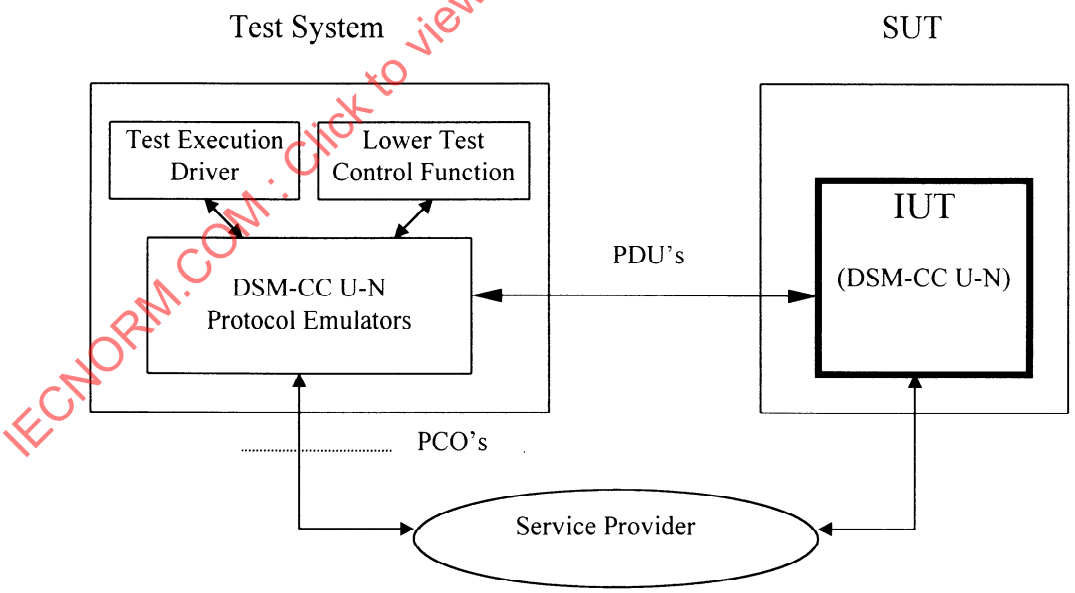


Figure 2 – Test Environment for DSM-CC User-to-Network Functions

The DSM-CC U-N Protocol Emulator serving the role of a Lower Tester (LT) transmits and examines DSM-CC U-N messages (that is, PDU's) in and out of the IUT. That is, the LT sends DSM-CC U-N services requests and waits for the correct responses of the corresponding DSM-CC U-N services of the IUT. In the Test System the test execution driver can also generate the invalid/inopportune requests to test the IUT's error handling capability.

The LT is assumed to have successfully done the required functions as a Network (that is containing a SRM) or a User device (such as, a Client or a Server) for testing DSM-CC U-N message exchanges over U-N network connections to the IUT.

Especially, multiple LT's in the Test System are necessary for testing a Network containing a SRM, which has relay function that is to communicate with more than two devices including a Server device and a Client device. Each LT has to communicate with the appropriate part of the IUT. The LT's can provide only preliminary test result, but cannot assign the final verdict to each test case. Therefore, the Test System should also have the Lower Tester Control Function that is required to coordinate the activity of the LT's and to assign the final verdict to each test case.

5.1.2 Test Environment of DSM-CC U-U Functions

Figure 3 shows the function and role of the Test System and the SUT in the test environment for testing DSM-CC U-U functions.

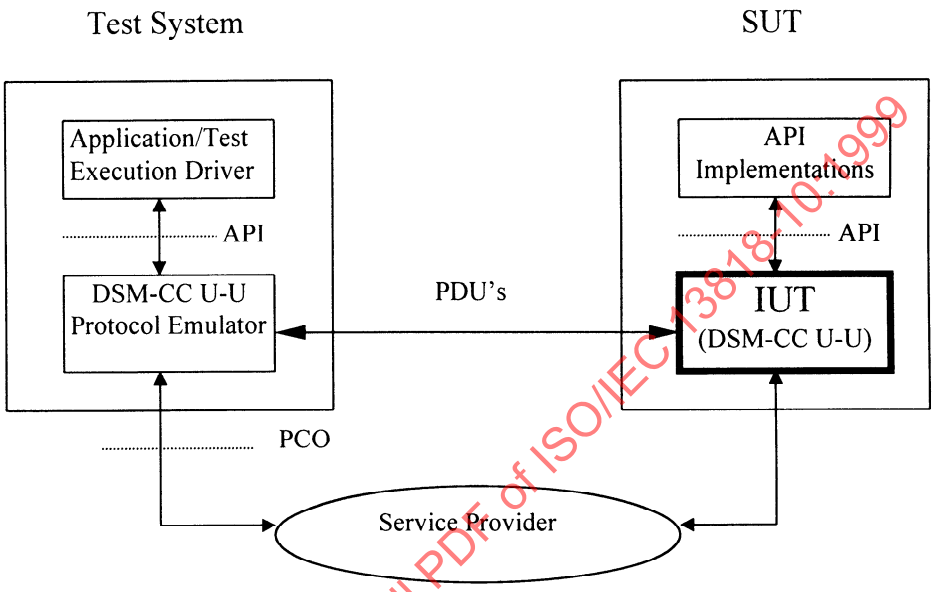


Figure 3 – Test Environment for DSM-CC User-to-User Interfaces

The DSM-CC U-U Protocol Emulator serving the role of a Lower Tester (LT) transmits and examines DSM-CC U-U primitives in and out of the IUT. That is, the LT sends DSM-CC U-U services requests and waits for the correct responses of the corresponding DSM-CC U-U services of the IUT. In the Test System the Application/Test Execution Driver can also generate the invalid/inopportune requests to test the IUT's error handling capability.

The Test System is assumed to have successfully done the application boot process before any of the following test cases are executed. In other words, the DSM-CC U-U library, the session object and the download object must have already existed on the LT. A resident test driver must also have existed and been ready to be launched for executing test cases.

5.2 Test Cases

5.2.1 DSM-CC User-to-Network

This subclause provides two sets of abstract test cases for testing DSM-CC U-N Configuration protocol implemented at a User device and DSM-CC U-N Download protocol implemented at a Server side under the test method and the test environment defined above. The DSM-CC U-N Configuration and Download protocols are optional, therefore each set of test cases for testing the protocols can only be used to test an IUT, depending on the implementation of the given protocol. Each test case consists only of the exchanges of messages with a correct syntax to test the dynamic behaviors of the IUT.

5.2.1.1 DSM-CC User-to-Network Configuration

Table 19 – DSM-CC User-to-Network Configuration Test Cases

Test Case #	Test Case Names	DSM-CC U-N Configuration Types	Reference to ISO/IEC 13818-6
1	Initiating U-N Configuration from the User	User Initiated Configuration Sequences	3.5

2	Initiating U-N Configuration from the Network	Network Initiated Configuration Sequences	3.6
3	Initiating U-N Configuration from the Network broadcasting	Broadcasting of Configuration Messages	3.7
4	Initiating U-N Configuration from the User and the Network simultaneously	Mixed User/Network Initiated Configuration Sequences	3.8

5.2.1.1.1 Test Case 1 - Initiating User-to-Network Configuration from the User

Test Purpose:

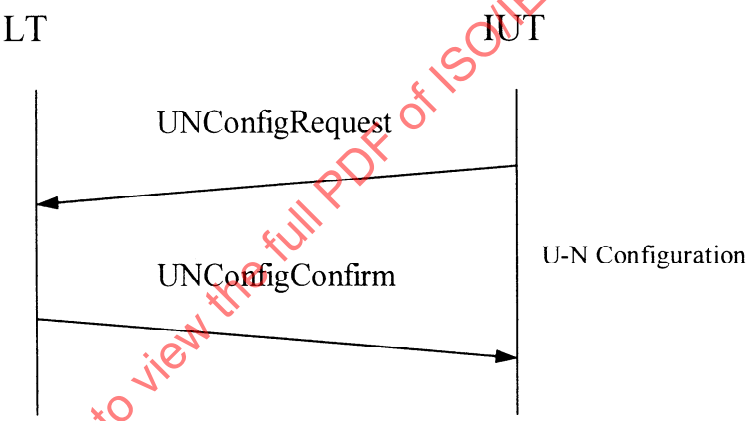
Verify that after establishing a network connection between the SUT and the LT, the User-to-Network Configuration from a User is initiated by sending a 'UNConfigRequest' message to get addresses and other configuration parameters.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. IUT sends a 'UNConfigRequest' message.
2. LT sends a 'UNConfigConfirm' message with the appropriate configuration parameters for the IUT.



Test Verdict:

Pass the test if the addresses and other configuration parameters to operate on the Network are updated in the IUT according to the parameters received from the LT.

5.2.1.1.2 Test Case 2 - Initiating U-N Configuration from the Network

Test Purpose:

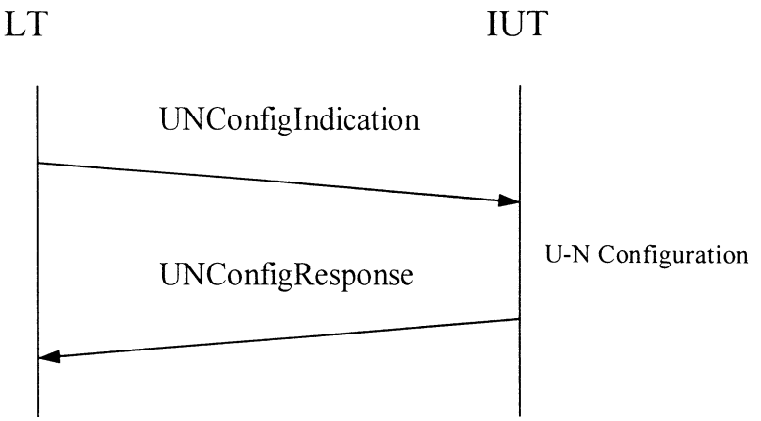
Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'UNConfigIndication' message, the appropriate configuration parameters are updated and a 'UNConfigResponse' message is received.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. LT sends a 'UNConfigIndication' message.
2. LT waits for a 'UNConfigResponse' message.



Test Verdict:

Pass the test if the appropriate configuration parameters are updated in the IUT according to the parameters received from the LT and a 'UNConfigResponse' message with the response value set to indicate the configuration indication accepted is received from the IUT when the IUT accepts the configuration indication, or a 'UNConfigResponse' message with the response value set to indicate the configuration indication rejected is received from the IUT when the IUT does not accept the configuration indication.

5.2.1.1.3 Test Case 3 - Initiating U-N Configuration from the Network broadcasting

Test Purpose:

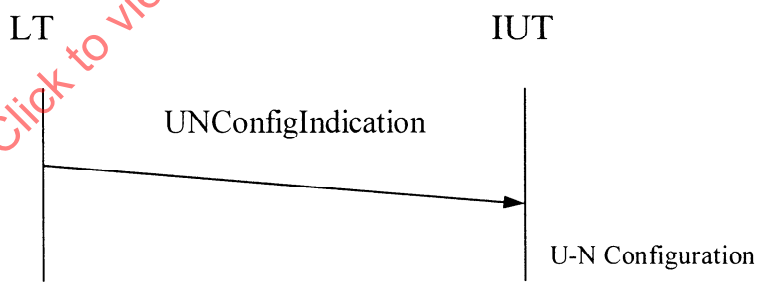
Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'UNConfigIndication' message using a pre-defined broadcasting mechanism, the appropriate configuration parameters are updated in the IUT when a configuration indication matches its deviceId.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. LT sends a 'UNConfigIndication' message using a pre-defined broadcasting mechanism.



Test Verdict:

Pass the test if the appropriate configuration parameters are updated in the IUT when the configuration indication received from the LT matches its deviceId.

5.2.1.1.4 Test Case 4 - Initiating U-N Configuration from the User and the Network simultaneously

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the IUT sends a 'UNConfigRequest' message and the LT sends a 'UNConfigIndication' message simultaneously, the appropriate configuration parameters are updated and a 'UNConfigResponse' message is received.

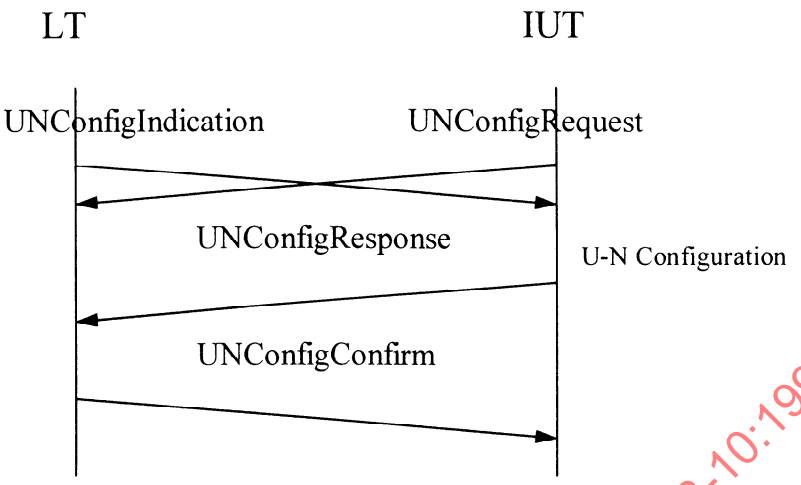
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. IUT sends a 'UNConfigRequest' message and LT sends a 'UNConfigIndication' message simultaneously.

2. LT waits for a 'UNConfigResponse' message.
3. LT sends a 'UNConfigConfirm' message.



Test Verdict:

Pass the test if the process of U-N Configuration initiated from the Network (such as Test Case 2) is carried out when a 'UNConfigIndication' message is received, and the appropriate configuration parameters are again updated in the IUT according to the parameters received from the LT (such a test case 1) when a 'UNConfigConfirm' message is received.

5.2.1.2 DSM-CC User-to-Network Download

Table 20 – DSM-CC User-to-Network Download Test Cases

Test Case #	Test Case Names	DSM-CC U-N Download Types	Reference to ISO/IEC 13818-6
1	Flow-Controlled Download	The Client controls the transfer of data or software via a control channel to the Download Server.	7.4
2	Non-Flow-Controlled Download	The Client doesn't control the transfer of data or software but the transfer is based on mutual agreement about the used transfer parameters.	7.6
3	Data Carousel	Clients will generally only acquire a subset of the transmitted data or software from the network depending on the application.	7.5

5.2.1.2.1 Test Case 1 - Flow-Controlled Download

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, the downloading of a complete set of data or software from a Server is completed in a flow controlled way.

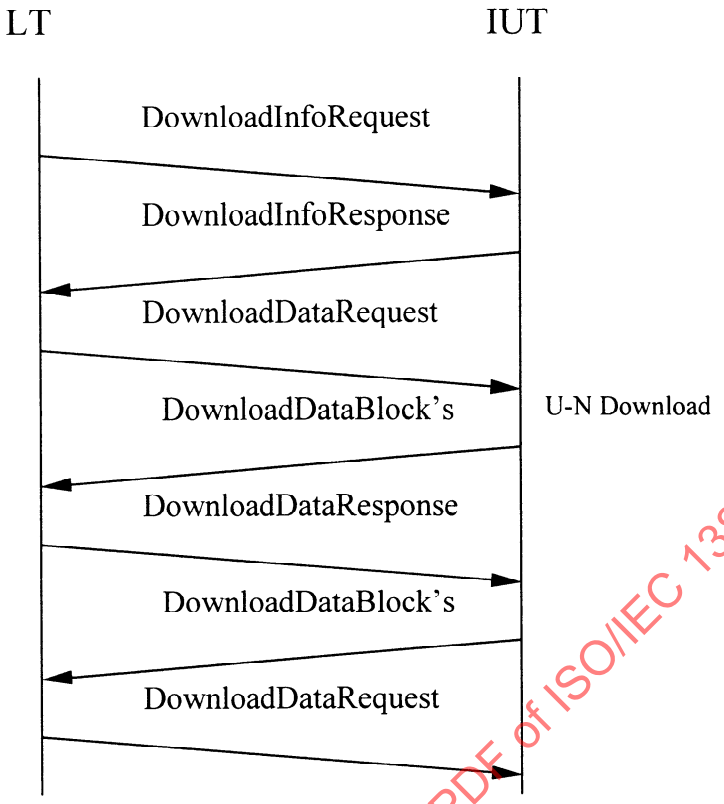
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. LT sends a 'DownloadInfoRequest' message.
2. LT waits for a 'DownloadInfoResponse' message.
3. LT sends a 'DownloadDataRequest' message with a 'rsnStart' downloadReason.
4. LT waits for 'DownloadDataBlock' messages.
5. LT sends a 'DownloadDataRequest' message with a 'rsnAckCont', 'rsnNakRetransBlock', or 'rsnNakRetransWindow' downloadReason depending on the received data blocks.

6. LT waits for 'DownloadDataBlock' messages.
7. LT sends a 'DownloadDataResponse' message with a 'rsnEnd' downloadReason.



Test Verdict:

Pass the test if the downloading of a complete set of requested data or software from the IUT is completed in a flow controlled way after exchanging basic parameter information to be used during the download.

5.2.1.2.2 Test Case 2 - Non-Flow-Controlled Download

Test Purpose:

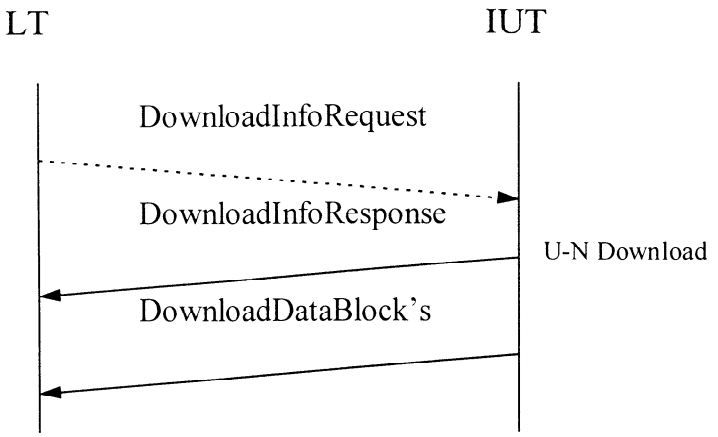
Verify that after establishing a network connection between the SUT and the LT, the downloading of a complete set of data or software from a Server is completed in a non-flow controlled way.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. LT sends a 'DownloadInfoRequest' message (Optional).
2. LT waits for a 'DownloadInfoResponse' message.
3. LT waits for 'DownloadDataBlock' messages.



----------: The message is optional.

Test Verdict:

Pass the test if the downloading of a complete set of data or software from the IUT is completed in a non-flow controlled way after the basic parameter information to be used during the download is received from the IUT.

5.2.1.2.3 Test Case 3 - Data Carousel

Test Purpose:

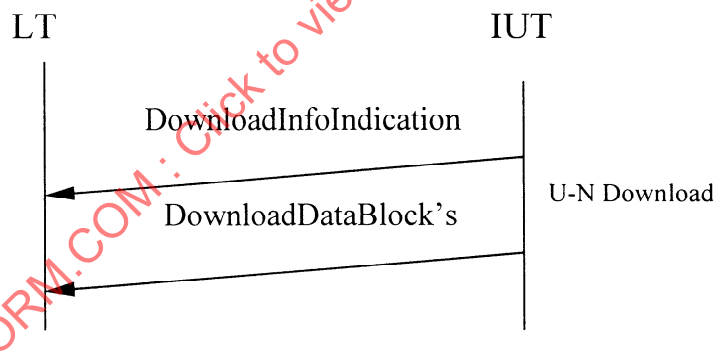
Verify that after establishing a network connection between the SUT and the LT, the periodic transmission of data or software from a Server is accomplished.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1. LT waits for a 'DownloadInfoIndication' message.
2. LT waits for 'DownloadDataBlock' messages.



Test Verdict:

Pass the test if the blocks of all modules within the data carousal are transmitted periodically from the IUT after the basic parameter information to be used during the data carousel is received from the IUT.

5.2.1.3 DSM-CC U-N SDB-CCP

Table 21 – DSM-CC User-to-Network SDB CCP Test Cases

Test Case #	Test Case Names	DSM-CC U-N SDB-CCP Functions	Reference to ISO/IEC 13818-6
1	Selecting a broadcast program to be provided	The Client initiates to select a broadcast program to be provided.	10.3.1
2	Determining to switch to another broadcast program	The SDB server provides the new broadcast program to the Client in order to switch to another broadcast program.	10.3.2

5.2.1.3.1 Test Case 1 - Selecting a broadcast program to be provided

Test Purpose:

Verify that after establishing a communication path for providing the SDB service between the SUT and the LT, when the LT sends a 'SDBProgramSelectRequest' message, the connection for the selected broadcast program is established.

Test Preamble:

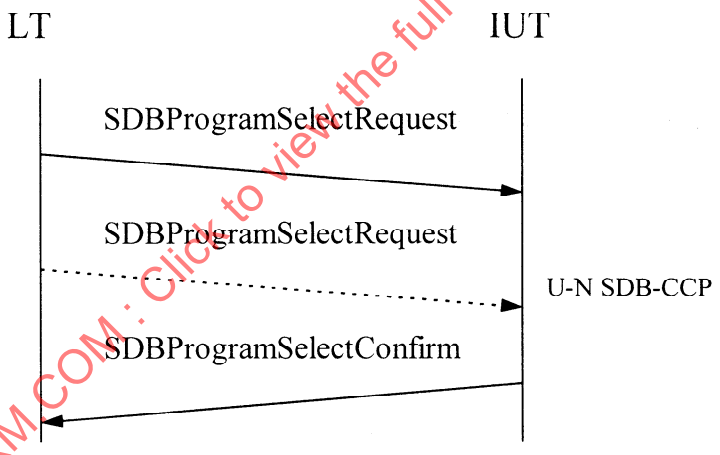
Establish a communication path for providing the SDB service between the SUT and the LT.

Test Procedure:

1) LT sends a 'SDBProgramSelectRequest' message.

2) LT sends again a 'SDBProgramSelectRequest' message. (Optional)

3) LT waits for a 'SDBProgramSelectConfirm' message.



..... : The message may be sent only when the timer tMsg expires.

Test Verdict:

Pass the test if a 'SDBProgramSelectConfirm' message with the response code set to the value such as 'rspNoSession', 'rspFormatError', 'rspBcProgramOutOfService', 'rspRedirect', 'rspEntitlementFailure', 'rspNoServerResponse', or 'rspNoNetworkResponse' is received from the IUT depending on the payload of the received message when the IUT can not accept the request,

or a 'SDBProgramSelectConfirm' message with the sessionId set to the value of the sessionId of a 'SDBProgramSelectRequest' message and the response code set to 'rspOk' is received from the IUT when the IUT can accept the request.

5.2.1.3.2 Test Case 2 - Determining to switch to another broadcast program

Test Purpose:

Verify that after establishing a communication path for providing the SDB service between the SUT and the LT, the IUT initiates to switch to another broadcast program sending a 'SDBProgramSelectIndication' message.

Test Preamble:

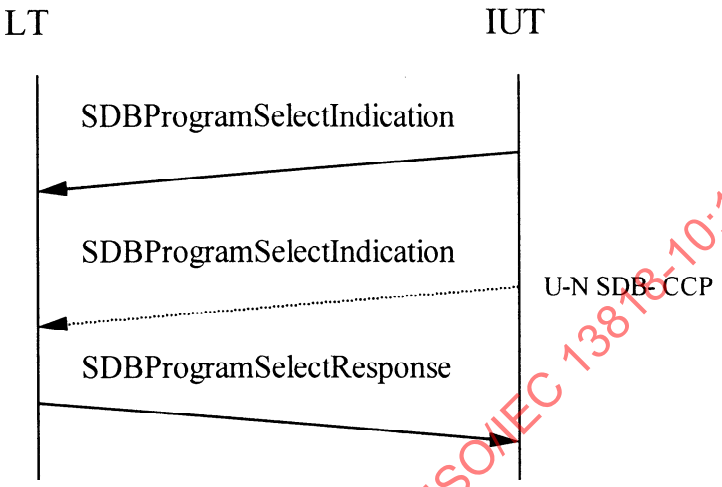
Establish a communication path for providing the SDB service between the SUT and the LT

Test Procedure:

1) IUT sends a 'SDBProgramSelectIndication' message.

2) IUT sends again a 'SDBProgramSelectIndication' message. (Optional)

3) LT sends a 'SDBProgramSelectResponse' message.



..... : The message may be sent only when the timer tMsg expires.

Test Verdict:

Pass the test if the Client' response to the new broadcast program is considered suitably by the IUT depending on the response code of a 'SDBProgramSelectResponse' message set to the value such as 'rspOk', 'rspNoSession' or 'rspFormatError'.

5.2.1.4 DSM-CC User-to-Network Pass-Thru (SRM)

Table 22 – DSM-CC User-to-Network Pass-Thru Test Cases for SRM Testing

Test Case #	Test Case Names	DSM-CC U-N Path-Thru Functions	Reference to ISO/IEC 13818-6
1	Passing a message payload through the Network	The Users communicate with themselves to pass a message payload through the Network.	12.4.1
2	Requesting the recipient's response to a message payload to be passed	The User passes a message payload to another User through the Network and requests the recipient's response to the message.	12.4.2

5.2.1.4.1 Test Case 1 - Passing a message payload through the Network

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'PassThruRequest' message, the message payload is passed to any other User.

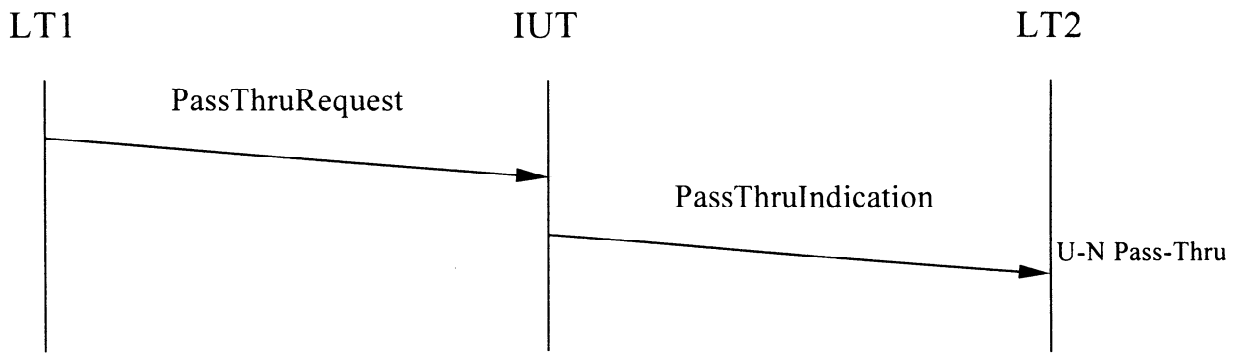
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT1 sends a 'PassThruRequest' message.

2) LT2 waits for a 'PassThruIndication' message.



Test Verdict:

Pass the test if a 'PassThruIndication' message with the message payload identical to the data of a 'PassThruRequest' message is received from the IUT when the IUT can accept the request.

5.2.1.4.2 Test Case 2 - Requesting the recipient's response to a message payload to be passed

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 send a 'PassThruReceiptRequest' message, the message payload is passed to any other User and the receiving User's response to the message is received.

Test Preamble:

Establish a network connection between the SUT and the LT

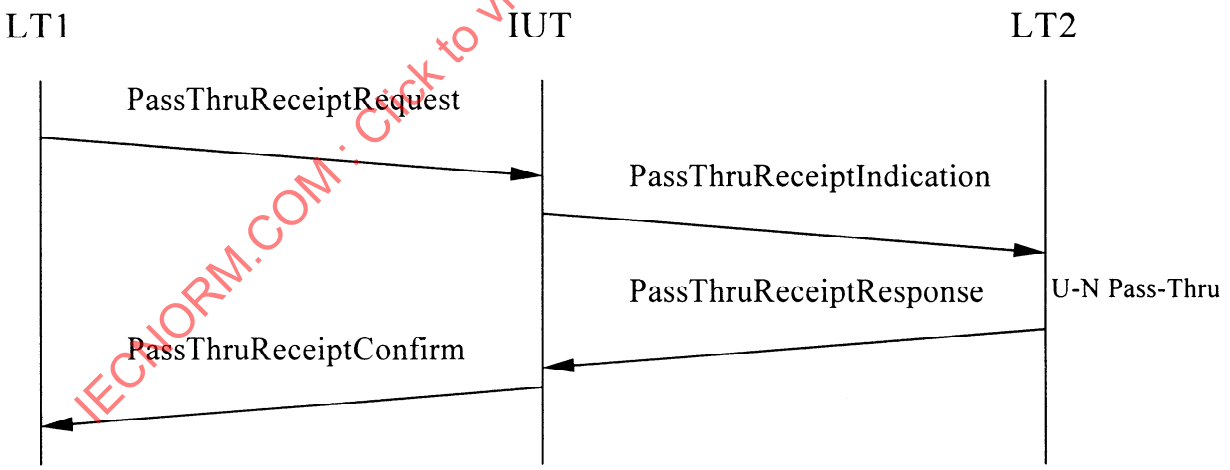
Test Procedure:

1) LT1 sends a 'PassThruReceiptRequest' message.

2) LT2 waits for a 'PassThruReceiptIndication' message.

3) LT2 sends a 'PassThruReceiptResponse' message.

4) LT1 waits for a 'PassThruReceiptConfirm' message.



Test Verdict:

Pass the test if a 'PassThruReceiptIndication' message with the message payload identical to the data of a 'PassThruReceiptRequest' message is received from the IUT when the IUT can accept the request and the timer tMsg is stopped in the IUT on receipt of a 'PassThruReceiptResponse' message, and a 'PassThruReceiptConfirm' message with the message payload identical to the data of a 'PassThruReceiptResponse' message is received from the IUT, or a 'PassThruReceiptConfirm' message with the response code set to indicate no response is received from the IUT when the timer tMsg expires before a 'PassThruReceiptResponse' message is received.

5.2.1.5 DSM-CC User-to-Network Pass-Thru (User)

Table 23 – DSM-CC User-to-Network Pass-Thru Test Cases for User Testing

Test Case No.	Test Case Names	DSM-CC U-N Path-Thru Functions	Reference to ISO/IEC 13818-6
1	Passing a message payload through the Network : a User initiates the request	A sending User creates a 'PassThruRequest' message to pass a message payload through the Network.	12.4.1
2	Passing a message payload through the Network : a User receives the indication	A receiving User processes the message payload if it is applicable.	12.4.1
3	Requesting the recipient's response to a message payload to be passed : a User initiates the request	A sending User creates a 'PassThruReceiptRequest' message to pass a message payload and request the recipient's response to the message through the Network.	12.4.2
4	Requesting the recipient's response to a message payload to be passed : a User receives the indication	A receiving User processes the message payload if it is applicable and sends a 'PassThruReceiptResponse' message to the Network.	12.4.2

5.2.1.5.1 Test Case 1 - Passing a message payload through the Network : a User initiates the request

Test Purpose:

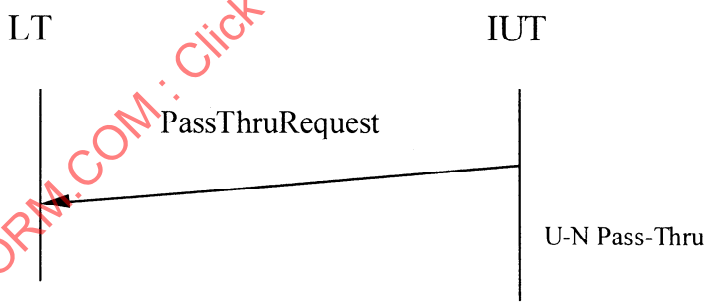
Verify that after establishing a network connection between the SUT and the LT, the IUT initiates to pass a message payload to any other User sending a 'PassThruRequest' message.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT waits for a 'PassThruRequest' message.



Test Verdict:

Pass the test if the data received from the IUT are valid.

5.2.1.5.2 Test Case 2 - Passing a message payload through the Network : a User receives the indication

Test Purpose:

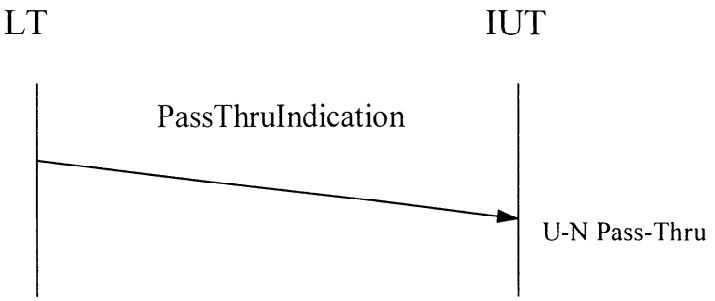
Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'PassThruIndication' message, the data of the message is processed.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT sends a 'PassThruIndication' message.



Test Verdict:

Pass the test if the message payload is processed by the IUT when the received message is applicable.

5.2.1.5.3 Test Case 3 - Requesting the recipient's response to a message payload to be passed : a User initiates the request

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, the IUT initiates to pass a message payload and to get the response to the message sending a 'PassThruReceiptRequest' message.

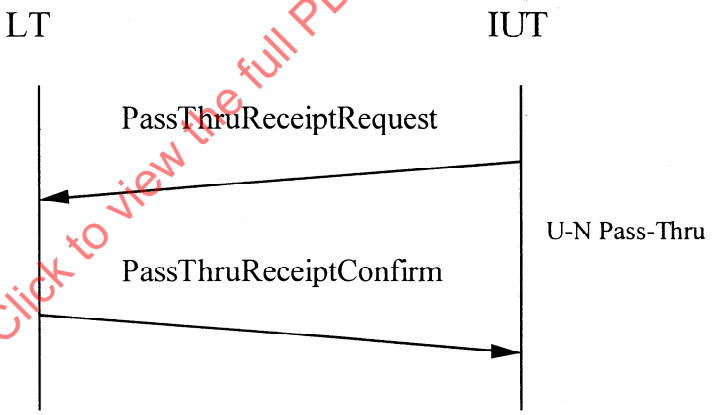
Test Preamble:

Establish a network connection between the SUT and the LT

Test Procedure:

1) IUT sends a 'PassThruReceiptRequest' message.

2) LT sends a 'PassThruReceiptConfirm' message.



Test Verdict:

Pass the test if the scenario is terminated and a received 'PassThruReceiptConfirm' message is discarded by the IUT when the timer tMsg expires or the timer tMsg is stopped in the IUT on receipt of a 'PassThruReceiptConfirm' message and the message is processed suitably by the IUT according to the value of the response code and PassThruData() of the received message.

5.2.1.5.4 Test Case 4 - Requesting the recipient's response to a message payload to be passed : a User receives the indication

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'PassThruReceiptIndication' message, the message payload is processed and a 'PassThruReceiptResponse' message is received.

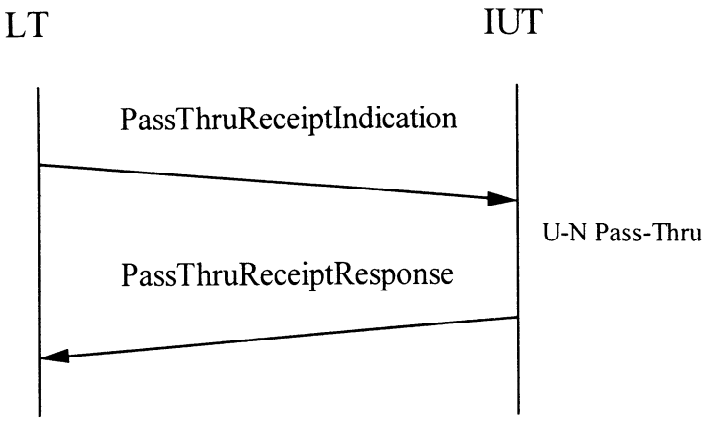
Test Preamble:

Establish a network connection between the SUT and the LT

Test Procedure:

1) LT sends a 'PassThruReceiptIndication' message.

2) LT waits for a 'PassThruReceiptResponse' message.



Test Verdict:

Pass the test if the message payload is processed by the IUT when the received message is applicable, and a 'PassThruReceiptResponse' message with the response code set to indicate the result of the message handled by the IUT is received from the IUT.

5.2.1.6 DSM-CC U-N Session (SRM)

Table 24 – DSM-CC User-to-Network Session Test Cases for SRM Testing

Test Case No.	Test Case Names	DSM-CC U-N Session Scenarios	Reference to ISO/IEC 13818-6
1	Setting up a new Session : Client initiates	Client Session Set-Up Command Sequence	4.8.1
2	Setting up a new Session : Network rejects	Client Session Set-Up Command Sequence	4.8.1
3	Setting up a new Session : Server rejects	Client Session Set-Up Command Sequence	4.8.1
4	Setting up a new Session : Client has Final UserData()	Client Session Set-Up Command Sequence	4.8.1
5	Setting up a new Session : Client initiates early release	Client Session Set-Up Command Sequence	4.8.1
6	Setting up a new Session : Server does not respond to a 'ServerSessionSetUpIndication' message	Client Session Set-Up Command Sequence	4.8.1
7	Setting up a new Session : Network rejects Server's Resource Allocation	Client Session Set-Up Command Sequence	4.8.1
8	Releasing a Session : Client initiates	Client Session Release Command Sequence	4.8.2
9	Releasing a Session : Network rejects	Client Session Release Command Sequence	4.8.2
10	Releasing a Session : Server rejects	Client Session Release Command Sequence	4.8.2
11	Asking a Session Status	Client Initiated Status Command Sequence	4.8.3

12	Setting up a Continuous Feed Session (CFS)	Server Continuous Feed Session Set-Up Command Sequence	4.9.1
13	Adding additional resources to an existing Session	Server Add Resource Command Sequence	4.9.2
14	Deleting resources from an existing Session	Server Session Delete Resource Command Sequence	4.9.3
15	Releasing a Session : Server initiates	Server Session Release Command Sequence	4.9.4
16	Releasing a Session : Client rejects	Server Session Release Command Sequence	4.9.4
17	Releasing a Continuous Feed Session (CFS) : Server initiates/Network rejects	Server Continuous Feed Session Release Command Sequence	4.9.5
18	Releasing a Continuous Feed Session (CFS) : Client rejects	Server Continuous Feed Session Release Command Sequence	4.9.5
19	Asking a Session status : Server initiates	Server Status Command Sequence	4.9.6
20	Forwarding a new Session : Server A requests	Server Session Forward Command Sequence	4.9.7
21	Forwarding a new Session : Network rejects	Server Session Forward Command Sequence	4.9.7
22	Transferring a Session	Server Session Transfer Command Sequence	4.9.8
23	Transferring a Session : Network rejects	Server Session Transfer Command Sequence	4.9.8
24	Transferring a Session : Server B rejects	Server Session Transfer Command Sequence	4.9.8
25	Transferring a Session : Server B is unable to allocate resources for transfer	Server Session Transfer Command Sequence	4.9.8
26	Transferring a Session : Client rejects	Server Session Transfer Command Sequence	4.9.8
27	Releasing a transferred Session : Client initiates	Transferred Session Release	4.9.9
28	Releasing a transferred Session : Server initiates	Transferred Session Release	4.9.9
29	Releasing a Session : Network initiates	Network Initiated Session Release Command Sequence	4.10.1
30	Releasing a Continuous Feed Session : Network initiates	Network Initiated Continuous Feed Session Release Command Sequence	4.10.2
31	Asking a Client Status : Network initiates	Network Initiated Client Status Command Sequence	4.10.3
32	Asking a Server Status : Network initiates	Network Initiated Server Status Command Sequence	4.10.4
33	Resetting for system recovery : Client initiates	Client Initiated Reset Command Sequence	4.11.1
34	Resetting for system recovery : Server initiates	Server Initiated Reset Command Sequence	4.11.2

35	Resetting for system recovery : Network initiates reset to a Server	Network Initiated Reset Command Sequence	4.11.3
36	Resetting for system recovery : Network initiates reset to a Client	Network Initiated Reset Command Sequence	4.11.3

5.2.1.6.1 Test Case 1 - Setting up a new Session : Client initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientSessionSetUpRequest' message, a new Session is set up with a Server.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

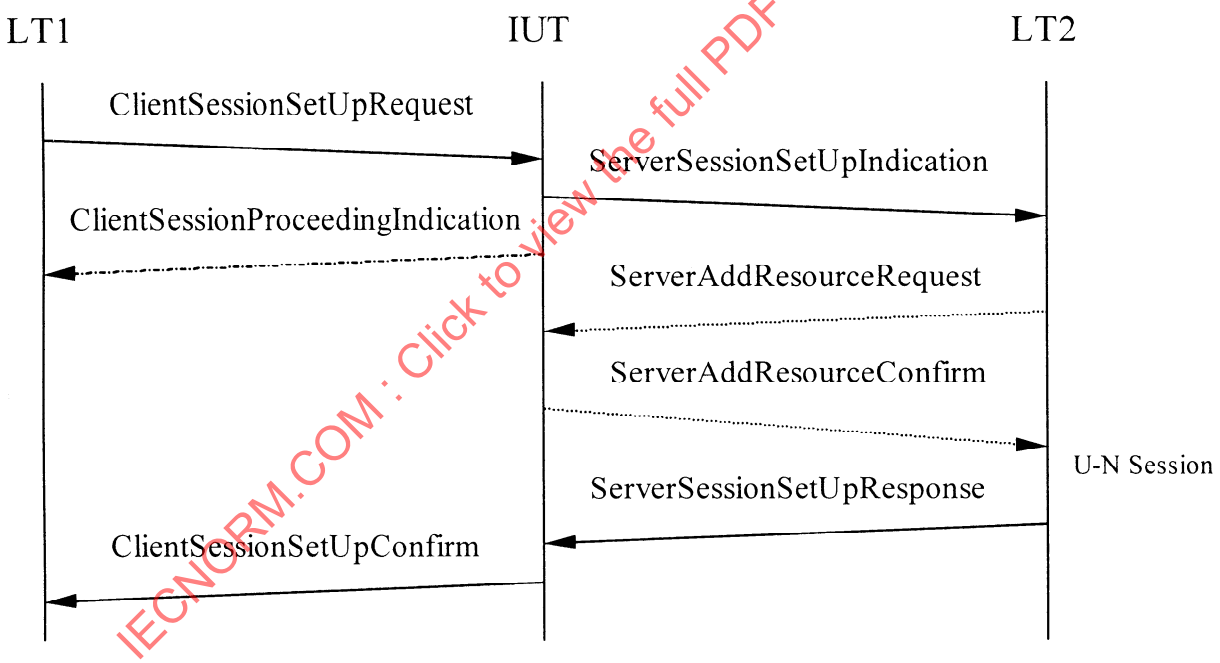
3) LT1 waits for a 'ClientSessionProceedingIndication' message. (Optional)

4) LT2 sends a 'ServerAddResourceRequest' message. (Optional)

5) LT2 waits for a 'ServerAddResourceConfirm' message. (Optional)

6) LT2 sends a 'ServerSessionSetUpResponse' message with the response code set to 'rspOk'.

7) LT1 waits for a 'ClientSessionSetUpConfirm' message.



----------: The message may be sent zero or more times.

..........: These messages may be sent zero or only once.

Test Verdict:

Pass the test if the timer tMsg is terminated in the IUT on receipt of a 'ServerSessionSetUpResponse' message, and a 'ClientSessionSetUpConfirm' message with the transactionId set to the value of the transactionId of a 'ClientSessionSetUpRequest' message and the sessionId set to the value of the sessionId of a 'ServerSessionSetUpResponse' message and the response code set to 'rspOk' is received from the IUT.

5.2.1.6.2 Test Case 2 - Setting up a new Session : Network rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientSessionSetUpRequest message with the invalid clientId or serverId, or the Network cannot support a new session, a new Session request is rejected.

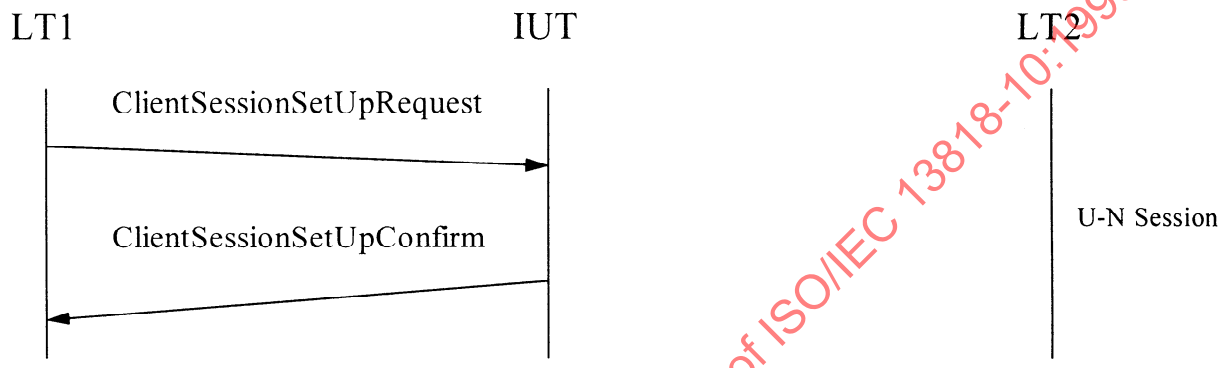
Test Preamble:

Set up a network connection between the SUT and the LT.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT1 waits for a 'ClientSessionSetUpConfirm' message.



Test Verdict:

Pass the test if a 'ClientSessionSetUpConfirm' message with the transactionId set to the value of the transactionId of a 'ClientSessionSetUpRequest' message and the response code set to indicate the reason of the session request rejected such as 'RspNoCalls', 'RspInvalidClientId', or 'RspInvalidServerId' is received from the IUT when the clientId or serverId received from the LT is invalid or the IUT cannot support a new Session.

5.2.1.6.3 Test Case 3 - Setting up a new Session : Server rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT2 rejects a session request, the session establishment request with a Server is terminated.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

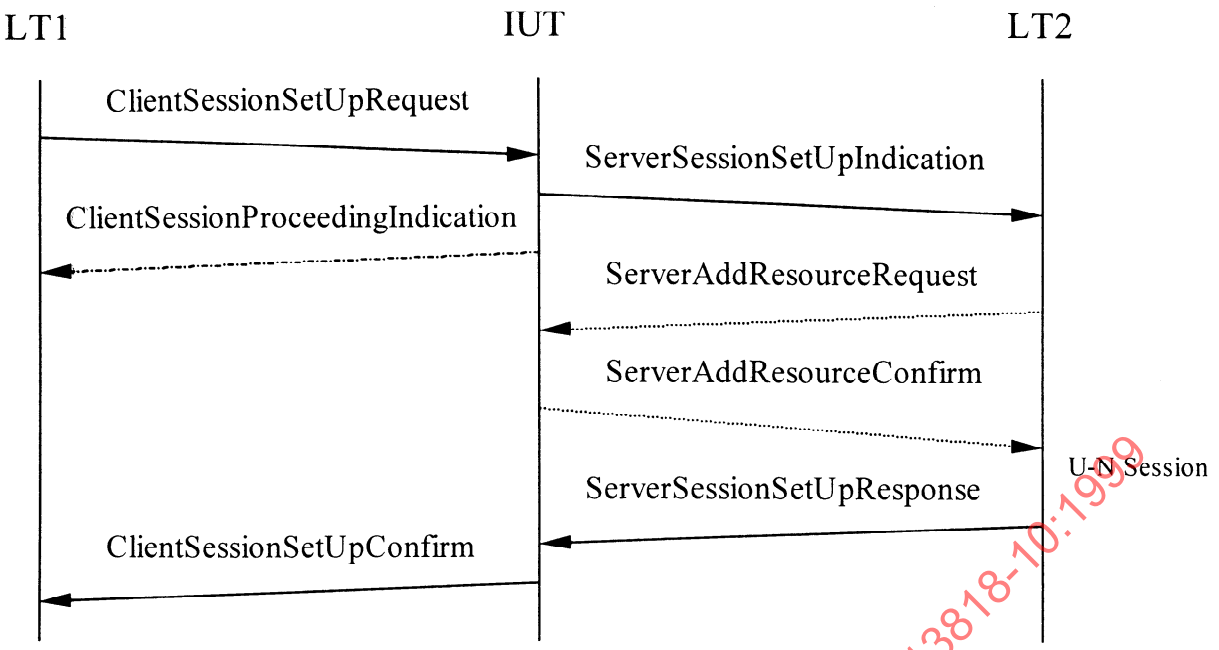
3) LT1 waits for a 'ClientSessionProceedingIndication' message. (Optional)

4) LT2 sends a 'ServerAddResourceRequest' message. (Optional)

5) LT2 waits for a 'ServerAddResourceConfirm' message. (Optional)

6) LT2 sends a 'ServerSessionSetUpResponse' message with the response code set to indicate the reason of the session establishment request rejected such as 'RspNoCalls', 'RspInvalidClientId', or 'RspServiceUnavailable'.

7) LT1 waits for a 'ClientSessionSetUpConfirm' message.



-·-·-·-: The message may be sent zero or more times.

·····: These messages may be sent zero or only once.

Test Verdict:

Pass the test if the session establishment with a Server is terminated by the IUT on receipt of a 'ServerSessionSetUpResponse' message, and a 'ClientSessionSetUpConfirm' message with the transactionId set to the value of the transactionId of a 'ClientSessionSetUpRequest' message and the response code set to the value of the response code of a 'ServerSessionSetUpResponse' message is received from the IUT.

5.2.1.6.4 Test Case 4 - Setting up a new Session : Client has Final UserData()

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientConnectRequest' message, the UserData() is sent to a Server.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

3) LT1 waits for a 'ClientSessionProceedingIndication' message. (Optional)

4) LT2 sends a 'ServerAddResourceRequest' message. (Optional)

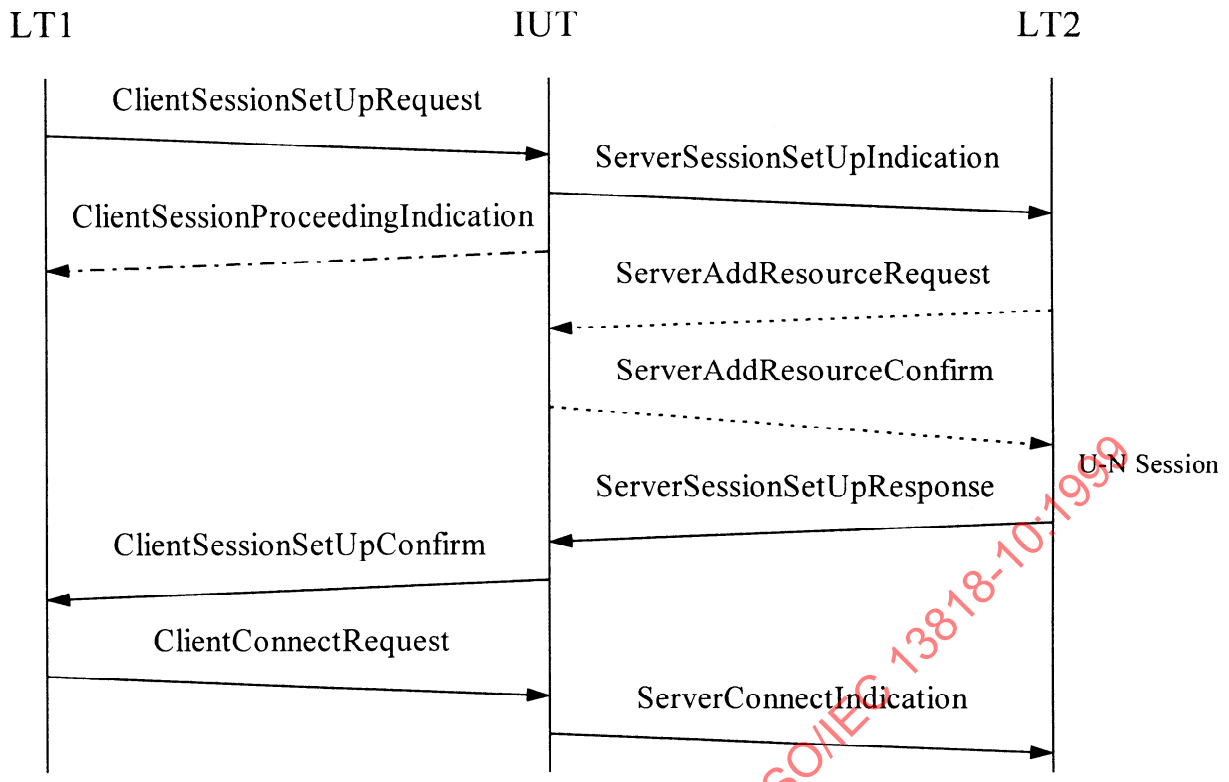
5) LT2 waits for a 'ServerAddResourceConfirm' message. (Optional)

6) LT2 sends a 'ServerSessionSetUpResponse' message.

7) LT1 waits for a 'ClientSessionSetUpConfirm' message.

8) LT1 sends a 'ClientConnectRequest' message.

9) LT2 waits for a 'ServerConnectIndication' message.



----------: The message may be sent zero or more times.

.....: The message may be sent zero or only once.

Test Verdict:

Pass the test if a 'ServerConnectIndication' message with the sessionId and UserData() set to the corresponding values of a 'ClientConnectRequest' message is received from the IUT.

5.2.1.6.5 Test Case 5 - Setting up a new Session : Client initiates Early Release

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientReleaseRequest' message prior to the receipt of the first confirm message, all resources allocated for the Session are released.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message. (Optional)

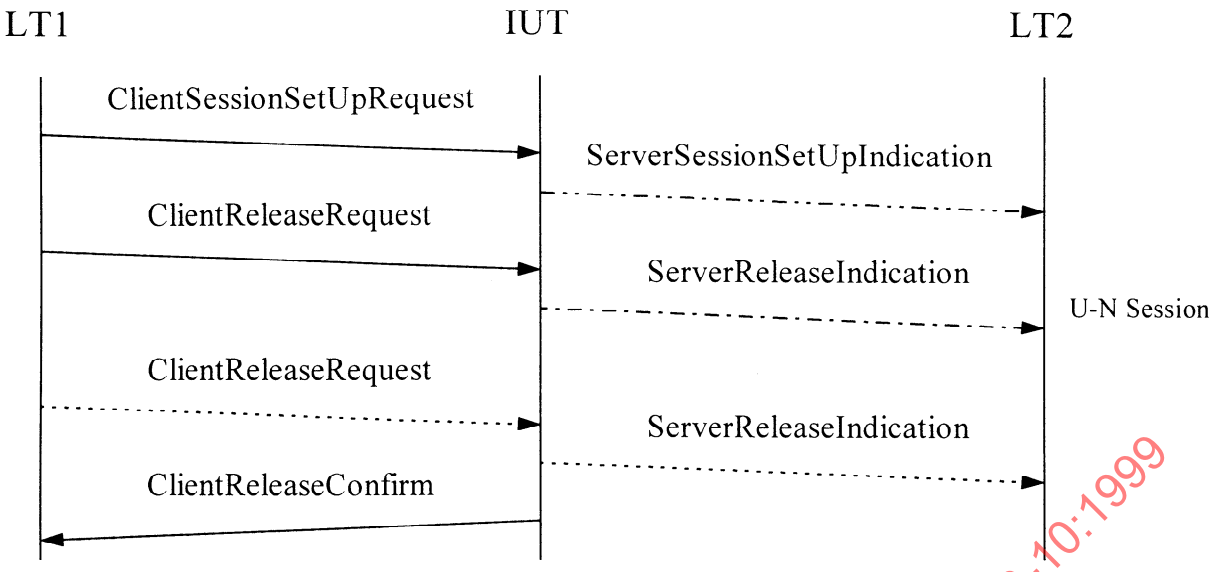
3) LT1 sends a 'ClientReleaseRequest' message.

4) LT2 waits for a 'ServerReleaseIndication' message. (Optional)

5) LT1 sends again a 'ClientReleaseRequest' message after the timer tMsg expires. (Optional)

6) LT2 waits for a 'ServerReleaseIndication' message. (Optional)

7) LT1 waits for a 'ClientReleaseConfirm' message.



·····: The message may be sent.

·····: The message may be sent only when a 'ServerSessionSetUpIndication' message has been sent.

·····: These messages may be sent only when the timer tMsg expires.

Test Verdict:

Pass the test if all resources allocated to the Session are released by the IUT on receipt of a 'ClientReleaseRequest' message and a 'ServerReleaseIndication' messages with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message is received from the IUT when the value of the sessionId of a 'ClientReleaseRequest' message corresponds to an existing Session and the session establishment to a Server has already begun, and then a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message is received from the IUT, or

all resources allocated to the Session are released by the IUT on receipt of a 'ClientReleaseRequest' message and a 'ServerReleaseIndication' messages with the sessionId set to the value of the sessionId of a 'ServerSessionSetUpIndication' message is received from the IUT when the value of the sessionId of a 'ClientReleaseRequest' message is "0'' and the value of the transactionId of a 'ClientReleaseRequest' message corresponds to the value of the sessionId of a 'ClientSessionSetUpRequest' message and the session establishment to a Server has already begun, and then a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message is received from the IUT.

5.2.1.6.6 Test Case 6 - Setting up a new Session : Server does not respond to a 'ServerSessionSetUpIndication' message

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT2 does not respond before the timer tMsg expires, a Session is terminated and a 'ClientSessionSetUpConfirm' message with the response code set to 'RspSeNoResponse' is received.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

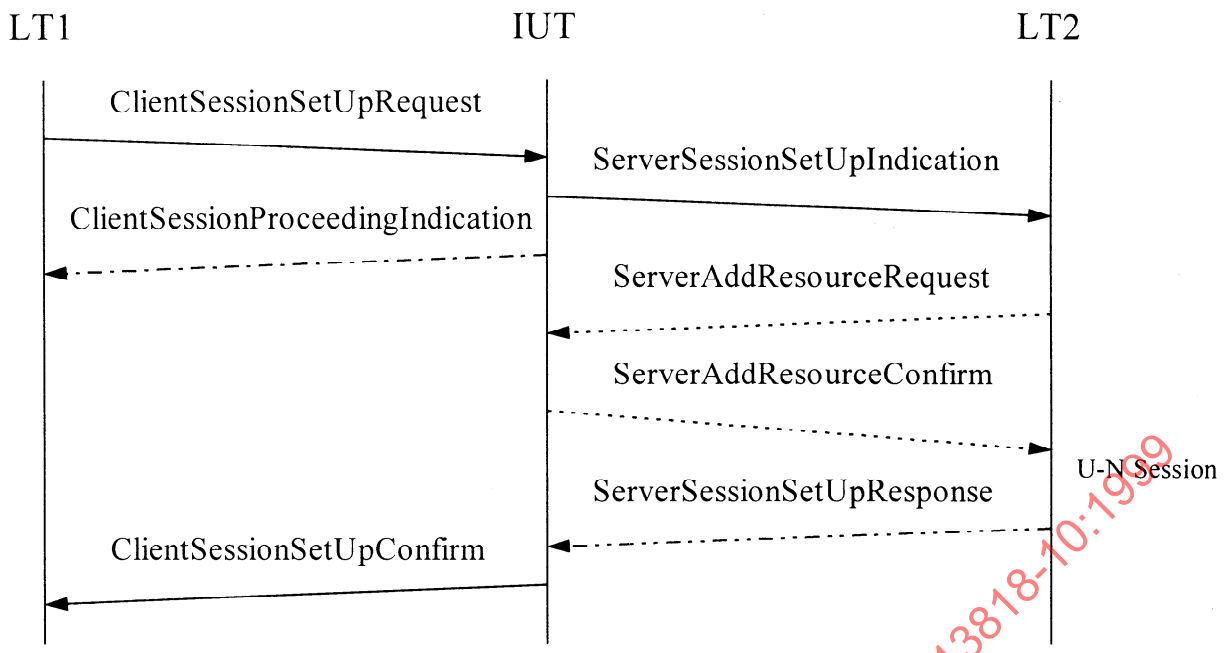
3) LT1 waits for a 'ClientSessionProceedingIndication' message. (Optional)

4) LT2 sends a 'ServerAddResourceRequest' message. (Optional)

5) LT2 waits for a 'ServerAddResourceConfirm' message. (Optional)

6) LT2 sends a 'ServerSessionSetUpResponse' message after the timer tMsg expires. (Optional)

7) LT1 waits for a 'ClientSessionSetUpConfirm' message.



----------: The message may be sent zero or more times.

..........: These messages may be sent zero or only once.

----------: The message may be sent only after the timer tMsg expires.

Test Verdict:

Pass the test if the Session is terminated by the IUT on the timer tMsg expires and a 'ClientSessionSetUpConfirm' massage with the response code set to 'RspSeNoResponse' is received from the IUT.

5.2.1.6.7 Test Case 7 - Setting up a new Session : Network rejects Server's Resource Allocation

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the IUT is unable to accept any resource allocated by the Server, a Session is terminated and a 'ClientSessionSetUpConfirm' message with response code set to 'RspNeResourceFailed' is received, and a Session release procedure is initiated.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

3) LT1 waits for a 'ClientSessionProceedingIndication' message. (Optional)

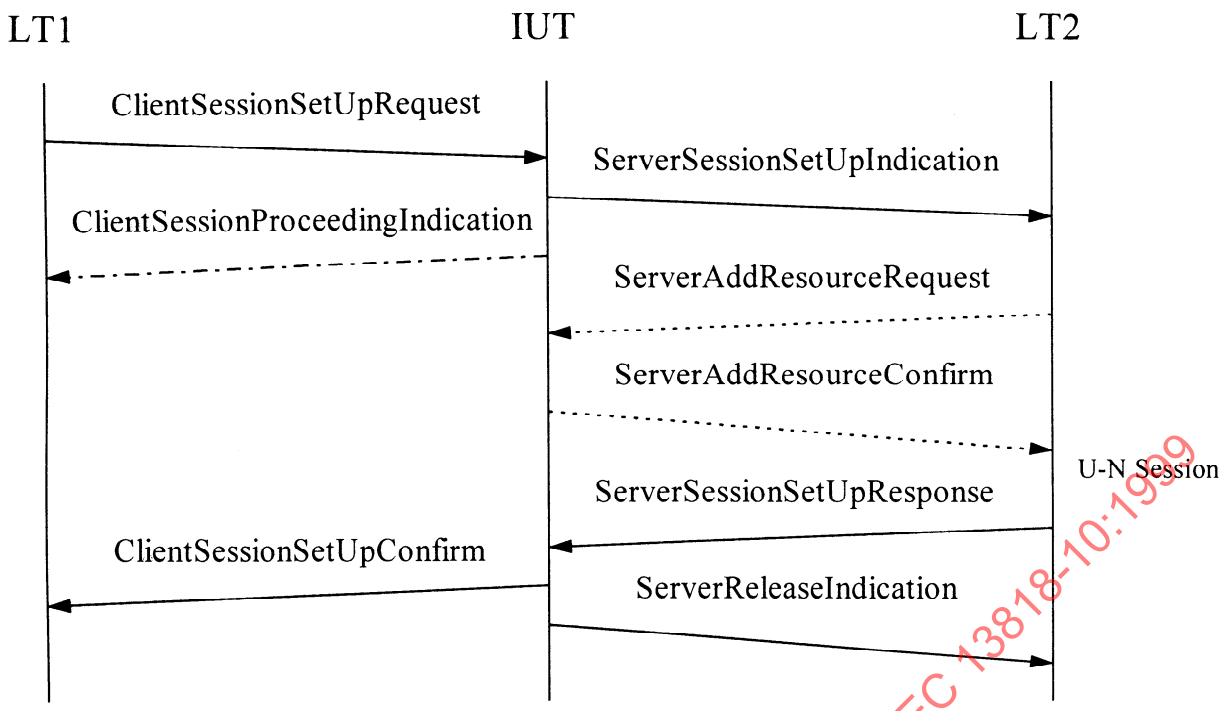
4) LT2 sends a 'ServerAddResourceRequest' message. (Optional)

5) LT2 waits for a 'ServerAddResourceConfirm' message. (Optional)

6) LT2 sends a 'ServerSessionSetUpResponse' message.

7) LT1 waits for a 'ClientSessionSetUpConfirm' message.

8) LT2 waits for a 'ServerReleaseIndication' message.



: The message may be sent zero or more times.

: These messages may be sent zero or only once.

Test Verdict:

Pass the test if the Session is terminated by the IUT when the IUT can not accept any of resources for a Session allocated by a Server, and then a 'ClientSessionSetUpConfirm' message with the response code set to 'RspNeResourceFailed' and a 'ServerReleaseIndication' message are received from the IUT.

5.2.1.6.8 Test Case 8 - Releasing a Session : Client initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientReleaseRequest' message, all resources allocated to the Session are released.

Test Preamble:

Establish a Session between the SUT and the LTs.

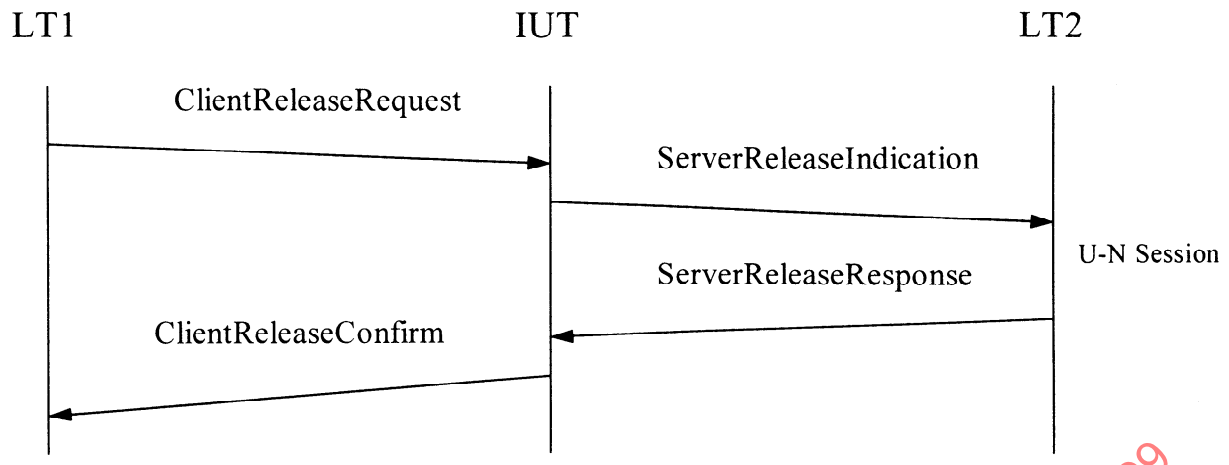
Test Procedure:

1) LT1 sends a 'ClientReleaseRequest' message.

2) LT2 waits for a 'ServerReleaseIndication' message.

3) LT2 sends a 'ServerReleaseResponse' message.

4) LT1 waits for a 'ClientReleaseConfirm' message.



Test Verdict:

Pass the test if a 'ServerReleaseIndication' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message and the reason code set to indicate the session released by the Client is received from the IUT and all resources are released by the IUT on receipt of a 'ServerReleaseResponse' message, and then a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message is received from the IUT.

5.2.1.6.9 Test Case 9 - Releasing a Session : Network rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientReleaseRequest' message with an invalid sessionId or a sessionId not owned by the User, the release request is rejected.

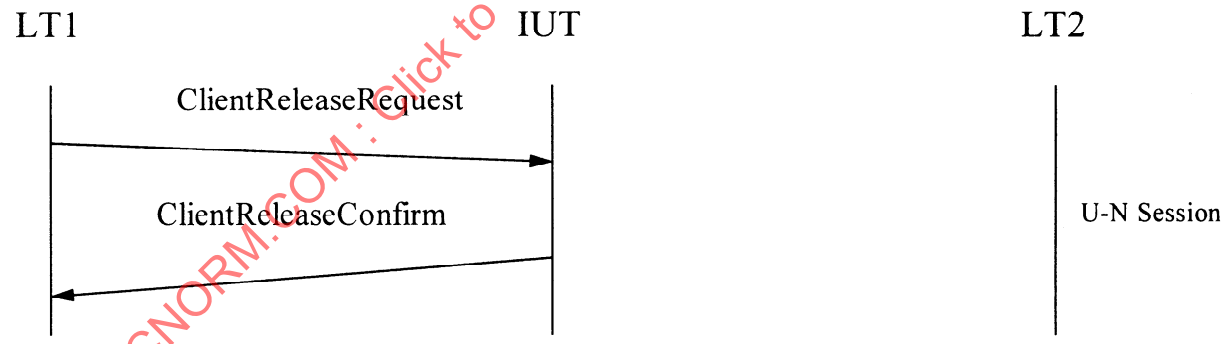
Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientReleaseRequest' message.

2) LT1 waits for a 'ClientReleaseConfirm' message.



Test Verdict:

Pass the test if a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message and the response code set to indicate the reason of the release request rejected such as 'RspNeNoSession', or 'RspNeNotOwner' is received from the IUT.

5.2.1.6.10 Test Case 10 - Releasing a Session : Server rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT2 rejects a release indication, the release procedure is terminated.

Test Preamble:

Establish a Session between the SUT and the LTs.

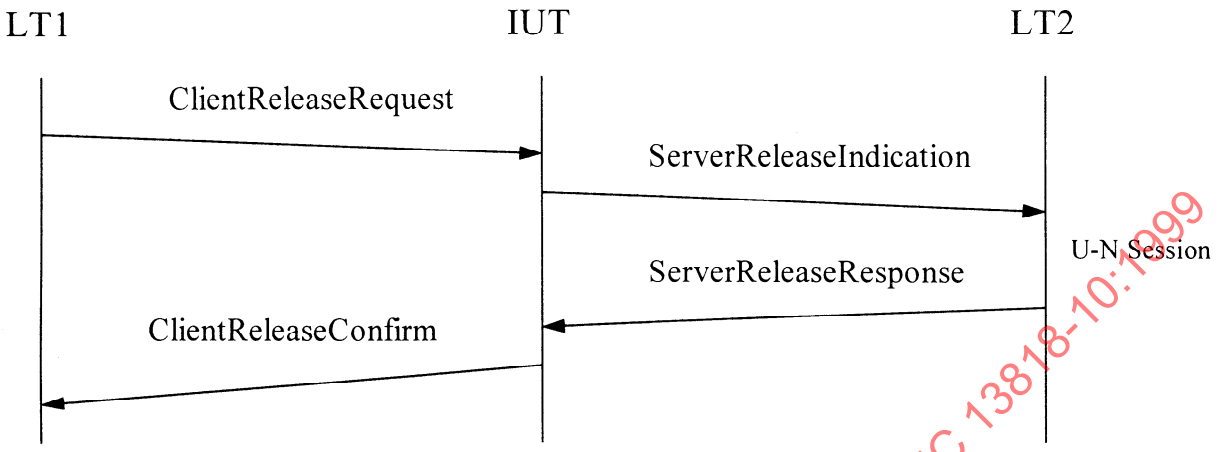
Test Procedure:

1) LT1 sends a 'ClientReleaseRequest' message.

2) LT2 waits for a 'ServerReleaseIndication' message.

3) LT2 sends a 'ServerReleaseResponse' message with the response code set to 'RspInvalidSessionId'.

4) LT1 waits for a 'ClientReleaseConfirm' message.



Test Verdict:

Pass the test if all resources assigned to the Session are released by the IUT on receipt of a 'ServerReleaseResponse' message, and then a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ServerReleaseResponse' message and the response code set to 'rspOk' is received from the IUT.

5.2.1.6.11 Test Case 11 - Asking a Session Status

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientStatusRequest' message, the information of the requested type is obtained.

Test Preamble:

Establish a network connection between the SUT and the LTs.

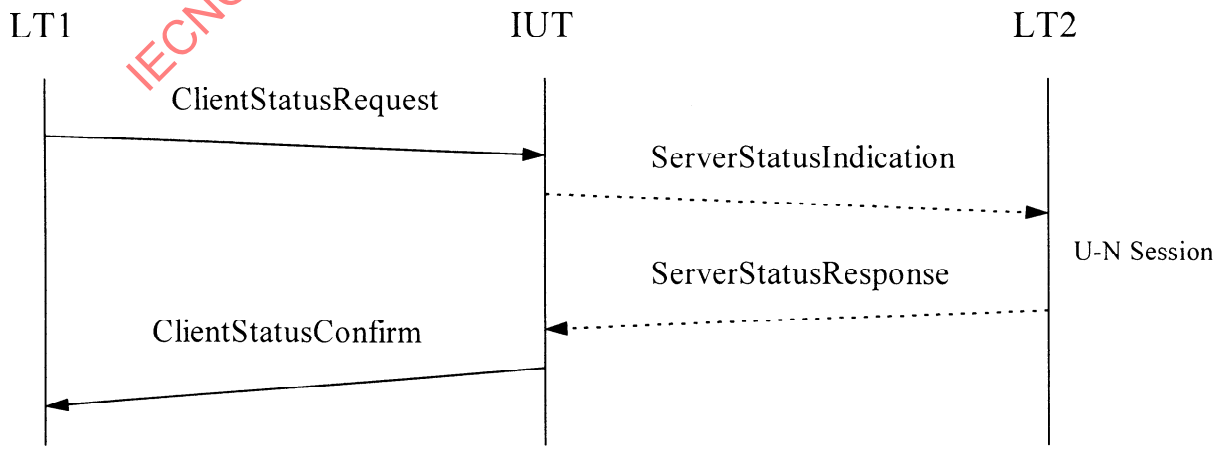
Test Procedure:

1) LT1 sends a 'ClientStatusRequest' message.

2) LT2 waits for a 'ServerStatusIndication' message. (Optional)

3) LT2 sends a 'ServerStatusResponse' message. (Optional)

4) LT1 waits for a 'ClientStatusConfirm' message.



..... These messages may be sent only when the IUT wants to get an answer of the status request from a Server.

Test Verdict:

Pass the test if the requested status information is obtained from the IUT through a 'ClientStatusConfirm' message.

5.2.1.6.12 Test Case 12 - Setting up a new Continuous Feed Session (CFS)

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT2 sends a 'ServerContinuousFeedSessionRequest' message, a Continuous Feed Session is established.

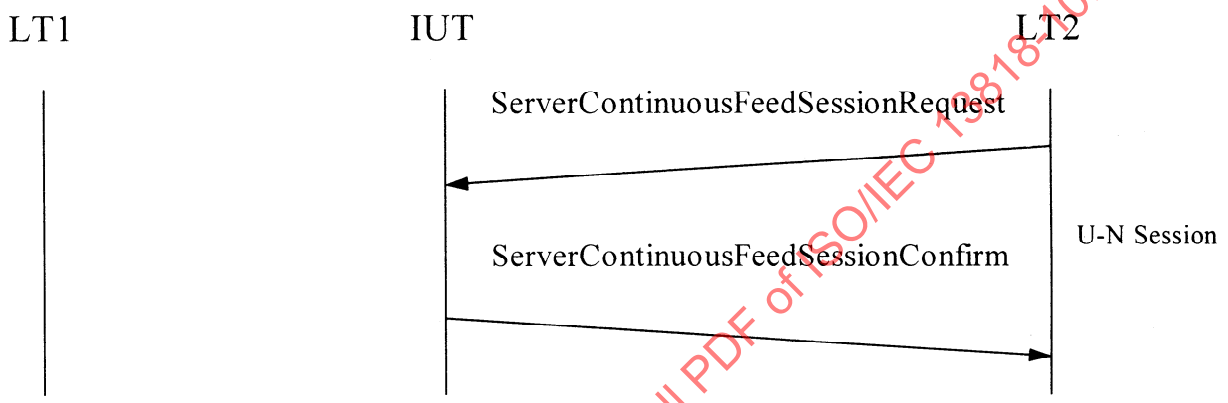
Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT2 sends a 'ServerContinuousFeedSessionRequest' message.

2) LT2 waits for a 'ServerContinuousFeedSessionConfirm' message.



Test Verdict:

Pass the test if a 'ServerContinuousFeedSessionConfirm' message with the response code set to indicate the session status is received from the IUT.

5.2.1.6.13 Test Case 13 - Adding additional resources to an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT2 sends a 'ServerAddResourceRequest' message, additional resources are added to the session.

Test Preamble:

Establish a Session between the SUT and the LTs.

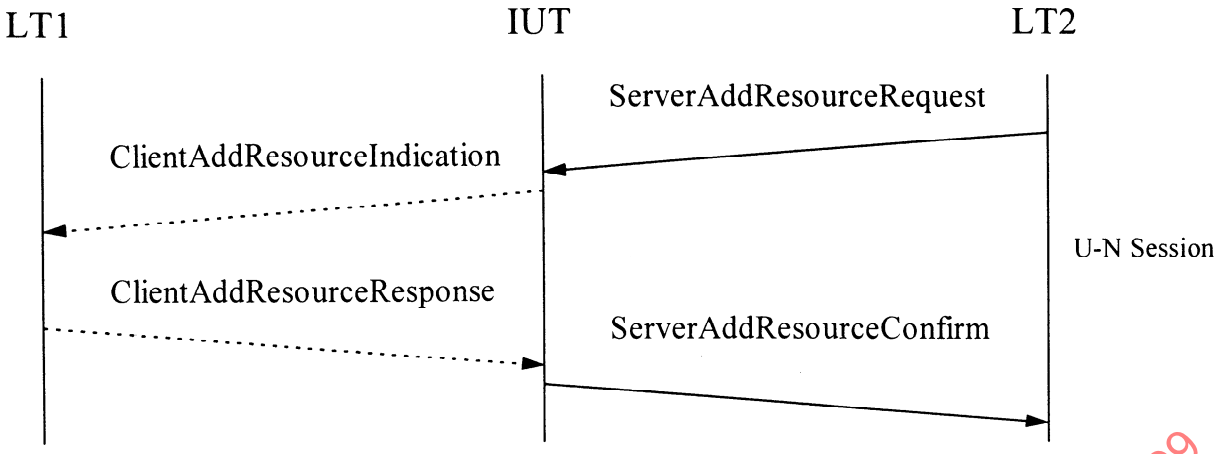
Test Procedure:

1) LT2 sends a 'ServerAddResourceRequest' message.

2) LT1 waits for a 'ClientAddResourceIndication' message. (Optional)

3) LT1 sends a 'ClientAddResourceResponse' message. (Optional)

4) LT2 waits for a 'ServerAddResourceConfirm' message.



.....: These messages may be sent only when the IUT can accept the add resource request.

Test Verdict:

Pass the test if a 'ServerAddResourceConfirm' message with the response code set to indicate the request failed is received from the IUT when the IUT can not accept the request, or a 'ClientAddResourceIndication' message is received from the IUT when the IUT can accept the request, and then a 'ServerAddResourceConfirm' message with the response code set to the value of the response code of a 'ClientAddResourceResponse' message is received from the IUT.

5.2.1.6.14 Test Case 14 - Deleting resources from an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT2 sends a 'ServerDeleteResourceRequest' message, all resources are deleted from an existing session.

Test Preamble:

Establish a Session between the SUT and the LTs.

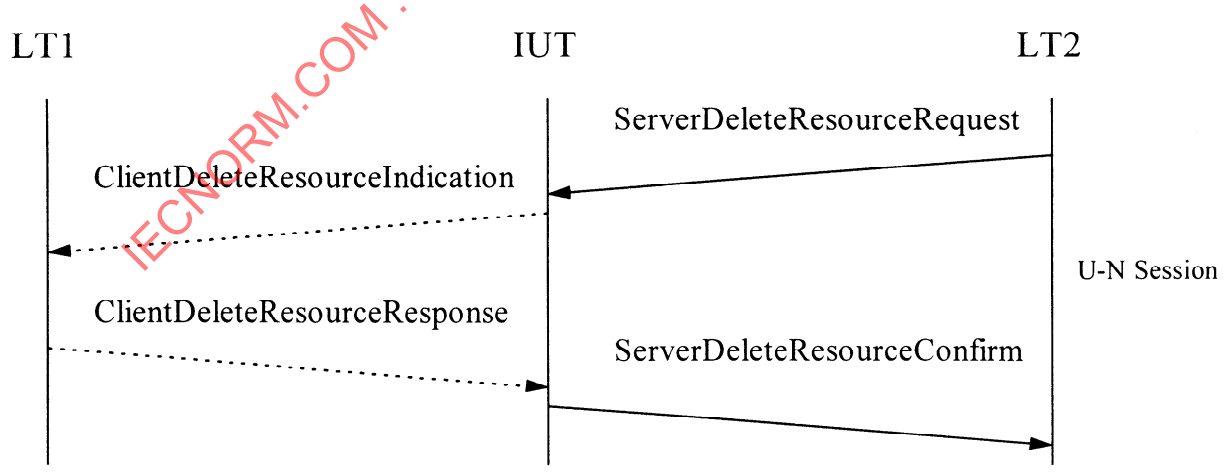
Test Procedure:

1) LT2 sends a 'ServerDeleteResourceRequest' message.

2) LT1 waits for a 'ClientDeleteResourceIndication' message. (Optional)

3) LT1 sends a 'ClientDeleteResourceResponse' message. (Optional)

4) LT2 waits for a 'ServerDeleteResourceConfirm' message.



.....: These messages may be sent only when the IUT can accept the delete resource request.

Test Verdict:

Pass the test if a 'ServerDeleteResourceConfirm' message with the response code set to indicate the reason of the request denied is received from the IUT when the IUT can not accept the request, or a 'ClientDeleteResourceIndication' message with the reason code set to indicate the resources delete request initiated by a Server is received from the IUT when the IUT can

accept the request, and then a 'ServerDeleteResourceConfirm' message with the response code set to the value of the response code of a 'ClientDeleteResourceResponse' message are received from the IUT.

5.2.1.6.15 Test Case 15 - Releasing a Session : Server initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT2 sends a 'ServerReleaseRequest' message, an existing session is released.

Test Preamble:

Establish a Session between the SUT and the LTs.

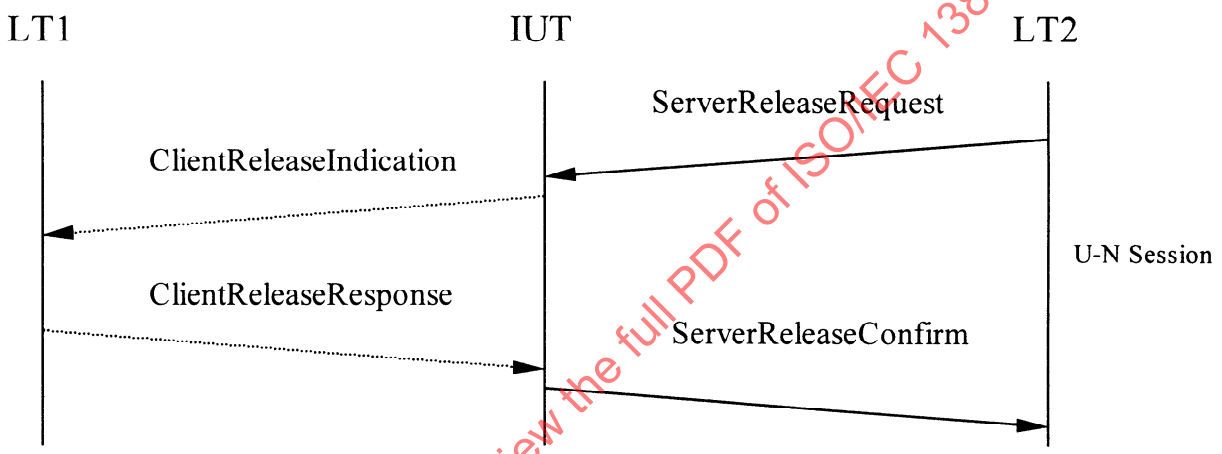
Test Procedure:

1) LT2 sends a 'ServerReleaseRequest' message.

2) LT1 waits for a 'ClientReleaseIndication' message. (Optional)

3) LT1 sends a 'ClientReleaseResponse' message. (Optional)

4) LT2 waits for a 'ServerReleaseConfirm' message.



·····: These messages may be sent only when the IUT can accept the release request.

Test Verdict:

Pass the test if a 'ServerReleaseConfirm' message with the response code set to indicate the reason of the request denied is received from the IUT when the IUT can not accept the request, or a 'ClientReleaseIndication' message with the reason code set to indicate the session release request initiated by a Server is received from the IUT when the IUT can accept the request, and all resources assigned to the Session are released by the IUT on receipt of a 'ClientReleaseResponse' message, and then a 'ServerReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseResponse' message is received from the IUT.

5.2.1.6.16 Test Case 16 - Releasing a Session : Client rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientReleaseResponse' message with the response code set to indicate an invalid sessionId, the release request is rejected.

Test Preamble:

Establish a Session between the SUT and the LTs.

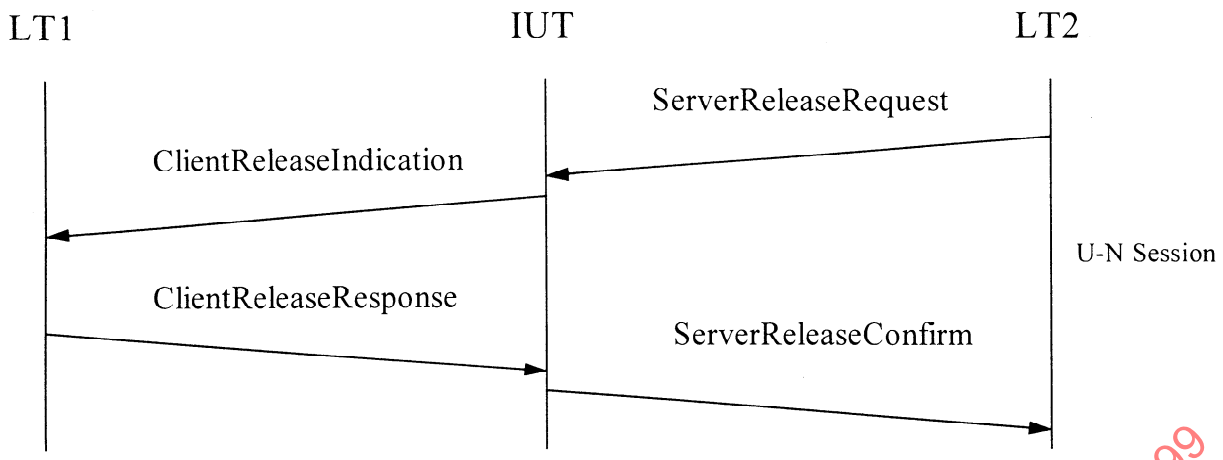
Test Procedure:

1) LT2 sends a 'ServerReleaseRequest' message.

2) LT1 waits for a 'ClientReleaseIndication' message.

3) LT1 sends a 'ClientReleaseResponse' message with the response code set to indicate an invalid sessionId.

4) LT2 waits for a 'ServerReleaseConfirm' message.



Test Verdict:

Pass the test if all resources assigned to the Session are released by the IUT on receipt of a 'ClientReleaseResponse' message, and then a 'ServerReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseResponse' message and the response code set to indicate the procedure error occurred with the Client is received from the IUT.

5.2.1.6.17 Test Case 17 - Releasing a Continuous Feed Session (CFS) : Server initiates/Network rejects

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT2, and a Session between the SUT and the LT1, when the LT2 sends a 'ServerReleaseRequest' message, a CFS is terminated.

Test Preamble:

Establish a Continuous Feed Session between the SUT and the LT2, and a Session between the SUT and the LT1.

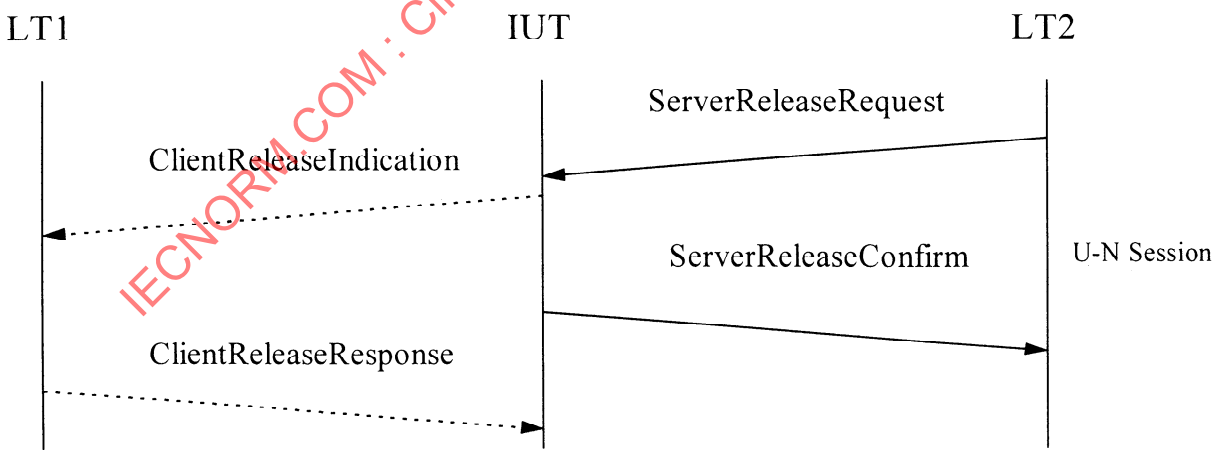
Test Procedure:

1) LT2 sends a 'ServerReleaseRequest' message.

2) LT1 waits for a 'ClientReleaseIndication' message. (Optional)

3) LT2 waits for a 'ServerReleaseConfirm' message.

4) LT1 sends a 'ClientReleaseResponse' message. (Optional)



.....: These messages may be sent only when the IUT can accept the release request.

Test Verdict:

Pass the test if a 'ServerReleaseConfirm' message with the response code set to indicate the reason of the request denied is received from the IUT when the IUT can not accept the request, or

a 'ClientReleaseIndication' message with the sessionId set to the value of the sessionId connected to the Continuous Feed Session and the reason code set to 'RsnClSessionRelease' and a 'ServerReleaseConfirm' message with the sessionId set to the

value of the sessionId of the Continuous Feed Session are received from the IUT when the IUT can accept the request, and then all Client interface resources assigned to the Continuous Feed Session are released by the IUT.

5.2.1.6.18 Test Case 18 - Releasing a Continuous Feed Session (CFS) : Client rejects

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT2, a Session between the SUT and the LT1, when the LT1 sends a 'ClientReleaseResponse' message with the response code set to indicate an invalid sessionId, all resources assigned to the session are released.

Test Preamble:

Establish a Continuous Feed Session between the SUT and the LT2, and a Session between the SUT and the LT1.

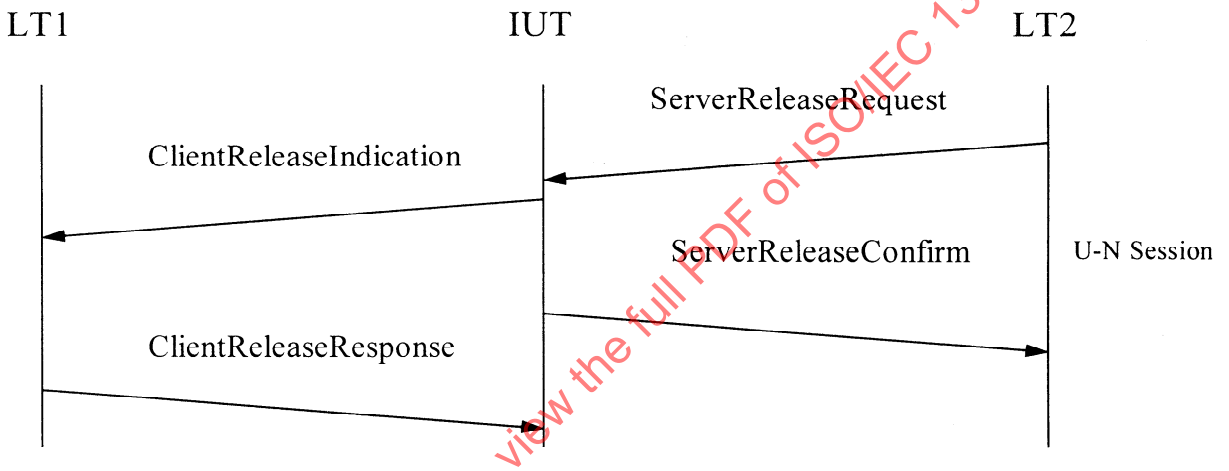
Test Procedure:

1) LT2 sends a 'ServerReleaseRequest' message.

2) LT1 waits for a 'ClientReleaseIndication' message.

3) LT2 waits for a 'ServerReleaseConfirm' message.

4) LT1 sends a 'ClientReleaseResponse' message with the response code set to 'RspClNoSession'.



Test Verdict:

Pass the test if all resources assigned to the session are released by the IUT on receipt of a 'ClientReleaseResponse' message.

5.2.1.6.19 Test Case 19 - Asking a Session status : Server initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT2 sends a 'ServerStatusRequest' message, the information of the requested type is obtained.

Test Preamble:

Establish a network connection between the SUT and the LTs.

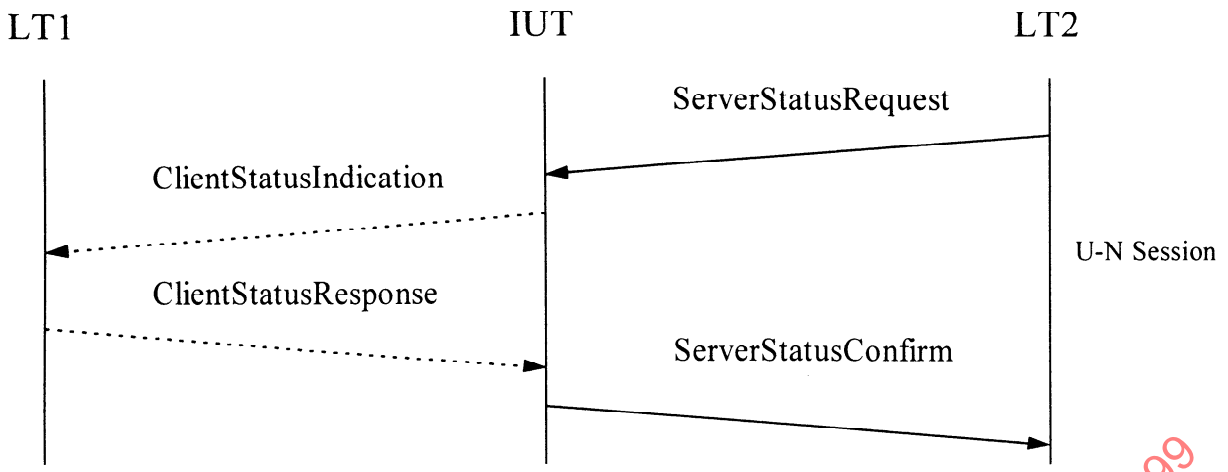
Test Procedure:

1) LT2 sends a 'ServerStatusRequest' message.

2) LT1 waits for a 'ClientStatusIndication' message. (Optional)

3) LT1 sends a 'ClientStatusResponse' message. (Optional)

4) LT2 waits for a 'ServerStatusConfirm' message.



.....: These messages may be sent only when the IUT wants to request a status from a Client.

Test Verdict:

Pass the test if the information of the requested type is obtained from the IUT through a 'ServerStatusConfirm' message.

5.2.1.6.20 Test Case 20 - Forwarding a new Session : Server A requests

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT2 sends a 'ServerSessionSetUpResponse' message with the response code set to 'rspForward' and the nextServerId set to the value of the userId of Server B, a new Session is forwarded to Server B.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientSessionSetUpRequest' message.

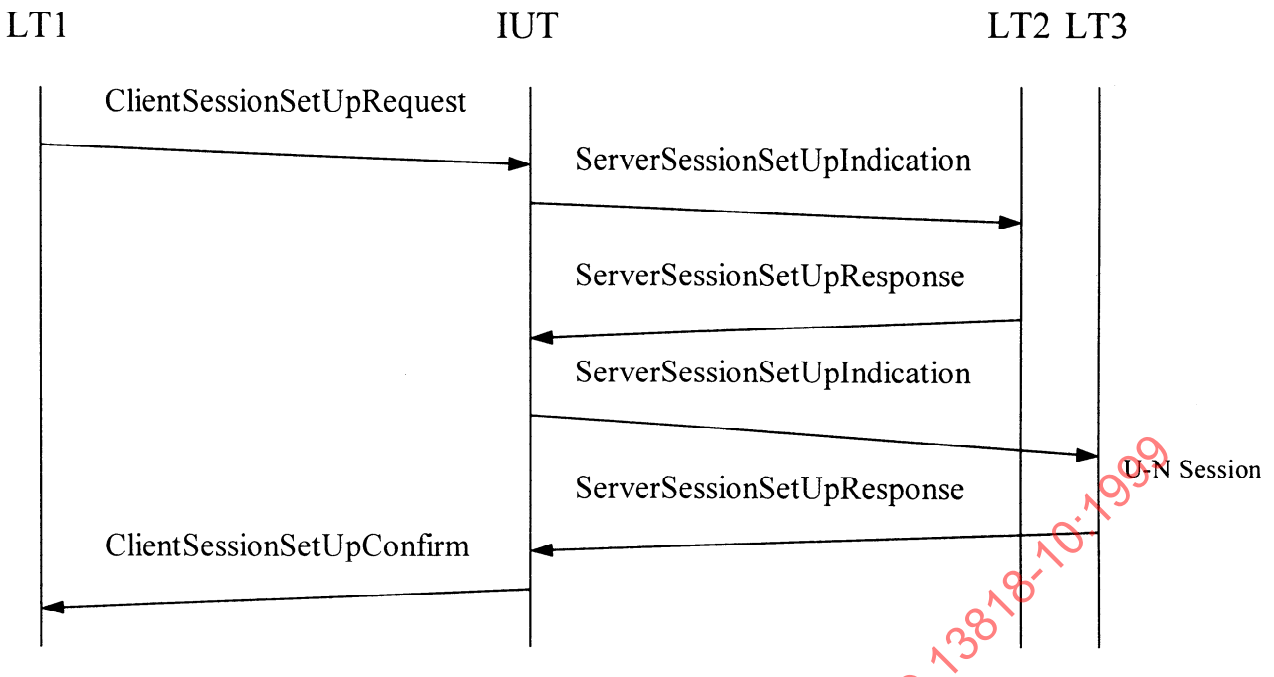
2) LT2 waits for a 'ServerSessionSetUpIndication' message.

3) LT2 sends a 'ServerSessionSetUpResponse' message with the response code set to 'rspForward' and the nextServerId set to the value of the userId of Server B.

4) LT3 waits for a 'ServerSessionSetUpIndication' message.

5) LT3 sends a 'ServerSessionSetUpResponse' message.

6) LT1 waits for a 'ClientSessionSetUpConfirm' message.



Test Verdict:

Pass the test if a 'ServerSessionSetUpIndication' messages with the forward count set to "0" and no forwardServerIds and a 'ServerSessionSetUpIndication' messages with the serverId set to Server B, the forwardCount set to "1" and the forwardServerId which contains the ID of Server A are received from the IUT, and then the Session set-up procedure as defined in the Client initiated Session set-up sequence is performed successfully by the IUT on receipt of a 'ServerSessionSetUpResponse' message.

5.2.1.6.21 Test Case 21 - Forwarding a new Session : Network rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the IUT can not process the forward requested in a 'ServerSessionSetUpResponse' message, a new Session is terminated.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

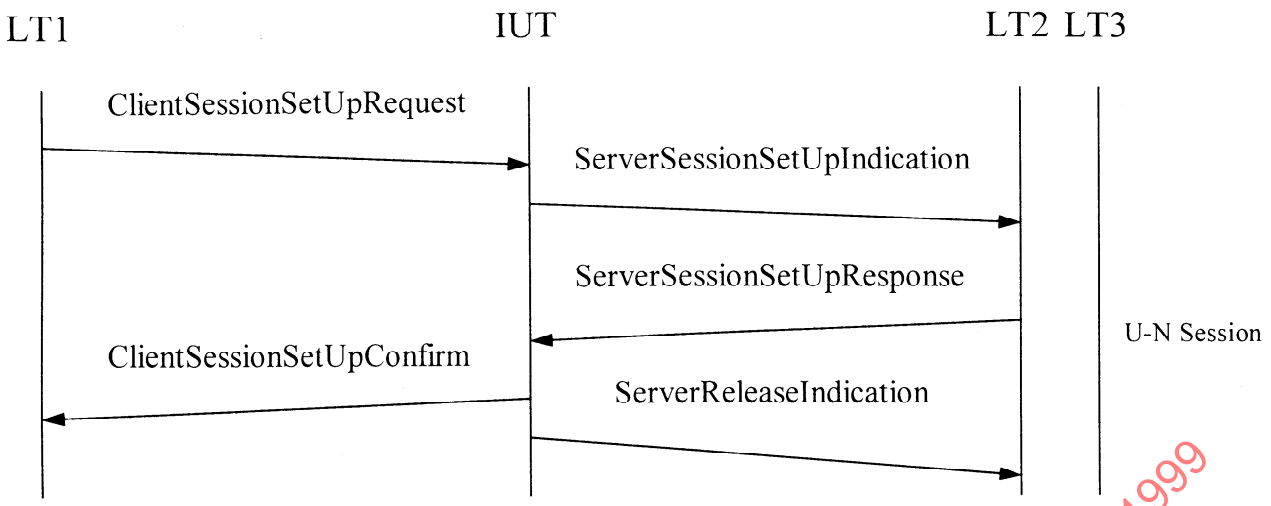
1) LT1 sends a 'ClientSessionSetUpRequest' message.

2) LT2 waits for a 'ServerSessionSetUpIndication' message.

3) LT2 sends a 'ServerSessionSetUpResponse' message with the response code set to 'rspForward' and the nextServerId set to the userId of Server B.

4) LT1 waits for a 'ClientSessionSetUpConfirm' message.

5) LT2 waits for a 'ServerReleaseIndication' message.



Test Verdict:

Pass the test if a 'ClientSessionSetUpConfirm' message with the response code set to 'rspNeForwardFailed' and a 'ServerReleaseIndication' message are received from the IUT when the IUT can not process the forward requested in a 'ServerSessionSetUpResponse' message.

5.2.1.6.22 Test Case 22 - Transferring a Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT2 sends a 'ServerSessionTransferRequest' message, a Session is transferred to another session gateway.

Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT2 sends a 'ServerSessionTransferRequest' message.

2) LT3 waits for a 'ServerSessionTransferIndication' message.

3) LT3 sends a 'ServerAddResourceRequest' message. (Optional)

4) LT3 waits for a 'ServerAddResourceConfirm' message. (Optional)

5) LT3 sends a 'ServerSessionTransferResponse' message.

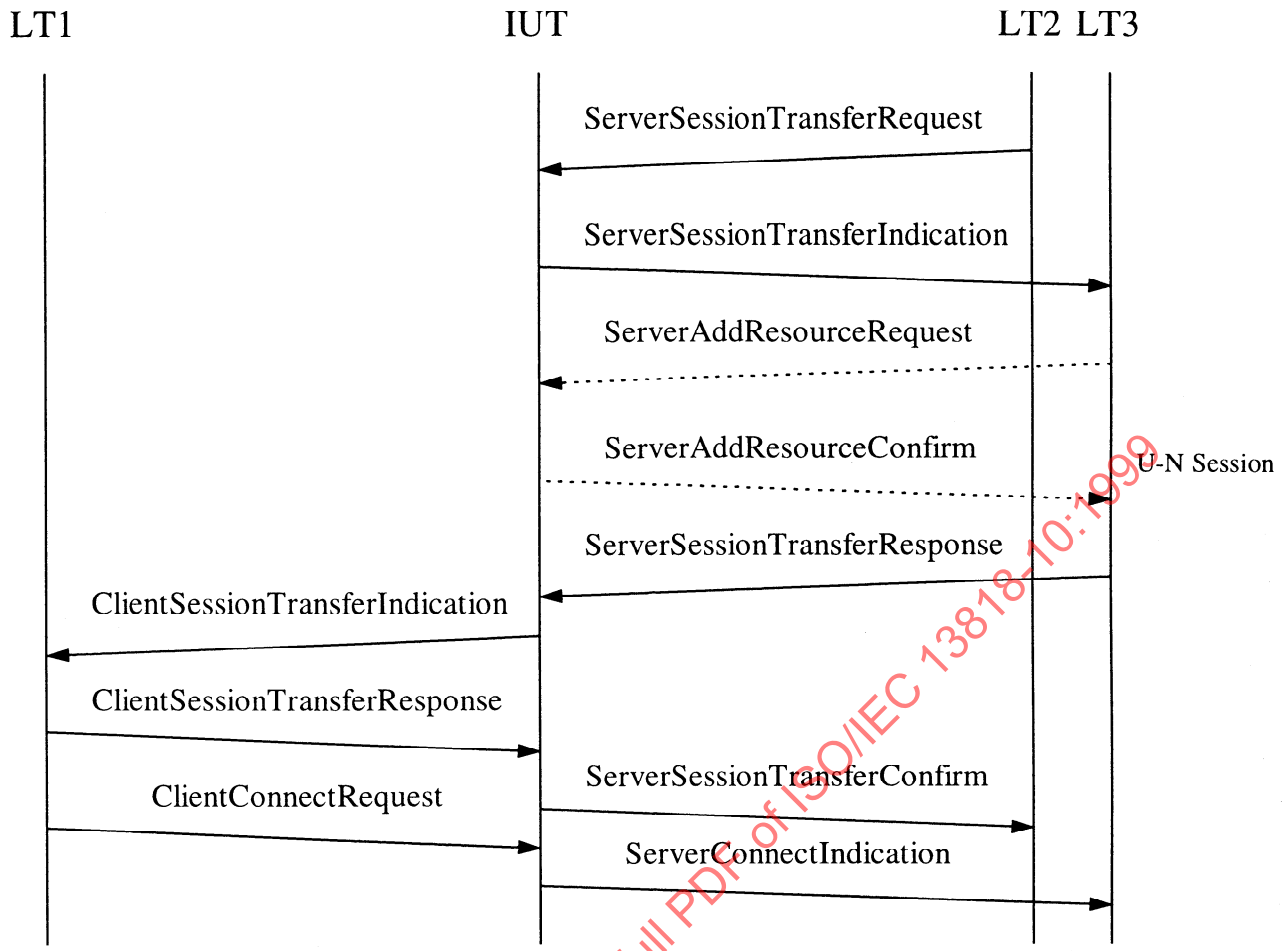
6) LT1 waits for a 'ClientSessionTransferIndication' message.

7) LT1 sends a 'ClientSessionTransferResponse' message.

8) LT2 waits for a 'ServerSessionTransferConfirm' message.

9) LT1 sends a 'ClientConnectRequest' message.

10) LT3 waits for a 'ServerConnectIndication' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if a 'ServerSessionTransferIndication' messages with the sessionId set to the value of the sessionId forwarded is received from the IUT when the IUT can accept the transfer request, a 'ClientSessionTransferIndication' message with the sessionId set to the value of the sessionId forwarded is received from the IUT on receipt of a 'ServerSessionTransferResponse' messages, all resources allocated to Server A are released by the IUT on receipt of a 'ClientSessionTransferResponse' messages, a 'ServerSessionTransferConfirm' message is received from the IUT, and then a 'ServerConnectIndication' message is received from the IUT on receipt of a 'ClientConnectRequest' message.

5.2.1.6.23 Test Case 23 - Transferring a Session : Network rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the IUT can not process the transfer requested in a 'ServerSessionTransferRequest' message, the transfer request is rejected.

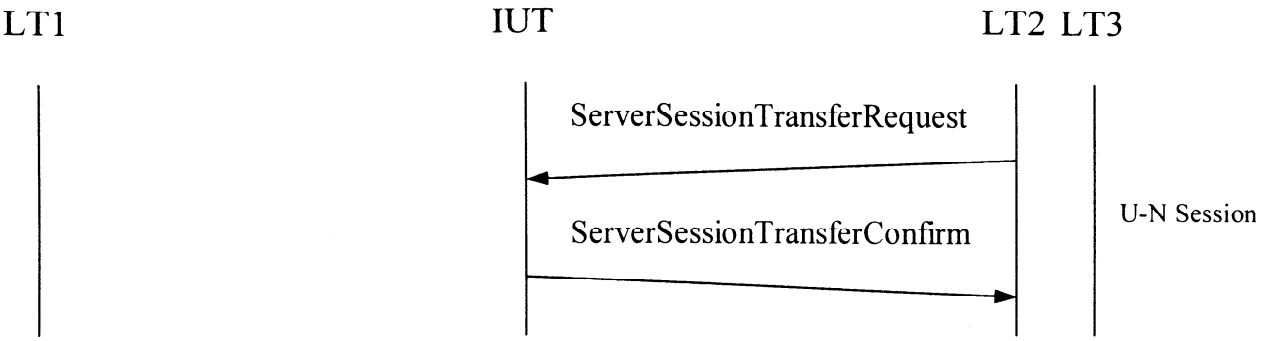
Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT2 sends a 'ServerSessionTransferRequest' message.

2) LT2 waits for a 'ServerSessionTransferConfirm' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferConfirm' message with the response code set to 'rspNeTransferFailed' is received from the IUT when the transfer requested in a 'ServerSessionTransferRequest' message can not be processed.

5.2.1.6.24 Test Case 24 - Transferring a Session : Server B rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT3 sends a 'ServerSessionTransferResponse' message with the response code set to 'rspSeTransferReject, the transfer request is rejected.

Test Preamble:

Establish a Session between the SUT and the LTs.

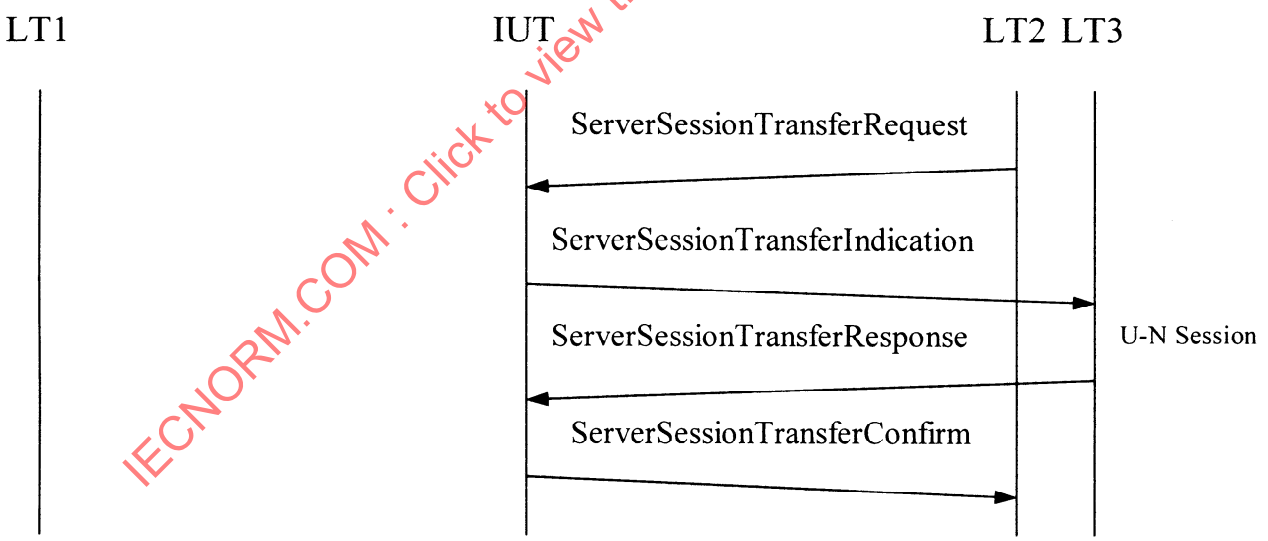
Test Procedure:

1) LT2 sends a 'ServerSessionTransferRequest' message.

2) LT3 waits for a 'ServerSessionTransferIndication' message.

3) LT3 sends a 'ServerSessionTransferResponse' message with the response code set to 'rspSeTransferReject'.

4) LT2 waits for a 'ServerSessionTransferConfirm' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferConfirm' message with the response code set to 'rspSeTransferReject' is received from the IUT on receipt of a 'ServerSessionTransferResponse' message.

5.2.1.6.25 Test Case 25 - Transferring a Session : Server B is unable to allocate resources for transfer

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT3 sends a 'ServerSessionTransferResponse' message with the response code set to 'rspSeTransferNoResource', the transfer request is rejected.

Test Preamble:

Establish a Session between the SUT and the LTs.

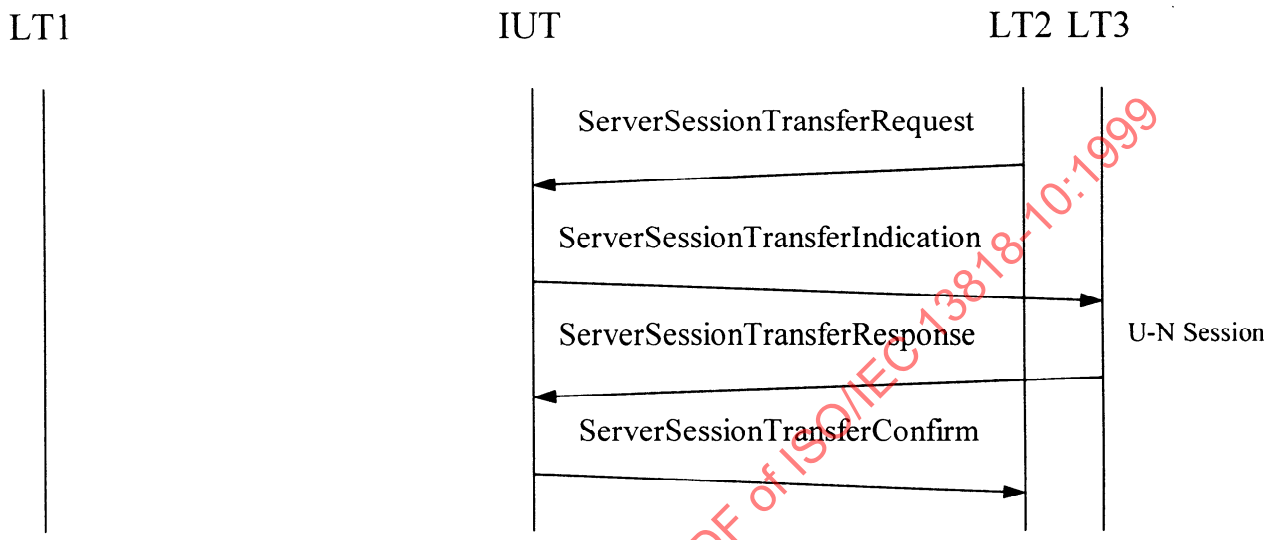
Test Procedure:

1) LT2 sends a 'ServerSessionTransferRequest' message.

2) LT3 waits for a 'ServerSessionTransferIndication' message.

3) LT3 sends a 'ServerSessionTransferResponse' message with the response code set to 'rspSeTransferNoResource'.

4) LT2 waits for a 'ServerSessionTransferConfirm' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferConfirm' message with the response code set to 'rspSeTransferNoResource' is received from the IUT on receipt of a 'ServerSessionTransferResponse' message.

5.2.1.6.26 Test Case 26 - Transferring a Session : Client rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientSessionTransferResponse' message with the response code set to 'rspClTransferReject', the transfer request is rejected

Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT2 sends a 'ServerSessionTransferRequest' message.

2) LT3 waits for a 'ServerSessionTransferIndication' message.

3) LT3 sends a 'ServerAddResourceRequest' message. (Optional)

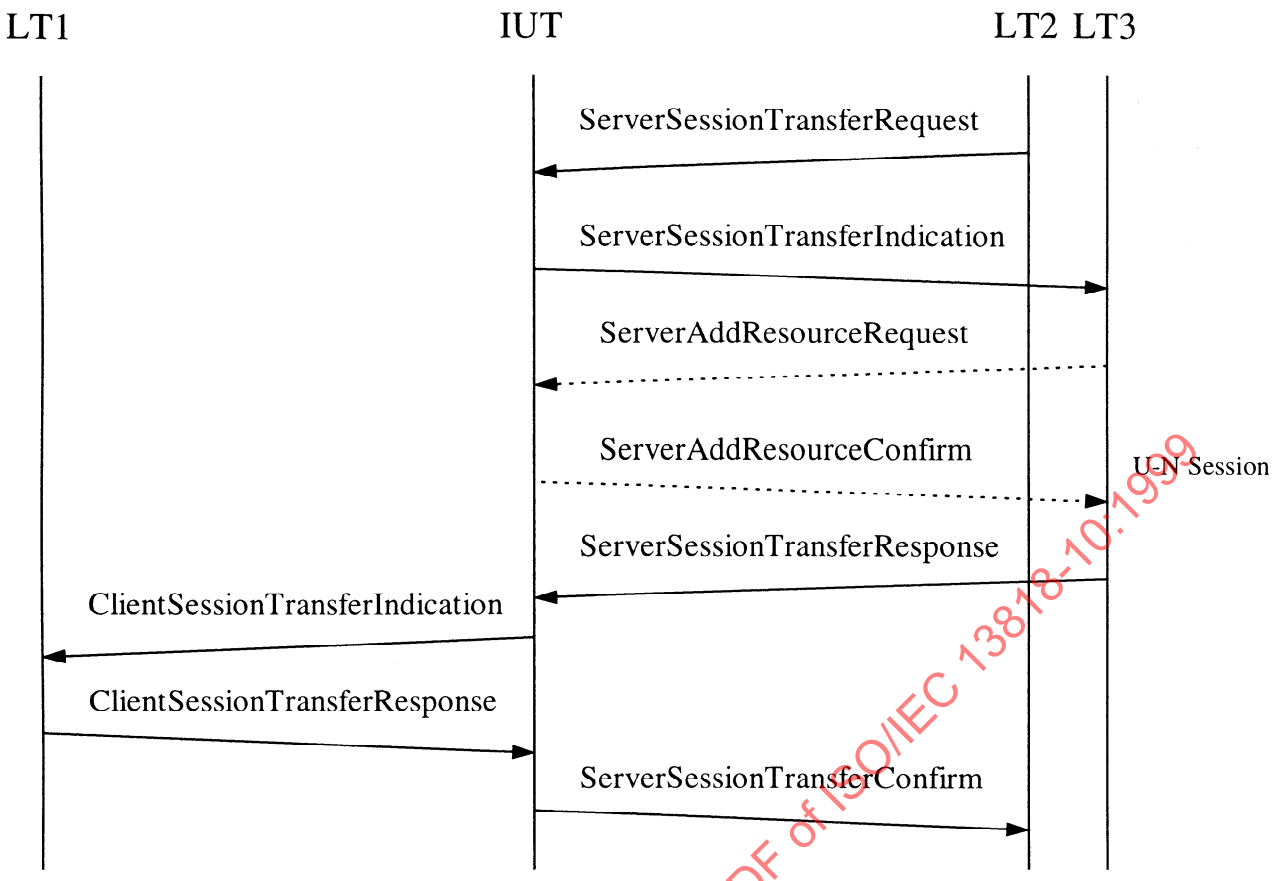
4) LT3 waits for a 'ServerAddResourceConfirm' message. (Optional)

5) LT3 sends a 'ServerSessionTransferResponse' message.

6) LT1 waits for a 'ClientSessionTransferIndication' message.

7) LT1 sends a 'ClientSessionTransferResponse' message with the response code set to 'rspClTransferReject'.

8) LT2 waits for a 'ServerSessionTransferConfirm' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if the Session on Server B is cleared by the IUT on receipt of a 'ClientSessionTransferResponse' message, and then a 'ServerSessionTransferConfirm' messages with the response code set to 'rspClTransferReject' is received from the IUT.

5.2.1.6.27 Test Case 27 - Releasing a transferred Session : Client initiates

Test Purpose:

Verify that after transferring a Session between the SUT and the LTs, when the LT1 sends a 'ClientReleaseRequest' message, the transferred session is released.

Test Preamble:

Transfer a Session between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientReleaseRequest' message.

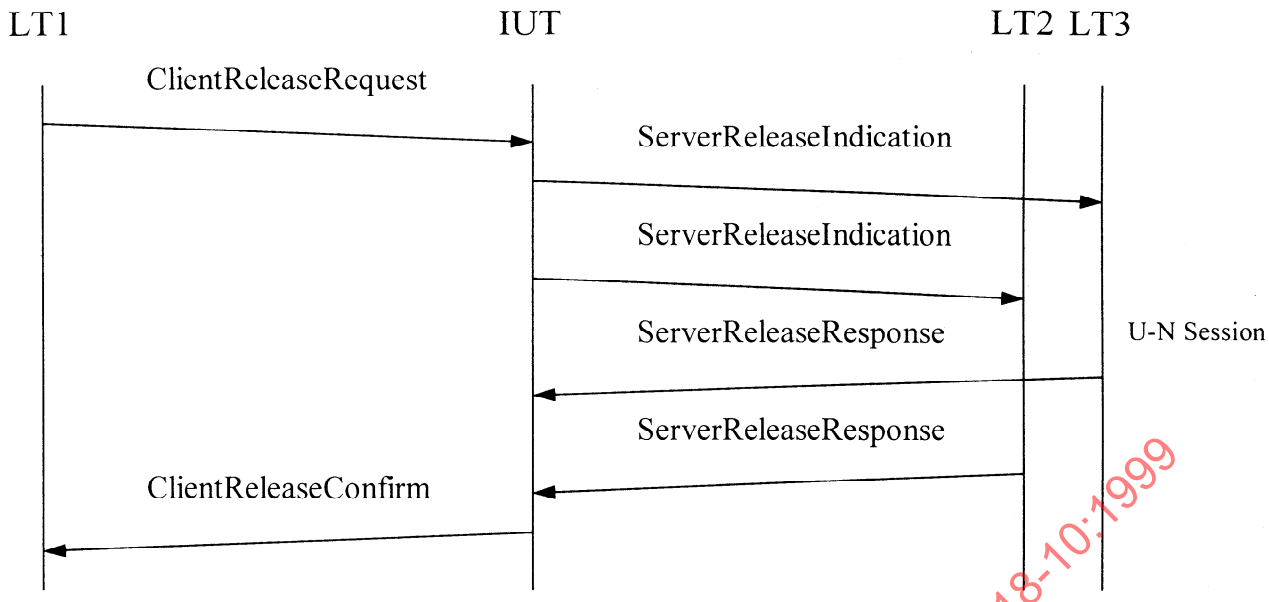
2) LT3 waits for a 'ServerReleaseIndication' message.

3) LT2 waits for a 'ServerReleaseIndication' message.

4) LT3 sends a 'ServerReleaseResponse' message.

5) LT2 sends a 'ServerReleaseResponse' message.

6) LT1 waits for a 'ClientReleaseConfirm' message.



Test Verdict:

Pass the test if two 'ServerReleaseIndication' messages and a 'ClientReleaseConfirm' message are received from the IUT.

5.2.1.6.28 Test Case 28 - Releasing a transferred Session : Server initiates

Test Purpose:

Verify that after transferring a Session between the SUT and the LTs, when the LT3 sends a 'ServerReleaseRequest' message, the transferred session is released.

Test Preamble:

Transfer a Session between the SUT and the LTs.

Test Procedure:

1) LT3 sends a 'ServerReleaseRequest' message.

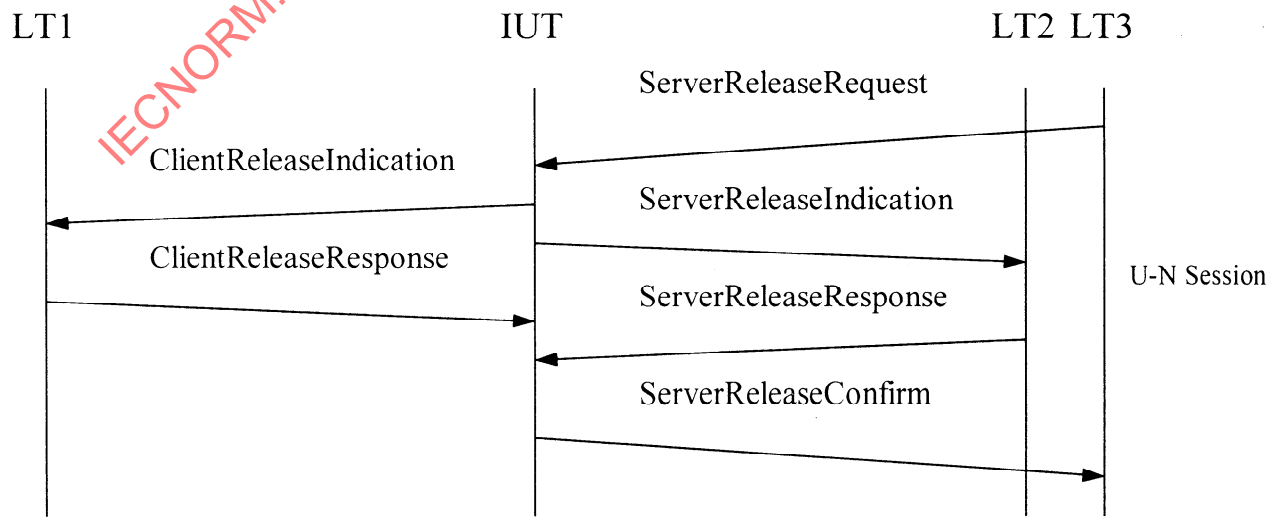
2) LT1 waits for a 'ClientReleaseIndication' message.

3) LT2 waits for a 'ServerReleaseIndication' message.

4) LT1 sends a 'ClientReleaseResponse' message.

5) LT2 sends a 'ServerReleaseResponse' message.

6) LT3 waits for a 'ServerReleaseConfirm' message.



Test Verdict:

Pass the test if a 'ClientReleaseIndication' message and a 'ServerReleaseIndication' message are received from the IUT on receipt of a 'ServerReleaseRequest' message, and then a 'ServerReleaseConfirm' messages with the sessionId set to the value of the sessionId of a 'ClientReleaseResponse' message is received from the IUT.

5.2.1.6.29 Test Case 29 - Releasing a Session : Network initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, the IUT initiates to release a Session.

Test Preamble:

Establish a Session between the SUT and the LTs.

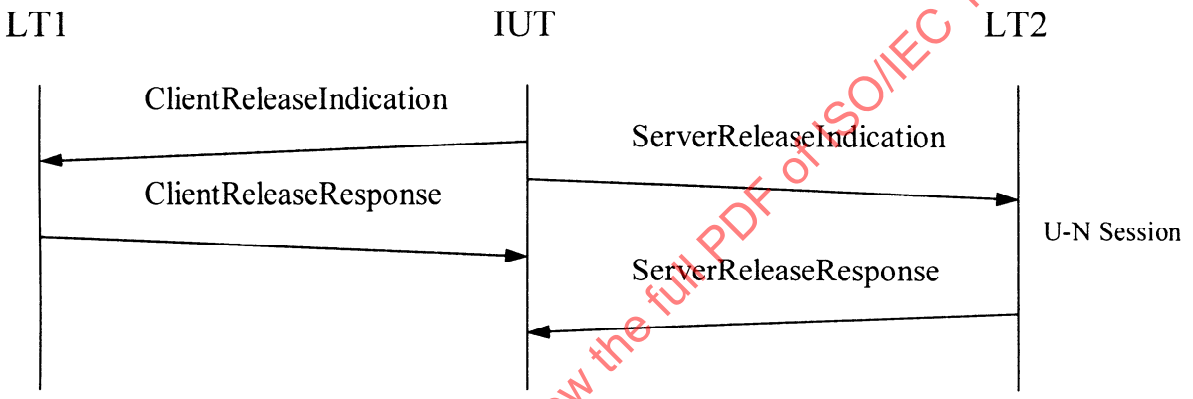
Test Procedure:

1) IUT sends a 'ClientReleaseIndication' message.

2) IUT sends a 'ServerReleaseIndication' message.

3) LT1 sends a 'ClientReleaseResponse' message.

4) LT2 sends a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if a 'ClientReleaseIndication' message and a 'ServerReleaseIndication' message with the sessionId set to the value of an existing Session are received from the IUT, and then all Client and Server interface resources assigned to the session are released by the IUT on receipt of a 'ClientReleaseResponse' message and a 'ServerReleaseResponse' message.

5.2.1.6.30 Test Case 30 - Releasing a Continuous Feed Session : Network initiates

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT2, a Session between the SUT and the LT1, the IUT initiates to release a Continuous Feed Session and a Session connected to a Continuous Feed Session.

Test Preamble:

Establish a Continuous Feed Session between the SUT and the LTs.

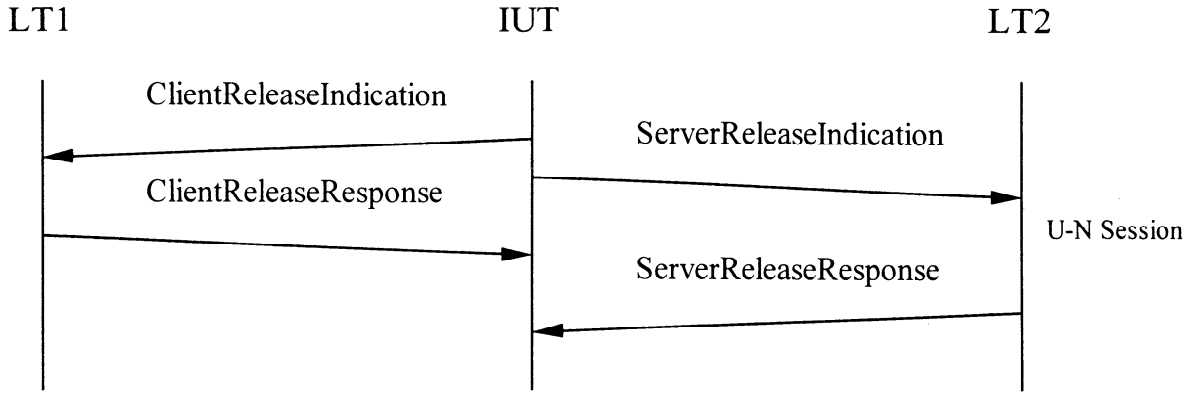
Test Procedure:

1) IUT sends a 'ClientReleaseIndication' message.

2) IUT sends a 'ServerReleaseIndication' message.

3) LT1 sends a 'ClientReleaseResponse' message.

4) LT2 sends a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if a 'ClientReleaseIndication' message with the sessionId set to the value of an existing Session and a 'ServerReleaseIndication' message with the sessionId set to the value of the Continuous Feed Session are received from the IUT, and then all Client and Server interface resources allocated to the corresponding Session are released by the IUT on receipt of a 'ClientReleaseResponse' message and a 'ServerReleaseResponse' message.

5.2.1.6.31 Test Case 31 - Asking a Client Status : Network initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT initiates to request the status of a Client sending a 'ClientStatusIndication' message.

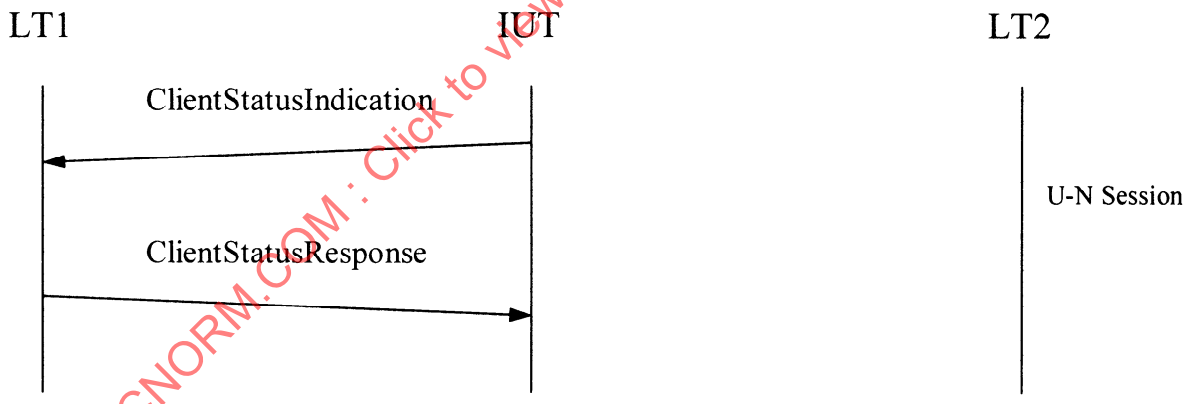
Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) IUT sends a 'ClientStatusIndication' message.

2) LT1 sends a 'ClientStatusResponse' message.



Test Verdict:

Pass the test if all fields of a 'ClientStatusIndication' message received from the IUT are valid and the sequence is terminated by the IUT on receipt of a 'ClientStatusResponse' message.

5.2.1.6.32 Test Case 32 - Asking a Server Status : Network initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT initiates to request the status of a Server sending a 'ServerStatusIndication' message

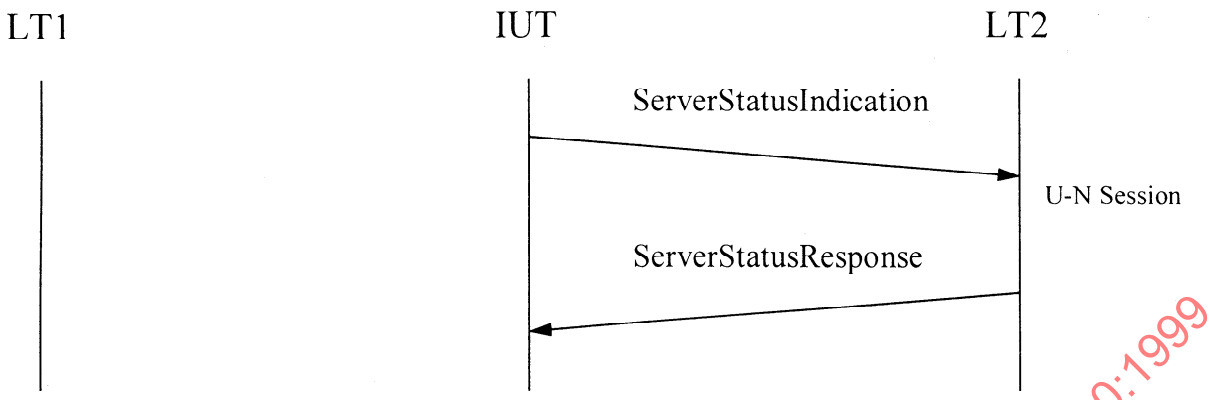
Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) IUT sends a 'ServerStatusIndication' message.

2) LT1 sends a 'ServerStatusResponse' message.



Test Verdict:

Pass the test if all fields of a 'ServerStatusIndication' message received from the IUT are valid and the sequence is terminated by the IUT on receipt of a 'ServerStatusResponse' message.

5.2.1.6.33 Test Case 33 - Resetting for system recovery : Client initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the LT1 sends a 'ClientResetRequest' message, all active sessions are cleared.

Test Preamble:

Establish a network connection between the SUT and the LTs.

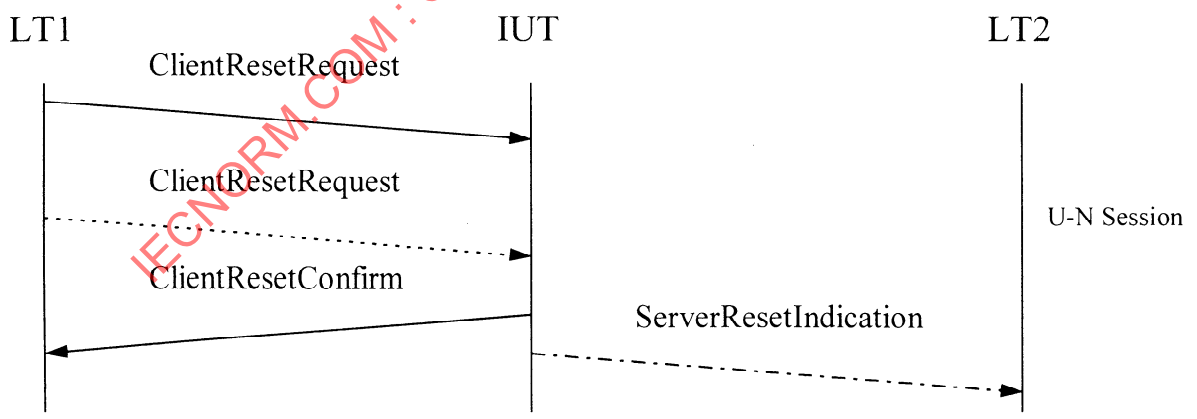
Test Procedure:

1) LT1 sends a 'ClientResetRequest' message.

2) LT1 sends again a 'ClientResetRequest' message after the timer tMsg expires. (Optional)

3) LT1 waits for a 'ClientResetConfirm' message.

4) LT2 waits for a 'ServerResetIndication' message. (Optional)



.....: The message may be sent only when the timer tMsg expires.

-.-.-.-: The message may be sent only if appropriate.

Test Verdict:

Pass the test if all active sessions with the Client are cleared and all timers reset and all resources released by the IUT on receipt of a 'ClientResetRequest' message, and a 'ClientResetConfirm' message is received from the IUT on successful execution, and then a 'ServerResetIndication' message is received from the IUT when it is appropriate.

5.2.1.6.34 Test Case 34 - Resetting for system recovery : Server initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the LT2 sends a 'ServerResetRequest' message, all sessions are cleared.

Test Preamble:

Establish a network connection between the SUT and the LTs.

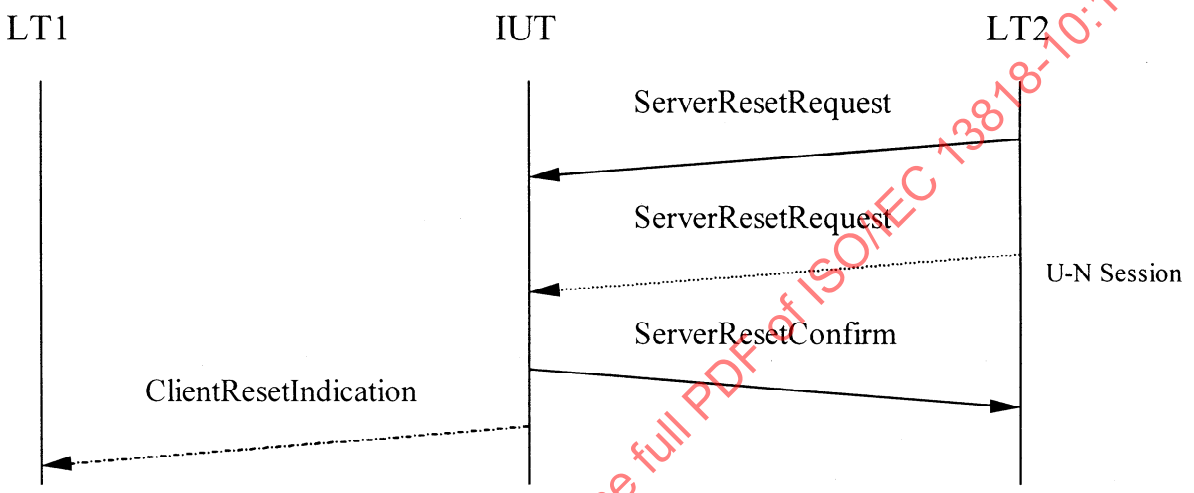
Test Procedure:

1) LT2 sends a 'ServerResetRequest' message.

2) LT2 sends again a 'ServerResetRequest' message after the timer tMsg expires. (Optional)

3) LT2 waits for a 'ServerResetConfirm' message.

4) LT1 waits for a 'ClientResetIndication' message. (Optional)



.....: The message may be sent only when the timer tMsg expires.

-.-.-.-: The message may be sent only if appropriate.

Test Verdict:

Pass the test if all active sessions with the Server are cleared and all timers reset and all resources released by the IUT on receipt of a 'ServerResetRequest' message, and a 'ServerResetConfirm' message is received from the IUT on successful execution, and then a 'ClientResetIndication' message is received from the IUT when it is appropriate.

5.2.1.6.35 Test Case 35 - Resetting for system recovery : Network initiates reset to a Server

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT initiates to clear all sessions with a Server sending a 'ServerResetIndication' message.

Test Preamble:

Establish a network connection between the SUT and the LTs.

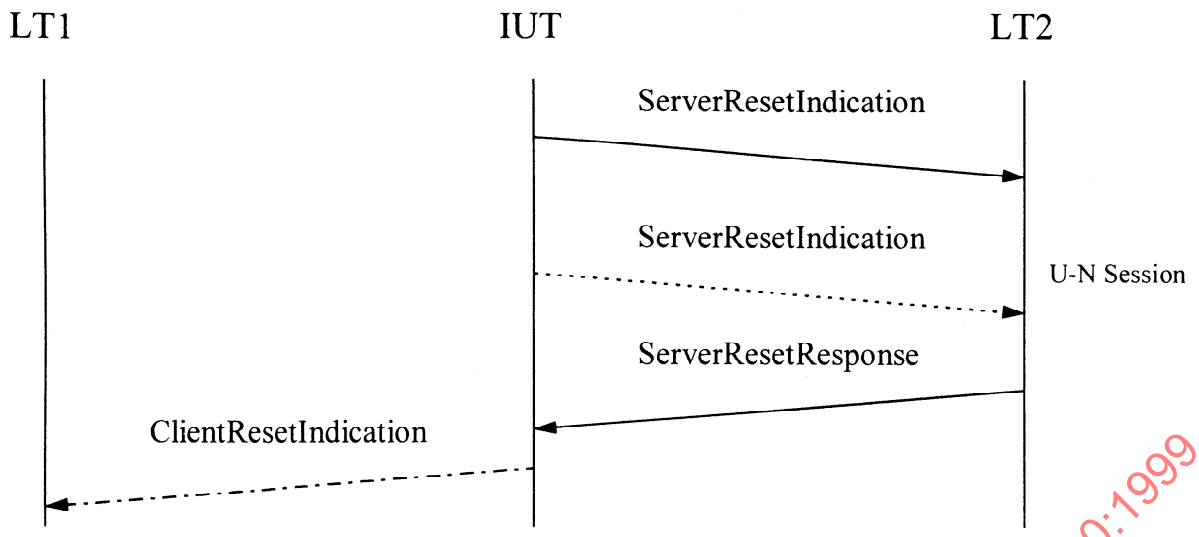
Test Procedure:

1) IUT sends a 'ServerResetIndication' message.

2) IUT sends again a 'ServerResetIndication' message after the timer tMsg expires. (Optional)

3) LT2 sends a 'ServerResetResponse' message.

4) LT1 waits for a 'ClientResetIndication' message. (Optional)



.....: The message may be sent only when the timer tMsg expires.

- - - - - -: The message may be sent only if appropriate.

Test Verdict:

Pass the test if the Reset procedure is terminated by the IUT when no response is received after the second timer tMsg expires, or the timer tMsg is stopped in the IUT on receipt of a 'ServerResetResponse' message, and then a 'ClientResetIndication' message is received from the IUT when it is appropriate.

5.2.1.6.36 Test Case 36 - Resetting for system recovery : Network initiates reset to a Client

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT initiates to clear all sessions with a Client sending a 'ClientResetIndication' message.

Test Preamble:

Establish a network connection between the SUT and the LTs.

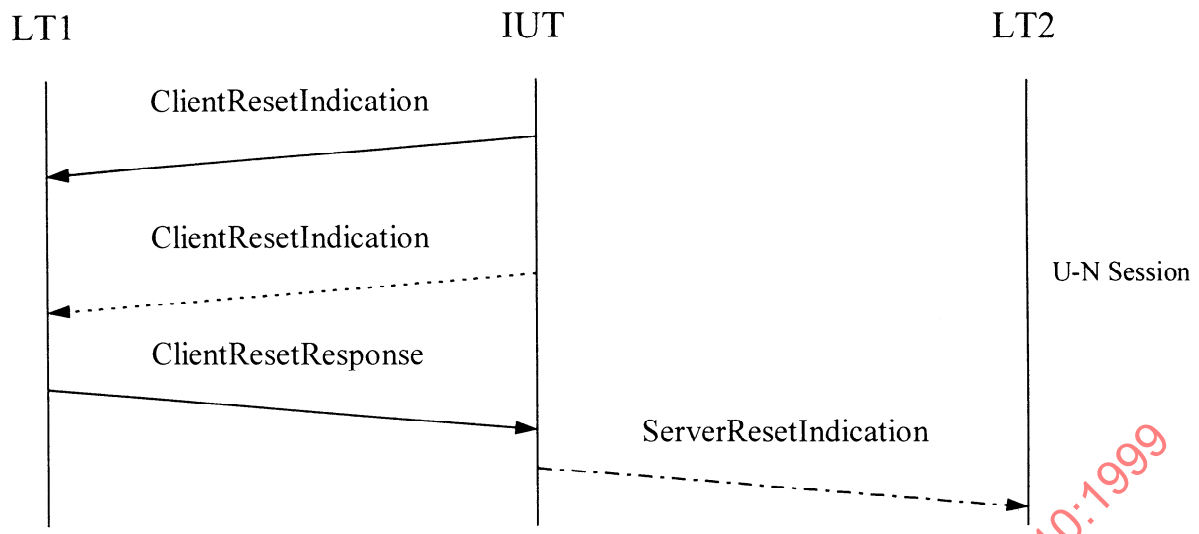
Test Procedure:

1) IUT sends a 'ClientResetIndication' message.

2) IUT sends again a 'ClientResetIndication' message after the timer tMsg expires. (Optional)

3) LT1 sends a 'ClientResetResponse' message.

4) LT2 waits for a 'ServerResetIndication' message. (Optional)



.....: The message may be sent only when the timer tMsg expires.

- - - - - -: The message may be sent only if appropriate.

Test Verdict:

Pass the test if the Reset procedure is terminated by the IUT when no response is received after the second timer tMsg expires, or the timer tMsg is stopped in the IUT on receipt of a 'ClientResetResponse' message, and then a 'ServerResetIndication' message is received from the IUT when it is appropriate.

5.2.1.7 DSM-CC U-N Session (Server)

Table 25 – DSM-CC User-to-Network Session Test Cases for Server Testing

Test Case No.	Test Case Names	DSM-CC U-N Session Scenarios	Reference to ISO/IEC 13818-6
1	Setting up a new Session : Client initiates	Client Session Set-Up Command Sequence	4.8.1
2	Setting up a new Session : Server rejects	Client Session Set-Up Command Sequence	4.8.1
3	Setting up a new Session : Client has Final UserData()	Client Session Set-Up Command Sequence	4.8.1
4	Releasing a Session : Client initiates	Client Session Release Command Sequence	4.8.2
5	Releasing a Session : Server rejects	Client Session Release Command Sequence	4.8.2
6	Setting up a Continuous Feed Session (CFS)	Server Continuous Feed Session Set-Up Command Sequence	4.9.1
7	Adding additional resources to an existing Session	Server Add Resource Command Sequence	4.9.2
8	Deleting resources from an existing Session	Server Session Delete Resource Command Sequence	4.9.3
9	Releasing a Session : Server initiates	Server Session Release Command Sequence	4.9.4
10	Releasing a Session : Network rejects	Server Session Release Command Sequence	4.9.4
11	Releasing a Session : Client rejects	Server Session Release Command Sequence	4.9.4

12	Releasing a Continuous Feed Session (CFS) : Server initiates	Server Continuous Feed Session Release Command Sequence	4.9.5
13	Releasing a Continuous Feed Session (CFS) : Network rejects	Server Continuous Feed Session Release Command Sequence	4.9.5
14	Asking a Session status	Server Status Command Sequence	4.9.6
15	Forwarding a new Session : Server A requests	Server Session Forward Command Sequence	4.9.7
16	Forwarding a new Session : Server B responds	Server Session Forward Command Sequence	4.9.7
17	Transferring a Session : Server A initiates	Server Session Transfer Command Sequence	4.9.8
18	Transferring a Session : Server B responds	Server Session Transfer Command Sequence	4.9.8
19	Transferring a Session : Network rejects	Server Session Transfer Command Sequence	4.9.8
20	Transferring a Session : Server B rejects	Server Session Transfer Command Sequence	4.9.8
21	Transferring a Session : Server B is unable to allocate resources for transfer	Server Session Transfer Command Sequence	4.9.8
22	Transferring a Session : Client rejects	Server Session Transfer Command Sequence	4.9.8
23	Releasing a Session : Network initiates	Network Initiated Session Release Command Sequence	4.10.1
24	Releasing a Continuous Feed Session : Network initiates	Network Initiated Continuous Feed Session Release Command Sequence	4.10.2
25	Asking a Server Status : Network initiates	Network Initiated Server Status Command Sequence	4.10.4
26	Resetting for system recovery : Server initiates	Server Initiated Reset Command Sequence	4.11.2
27	Resetting for system recovery : Network initiates	Network Initiated Reset Command Sequence	4.11.3

5.2.1.7.1 Test Case 1- Setting up a new Session : Client initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerSessionSetUpIndication' message, the response to the session set-up request is received.

Test Preamble:

Establish a network connection between the SUT and the LT.

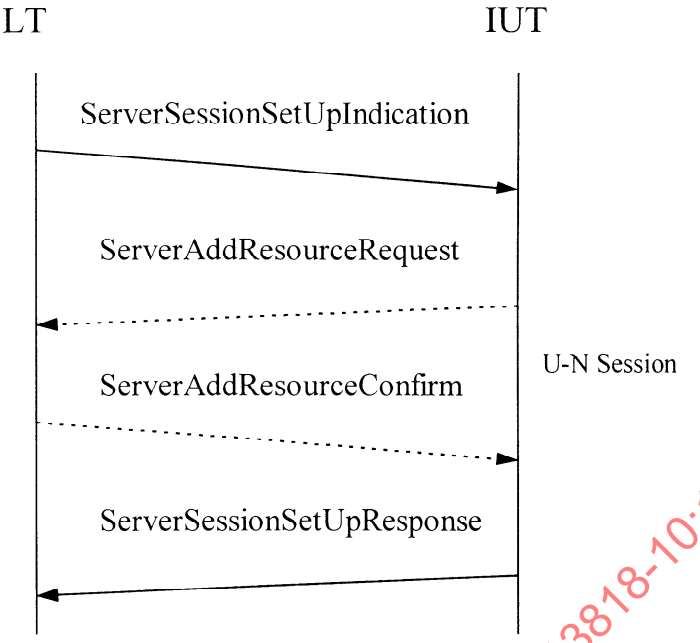
Test Procedure:

1) LT sends a 'ServerSessionSetUpIndication' message.

2) LT waits for a 'ServerAddResourceRequest' message. (Optional)

3) LT sends a 'ServerAddResourceConfirm' message. (Optional)

4) LT waits for a 'ServerSessionSetUpResponse' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if a 'ServerSessionSetUpResponse' message with the response to the session set-up request is received from the IUT.

5.2.1.7.2 Test Case 2 - Setting up a new Session : Server rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when a Sever is unable to accept the session request, the session establishment is rejected.

Test Preamble:

Establish a network connection between the SUT and the LT.

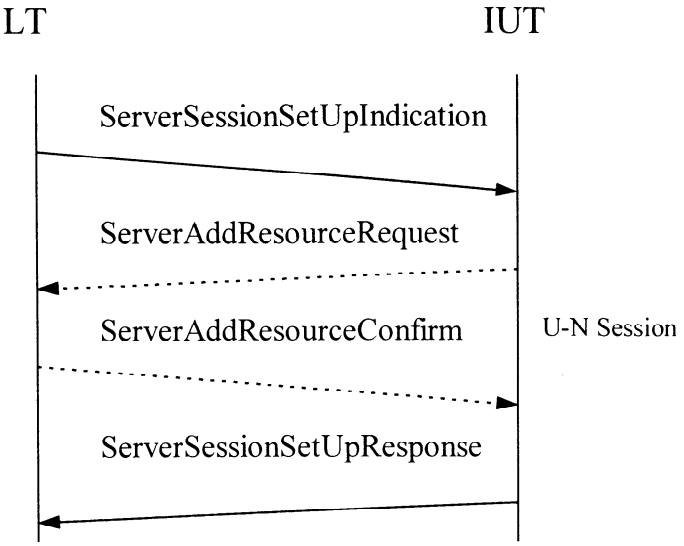
Test Procedure:

1) LT sends a 'ServerSessionSetUpIndication' message.

2) LT waits for a 'ServerAddResourceRequest' message. (Optional)

3) LT sends a 'ServerAddResourceConfirm' message. (Optional)

4) LT waits for a 'ServerSessionSetUpResponse' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if a 'ServerSessionSetUpResponse' message with the sessionId set to the value of sessionId of a 'ServerSessionSetUpIndication' message and the response code set to indicate the reason of the session establishment request rejected such as 'RspNoCalls', 'RspInvalidClientId', or 'RspServiceUnavailable' is received from the IUT when the IUT can not accept the session request.

5.2.1.7.3 Test Case 3 - Setting up a new Session : Client has Final UserData()

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerConnectIndication' message, the session is considered to be established end-to end through a Network.

Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

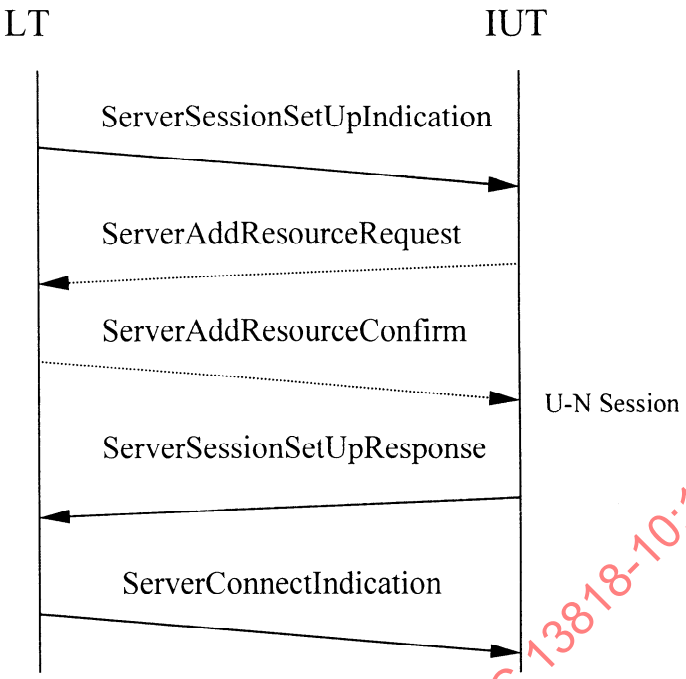
1) LT sends a 'ServerSessionSetUpIndication' message.

2) LT waits for a 'ServerAddResourceRequest' message. (Optional)

3) LT sends a 'ServerAddResourceConfirm' message. (Optional)

4) LT waits for a 'ServerSessionSetUpResponse' message.

5) LT sends a 'ServerConnectIndication' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if the session establishment should be considered by the IUT on receipt of a 'ServerConnectIndication' message.

5.2.1.7.4 Test Case 4 - Releasing a Session : Client initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when the LT sends a 'ServerReleaseIndication' message, all resources assigned to the session are released.

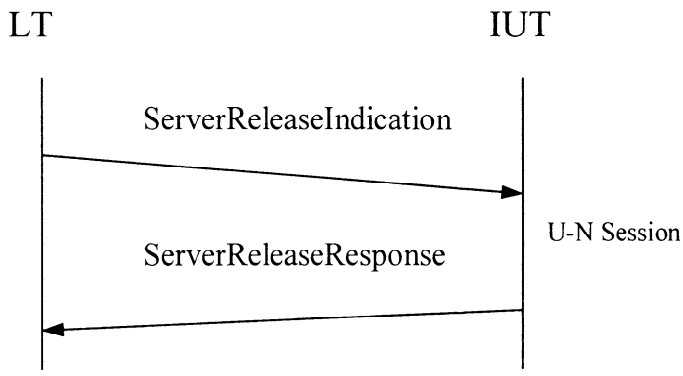
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerReleaseIndication' message.

2) LT waits for a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if all resources assigned to the session are released in the IUT when the received sessionId is valid, and then a 'ServerReleaseResponse' message with the sessionId set to the value of sessionId of a 'ServerReleaseIndication' message is received from the IUT.

5.2.1.7.5 Test Case 5 - Releasing a Session : Server rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when the LT sends an invalid 'ServerReleaseIndication' message, the release procedure is terminated.

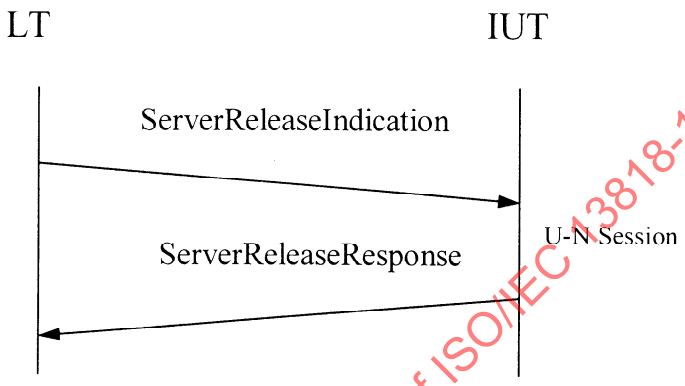
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerReleaseIndication' message with an invalid sessionId.

2) LT waits for a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if a 'ServerReleaseResponse' message with the sessionId set to the value of the sessionId of a 'ServerReleaseIndication' message and the reason code set to 'RspInvalidSessionId' is received from the IUT.

5.2.1.7.6 Test Case 6 - Setting up a new Continuous Feed Session (CFS)

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, the IUT initiates to set up a new Continuous Feed Session (CFS) sending a 'ServerContinuousFeedSessionRequest' message.

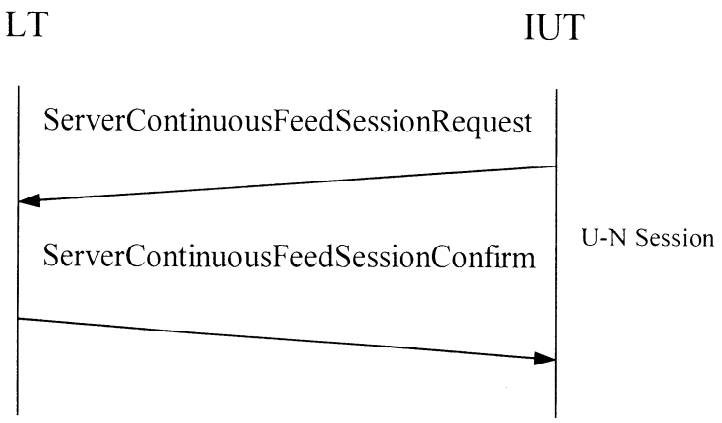
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerContinuousFeedSessionRequest' message.

2) LT sends a 'ServerContinuousFeedSessionConfirm' message.



Test Verdict:

Pass the test if a Continuous Feed Session (CFS) is considered to be active in the IUT when the response code of a 'ServerContinuousFeedSessionConfirm' message indicates the request accepted, or the session is considered to be terminated in the IUT when the response code of a 'ServerContinuousFeedSessionConfirm' message indicates the request rejected.

5.2.1.7.7 Test Case 7 - Adding additional resources to an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, the IUT initiates to add additional resources to an existing Session sending a 'ServerAddResourceRequest' message.

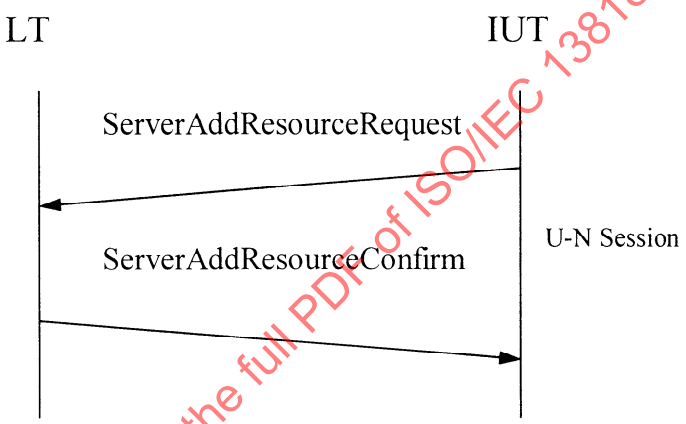
Test Preamble:

Set up a new Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerAddResourceRequest' message.

2) LT sends a 'ServerAddResourceConfirm' message.



Test Verdict:

Pass the test if the additional resources are considered to be committed to the session in the IUT when the response code of a 'ServerAddResourceConfirm' message indicates the successful add resource procedure, or the procedure is considered to be terminated and any of the requested resources is not used by the IUT when the response code of a 'ServerAddResourceConfirm' message indicates the failed add resource procedure.

5.2.1.7.8 Test Case 8 - Deleting resources from an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, the IUT initiates to delete resources from an existing Session sending a 'ServerDeleteResourceRequest' message.

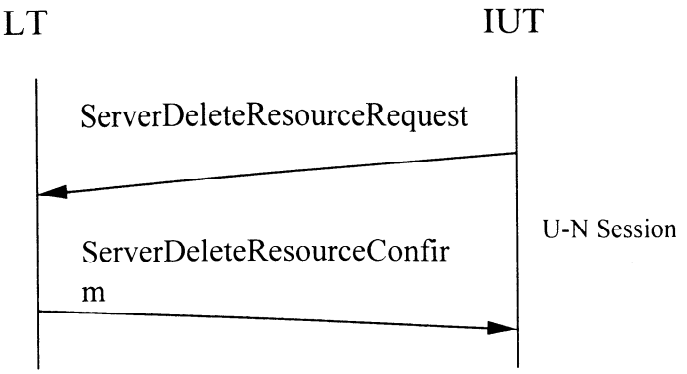
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerDeleteResourceRequest' message.

2) LT sends a 'ServerDeleteResourceConfirm' message.



Test Verdict:

Pass the test if the deleted resources assigned to the existing session are released in the IUT.

5.2.1.7.9 Test Case 9 - Releasing a Session : Server initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, the IUT initiates to release an existing Session sending a 'ServerReleaseRequest' message.

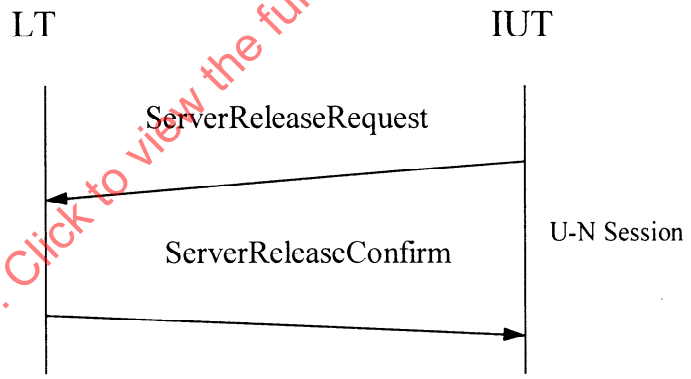
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerReleaseRequest' message.

2) LT sends a 'ServerReleaseConfirm' message.



Test Verdict:

Pass the test if all resources assigned to the existing session are released in the IUT.

5.2.1.7.10 Test Case 10 - Releasing a Session : Network rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when a Network rejects the release request, the release procedure is terminated

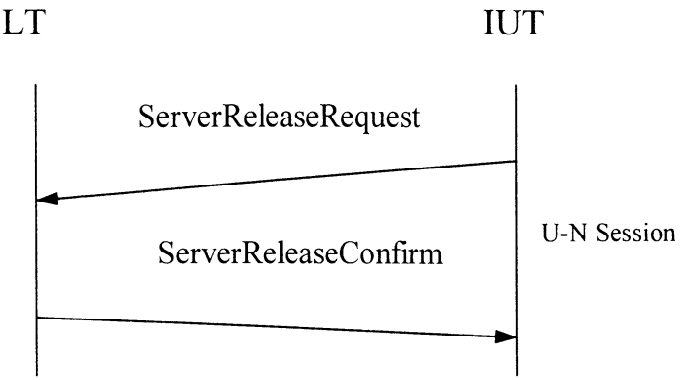
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerReleaseRequest' message.

2) LT sends a 'ServerReleaseConfirm' message with the reason code set to indicate an invalid sessionId or the sessionId not owned by the Server.



Test Verdict:

Pass the test if the session release procedure is terminated by the IUT.

5.2.1.7.11 Test Case 11 - Releasing a Session : Client rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when the IUT receives a 'ServerReleaseConfirm' message with the response code set to indicate the release request rejected in a Client, all resources assigned to a session are released.

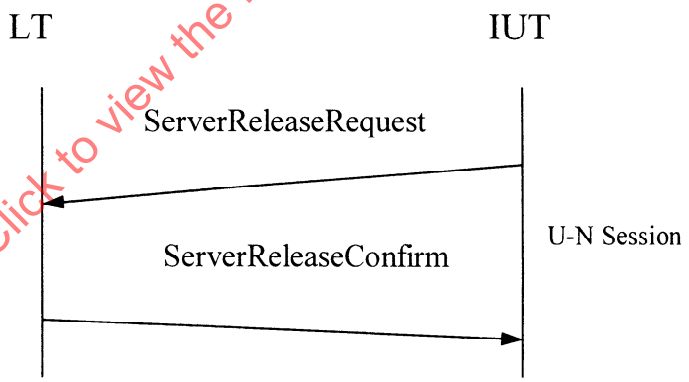
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerReleaseRequest' message.

2) LT sends a 'ServerReleaseConfirm' message with the reason code set to indicate the release request rejected in a Client.



Test Verdict:

Pass the test if all resources assigned to the session are released in the IUT.

5.2.1.7.12 Test Case 12 - Releasing a Continuous Feed Session (CFS) : Server initiates

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT, the IUT initiates to release a Continuous Feed Session sending a 'ServerReleaseRequest' message.

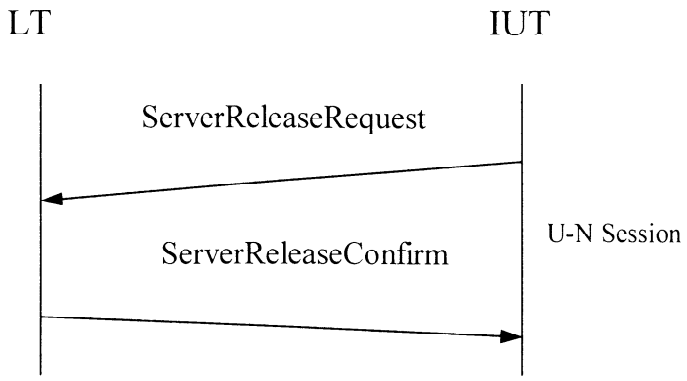
Test Preamble:

Establish a Continuous Feed Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerReleaseRequest' message.

2) LT sends a 'ServerReleaseConfirm' message.



Test Verdict:

Pass the test if all resources assigned to the Continuous Feed Session are released in the IUT.

5.2.1.7.13 Test Case 13 - Releasing a Continuous Feed Session (CFS) : Network rejects

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT, when the LT sends a 'ServerReleaseConfirm' message to indicate the sessionId is invalid or not owned by the Server, the session release procedure is terminated.

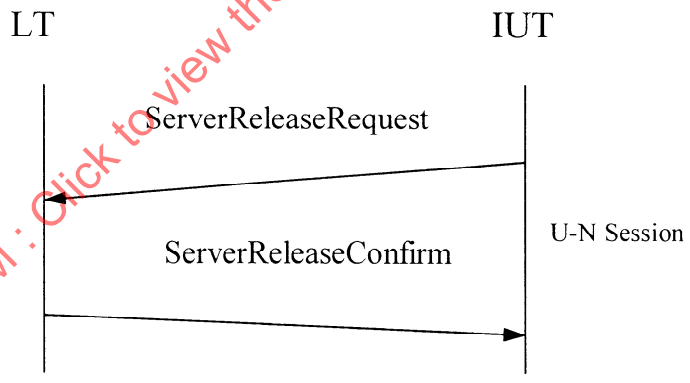
Test Preamble:

Establish a Continuous Feed Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerReleaseRequest' message.

2) LT sends a 'ServerReleaseConfirm' message with the reason code set to indicate an invalid sessionId or the sessionId not owned by the Server.



Test Verdict:

Pass the test if the session release procedure is terminated in the IUT.

5.2.1.7.14 Test Case 14 - Asking a Session status

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, the IUT initiates to request a session status sending a 'ServerStatusRequest' message to a Network.

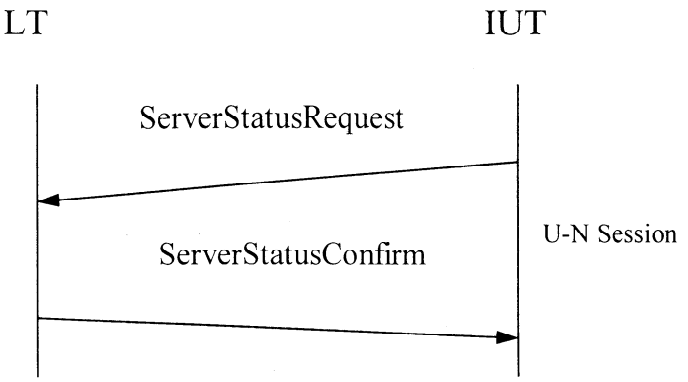
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerStatusRequest' message.

2) LT sends a 'ServerStatusConfirm' message.



Test Verdict:

Pass the test if the requested information is obtained in the IUT.

5.2.1.7.15 Test Case 15 - Forwarding a Session : Server A requests

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerSessionSetUpIndication' message, the decision for forwarding a session is made.

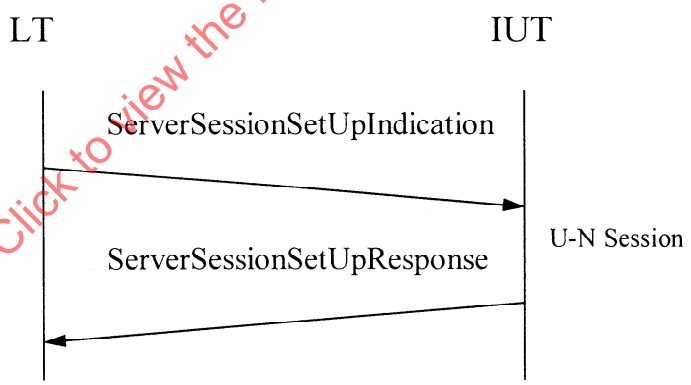
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerSessionSetUpIndication' message with the serverId set to Server A, the forward count set to 0, and no forwardServerIds.

2) IUT waits for a 'ServerSessionSetUpResponse' message.



Test Verdict:

Pass the test if a 'ServerSessionSetUpResponse' message with the response code set to 'RspForward' and the nextServerId set to the userId of the forwarded Server B is received from the IUT when the decision to forward is made.

5.2.1.7.16 Test Case 16 - Forwarding a Session : Server B responds

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerSessionSetUpIndication' message, the decision for forwarding a session is made.

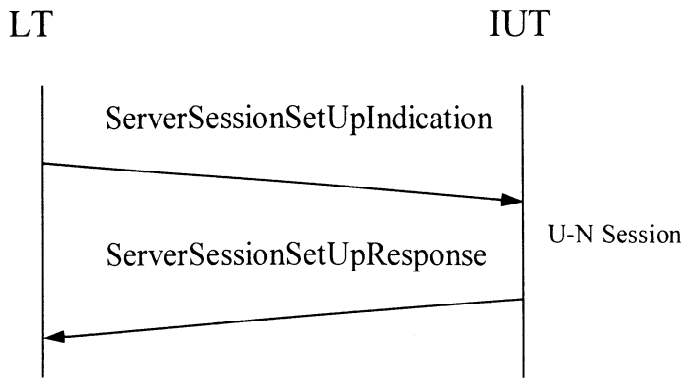
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerSessionSetUpIndication' message with the serverId set to Server B, the forward count set to 1, and the forwardServerId set to the ID of Server A.

2) LT sends a 'ServerSessionSetUpResponse' message.



Test Verdict:

Pass the test if a 'ServerSessionSetUpResponse' message with the response code set to 'RspOK' is received from the IUT.

5.2.1.7.17 Test Case 17 - Transferring a Session : Server A initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, the IUT initiates to transfer control of a session to Server B sending a 'ServerSessionTransferRequest' message.

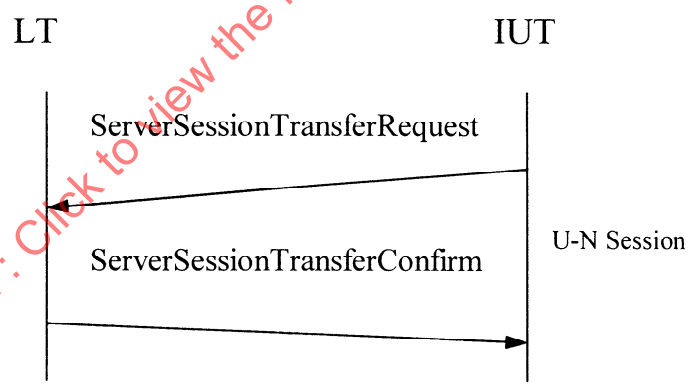
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerSessionTransferRequest' message.

2) LT sends a 'ServerSessionTransferConfirm' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferRequest' message with the sessionId set to the value of the transferred session, the destServerId set to the sessionId of a Server being to transferred to, and the baseServerId set to the value of the base serverId is received from the IUT and then all resources allocated for the Session are released in the IUT on receipt of a 'ServerSessionTransferConfirm' message.

5.2.1.7.18 Test Case 18 - Transferring a Session : Server B responds

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'SeverSessionTransferIndication' message, all necessary resources for the session are allocated.

Test Preamble:

Establish a network connection between the SUT and the LT.

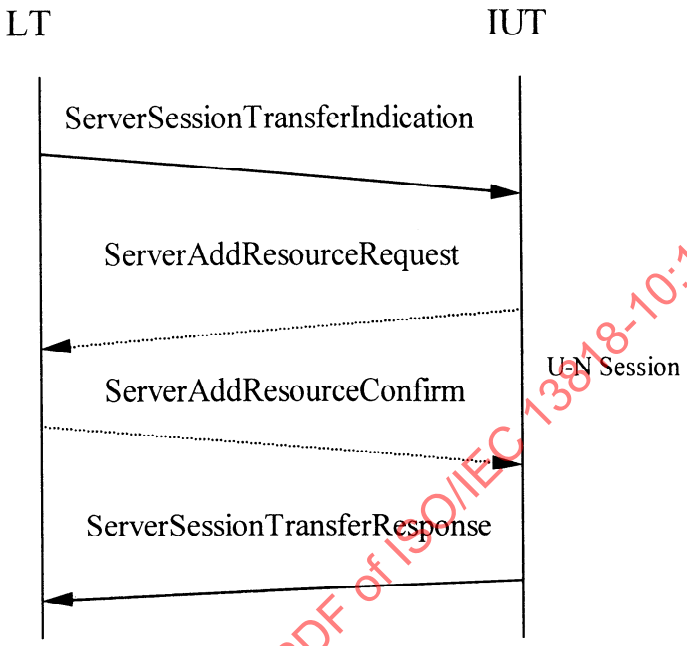
Test Procedure:

1) LT sends a 'ServerSessionTransferIndication' message.

2) LT waits for a 'ServerAddResourceRequest' message. (Optional)

3) LT sends a 'ServerAddResourceConfirm' message. (Optional)

4) LT waits for a 'ServerSessionTransferResponse' message.



.....: These messages may be sent zero or only one time.

Test Verdict:

Pass the test if all necessary resources for the session are allocated in the IUT when the IUT can accept the transfer request, and then a 'ServerSessionTransferResponse' message with the sessionId set to the value of the session being transferred and the response code set to 'RspOK' is received from the IUT.

5.2.1.7.19 Test Case 19 - Transferring a Session: Network rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when the LT sends a 'ServerSessionTransferConfirm' message to indicate the transfer request rejected, the request procedure of transferring a Session is terminated.

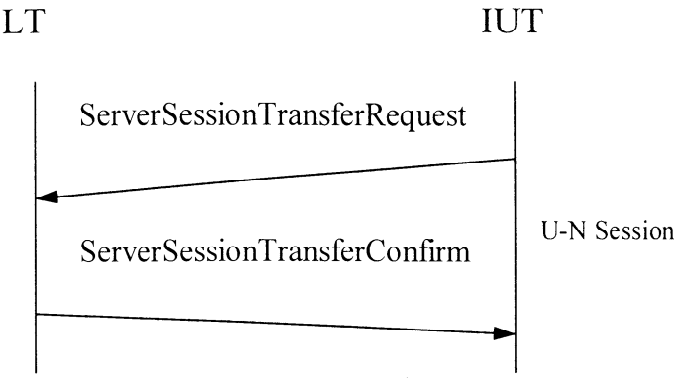
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerSessionTransferRequest' message.

2) LT sends a 'ServerSessionTransferConfirm' message with the response code set to 'RspNeTransferFailed'.



Test Verdict:

Pass the test if the transfer is considered to be failed in the IUT on receipt of a 'ServerSessionTransferConfirm' message.

5.2.1.7.20 Test Case 20 - Transferring a Session: Server B rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerSessionTransferIndication' message, the request of transferring a Session is rejected.

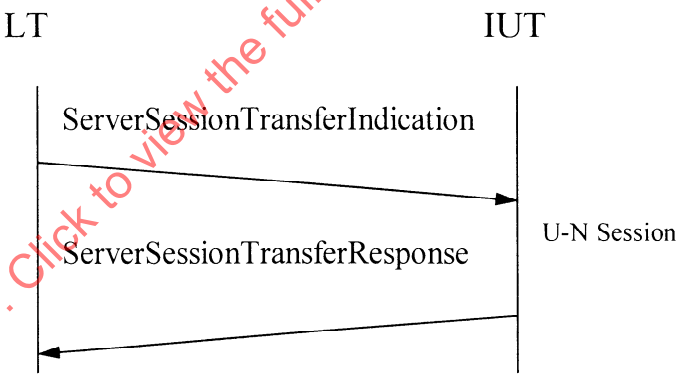
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerSessionTransferIndication' message.

2) LT waits for a 'ServerSessionTransferResponse' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferResponse' message with the response code set to 'RspSeTransferReject' is received from the IUT when the IUT is unable to accept the transfer.

5.2.1.7.21 Test Case 21 - Transferring a Session: Server B is unable to allocate resources for transfer

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the IUT is unable to get the resources to accept a transfer, the request of transferring a Session is rejected.

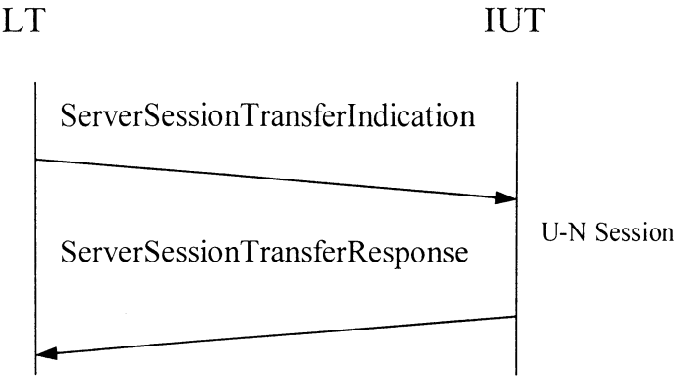
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerSessionTransferIndication' message.

2) LT waits for a 'ServerSessionTransferResponse' message.



Test Verdict:

Pass the test if a 'ServerSessionTransferResponse' message with the response code set to 'RspSeTransferNoResource' is received from the IUT when the IUT is unable to get the resources for the transfer.

5.2.1.7.22 Test Case 22 - Transferring a Session: Client rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerSessionTransferConfirm' message with the response code set to 'RspSeTransferNoResource', the transfer is considered to be failed.

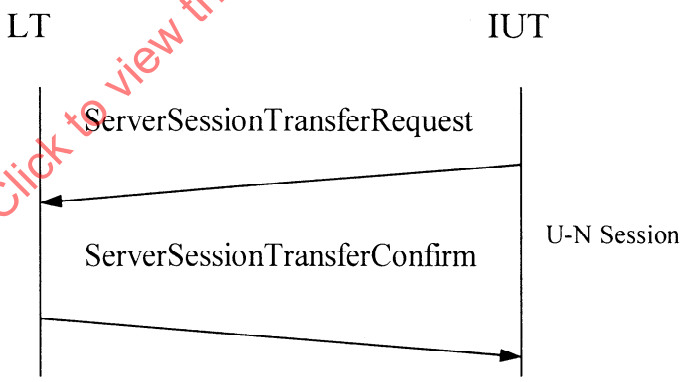
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerSessionTransferRequest' message.

2) LT sends a 'ServerSessionTransferConfirm' message with the response code set to 'RspSeTransferNoResource'.



Test Verdict:

Pass the test if the transfer is considered to be failed in the IUT on receipt of a 'ServerSessionTransferConfirm' message.

5.2.1.7.23 Test Case 23 - Releasing a Session : Network initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LT, when the LT sends a 'ServerReleaseIndication' message, all resources assigned to a session are released.

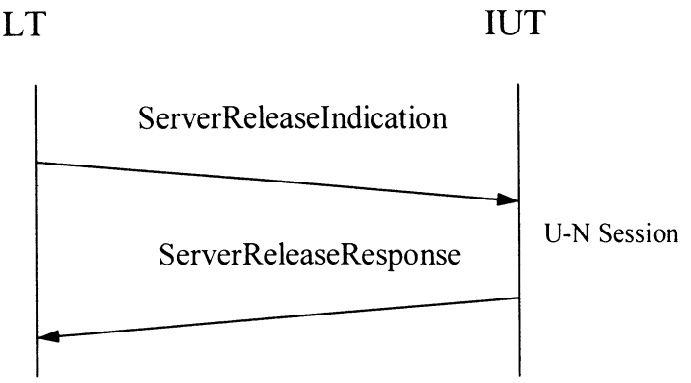
Test Preamble:

Establish a Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerReleaseIndication' message.

2) LT waits for a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if all resources assigned to the session are released in the IUT on receipt of a 'ServerReleaseIndication' message, and then a 'ServerReleaseResponse' message with the sessionId set to the value of the sessionId of a 'ServerReleaseIndication' message is received from the IUT.

5.2.1.7.24 Test Case 24 - Releasing a Continuous Feed Session

Test Purpose:

Verify that after establishing a Continuous Feed Session between the SUT and the LT, when the LT sends a 'ServerReleaseIndication' message, all resources assigned to the Continuous Feed Session are released.

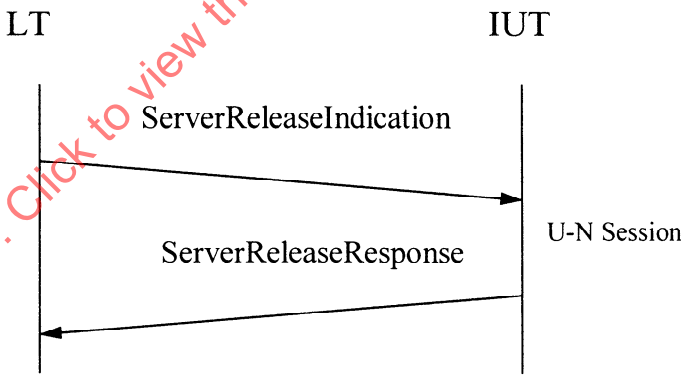
Test Preamble:

Establish a Continuous Feed Session between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerReleaseIndication' message.

2) LT waits for a 'ServerReleaseResponse' message.



Test Verdict:

Pass the test if all resources assigned to the Continuous Feed Session and all resources for any sessions connected to the Continuous Feed Session are released in the IUT on receipt of a 'ServerReleaseIndication' message, and then a 'ServerReleaseResponse' message with the sessionId set to the value of the sessionId of a 'ServerReleaseIndication' message is received from the IUT.

5.2.1.7.25 Test Case 25 - Asking a Server status : Network initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerStatusIndication' message, the status data for the requested status type are received.

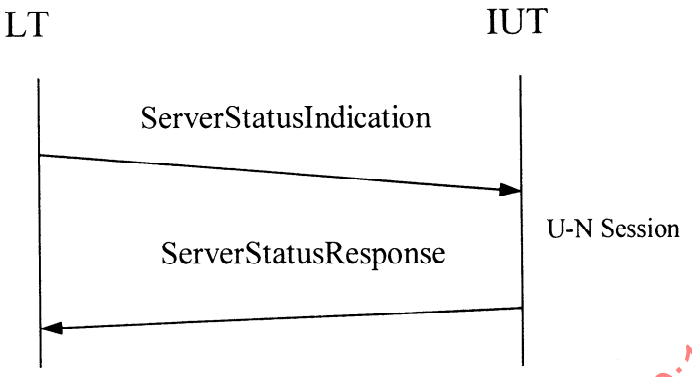
Test Preamble:

Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerStatusIndication' message.

2) LT waits for a 'ServerStatusResponse' message.



Test Verdict:

Pass the test if the requested status information is obtained from the IUT through a 'ServerStatusResponse' message.

5.2.1.7.26 Test Case 26 - Resetting for system recovery : Server initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, the IUT initiates to re-synchronize the interface sending a 'ServerResetRequest' message.

Test Preamble:

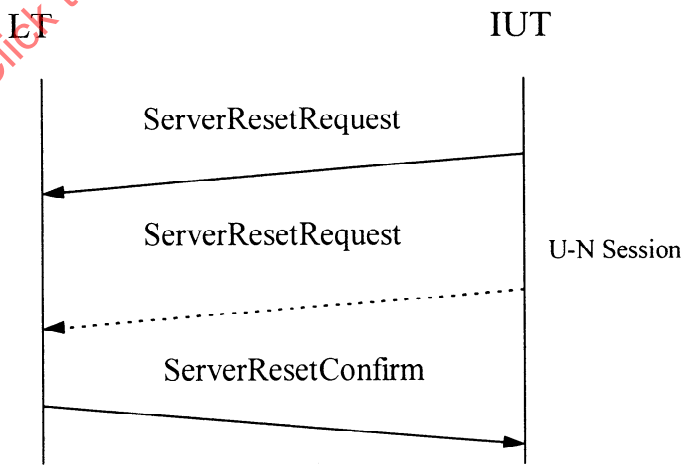
Establish a network connection between the SUT and the LT.

Test Procedure:

1) IUT sends a 'ServerResetRequest' message.

2) IUT sends again a 'ServerResetRequest' message. (Optional)

3) LT sends a 'ServerResetConfirm' message.



.....: The message may be sent only when the timer tMsg expires.

Test Verdict:

Pass the test if the Reset procedure is terminated in the IUT when the second timer tMsg expires without any response, or all resources are placed in the 'idle' state and the timer tMsg is stopped in the IUT on receipt of a 'ServerResetConfirm' message.

5.2.1.7.27 Test Case 27 - Resetting for system recovery : Network initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LT, when the LT sends a 'ServerResetIndication' message, all timers are reset and all resources are released.

Test Preamble:

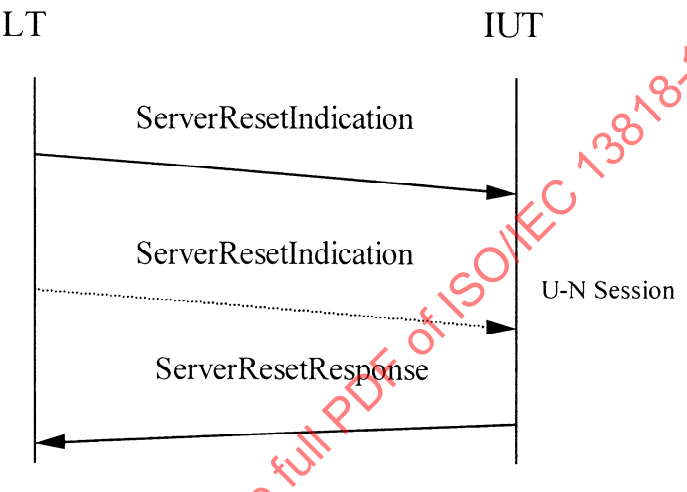
Establish a network connection between the SUT and the LT.

Test Procedure:

1) LT sends a 'ServerResetIndication' message.

2) LT sends again a 'ServerResetIndication' message after the timer tMsg expires. (Optional)

3) LT waits for a 'ServerResetResponse' message.



.....: The message may be sent only when the timer tMsg expires.

Test Verdict:

Pass the test if all Sessions for the SRM are cleared, all timers are reset, and all resources are released in the IUT on receipt of a 'ServerResetIndication' message, and then a 'ServerResetResponse' message is received from the IUT on successful execution.

5.2.1.8 DSM-CC U-N Session (Client)

Table 26 – DSM-CC User-to-Network Session Test Cases for Client Testing

Test Case No.	Test Case Names	DSM-CC U-N Session Scenarios	Reference to ISO/IEC 13818-6
1	Setting up a new Session: Client initiates	Client Session Set-Up Command Sequence	4.8.1
2	Setting up a new Session : Network rejects	Client Session Set-Up Command Sequence	4.8.1
3	Setting up a new Session : Client has Final UserData()	Client Session Set-Up Command Sequence	4.8.1
4	Setting up a new Session : Client initiates early release	Client Session Set-Up Command Sequence	4.8.1
5	Releasing a Session : Client initiates	Client Session Release Command Sequence	4.8.2
6	Releasing a Session : Network rejects	Client Session Release Command Sequence	4.8.2

7	Asking a Session Status	Client Initiated Status Command Sequence	4.8.3
8	Adding additional resources to an existing Session	Server Add Resource Command Sequence	4.9.2
9	Deleting resources from an existing Session	Server Session Delete Resource Command Sequence	4.9.3
10	Asking a Client Status : Network initiates	Network Initiated Client Status Command Sequence	4.10.3
11	Resetting for system recovery : Client initiates	Client Initiated Reset Command Sequence	4.11.1
12	Resetting for system recovery : Network initiates reset to a Client	Network Initiated Reset Command Sequence	4.11.3

5.2.1.8.1 Test Case 1 - Setting up a new Session: Client initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientSessionSetUpConfirm' message, a new Session is set up with the Network.

Test Preamble:

Establish a network connection between the SUT and the LTs.

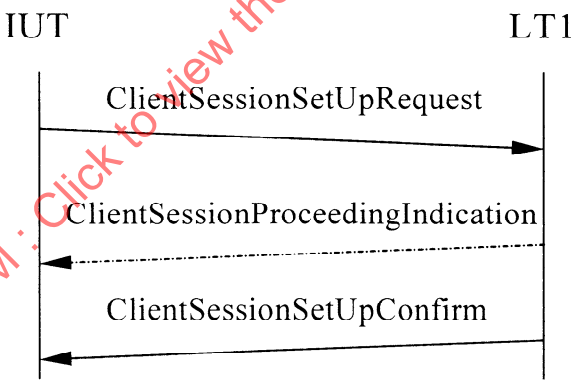
Test Procedure:

1) LT1 waits for a 'ClientSessionSetUpRequest' message.

2) IUT sends a 'ClientSessionSetUpRequest' message.

3) LT1 sends zero or more 'ClientSessionProceedingIndication' messages. (Optional)

4) LT1 sends a 'ClientSessionSetUpConfirm' message.



Test Verdict:

Pass the test if the timer tMsg is terminated in the IUT on receipt of a 'ClientSessionSetUpConfirm' message with the transactionId set to the value of the transactionId of a 'ClientSessionSetUpRequest' message and timer tMsg is reset to its original value if a ClientSessionProceedingIndication message is received.

5.2.1.8.2 Test Case 2 - Setting up a new Session : Network rejects

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the LT1 sends a 'ClientSessionSetUpConfirm message with the invalid clientId or serverId, or the Network cannot support a new session, a new Session request is rejected.

Test Preamble:

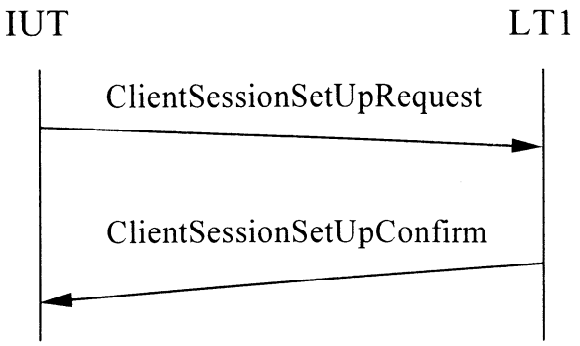
Set up a network connection between the SUT and the LT.

Test Procedure:

1) LT1 waits for a 'ClientSessionSetUpRequest' message.

2) IUT sends a 'ClientSessionSetUpRequest' message.

3) LT1 sends a 'ClientSessionSetUpConfirm' message with a response that indicates that the session is rejected.



Test Verdict:

Pass the test if the IUT terminates the session when a 'ClientSessionSetUpConfirm' message with the transactionId set to the value of the transactionId of a 'ClientSessionSetUpRequest' message and the response code set to indicate the reason of the session request rejected such as 'RspNoCalls', 'RspInvalidClientId', or 'RspInvalidServerId' is received from the LT1.

5.2.1.8.3 Test Case 3 - Setting up a new Session : Client has Final UserData()

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the session has been established and the IUT has final user data, the IUT sends a 'ClientConnectRequest' message with the final user data in the UserData().

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

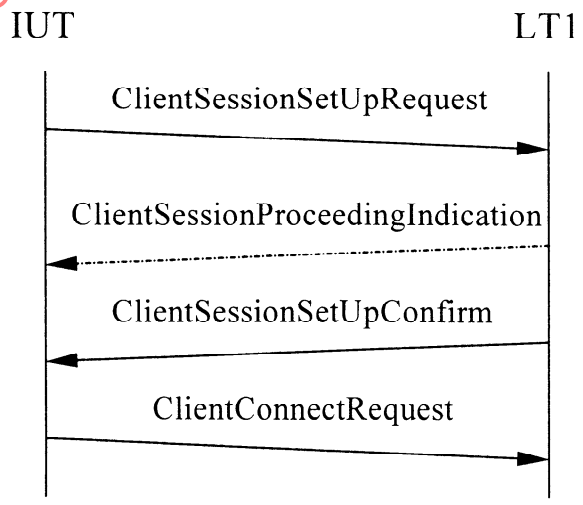
1) LT1 waits for a 'ClientSessionSetUpRequest' message.

2) IUT sends a 'ClientSessionSetUpRequest' message.

3) LT1 sends zero or more 'ClientSessionProceedingIndication' messages. (Optional)

4) LT1 sends a 'ClientSessionSetUpConfirm' message with a response set to RspOK.

5) IUT sends a 'ClientConnectRequest' message with the final user data.



: The message may be sent zero or more times.

Test Verdict:

Pass the test if a 'ClientConnectRequest' message with the sessionId set to the corresponding values of a 'ClientConnectRequest' message is sent to the IUT with the user data in the UserData().

5.2.1.8.4 Test Case 4 - Setting up a new Session : Client initiates Early Release

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the IUT sends a 'ClientReleaseRequest' message prior to the receipt of the first confirm message, the session is terminated and any subsequent messages received for that session are ignored.

Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

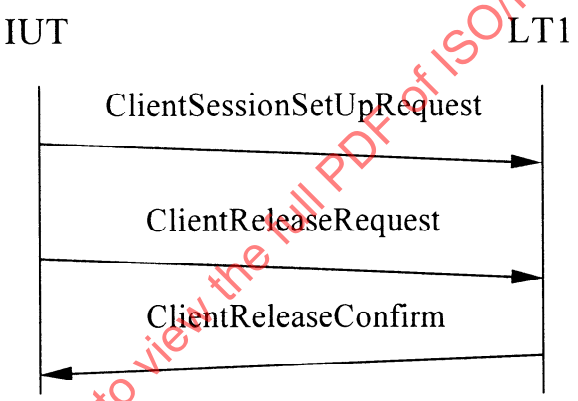
1) LT1 waits for a 'ClientSessionSetUpRequest' message.

2) IUT sends a 'ClientSessionSetUpRequest' message.

3) LT1 delays the response to the 'ClientReleaseRequest' message.

4) IUT sends a ClientReleaseRequest message.

5) LT1 sends a 'ClientReleaseConfirm' message



Test Verdict:

Pass the test if the session is terminated.

5.2.1.8.5 Test Case 5 - Releasing a Session : Client initiates

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the IUT sends a 'ClientReleaseRequest' message, all resources allocated to the Session are released.

Test Preamble:

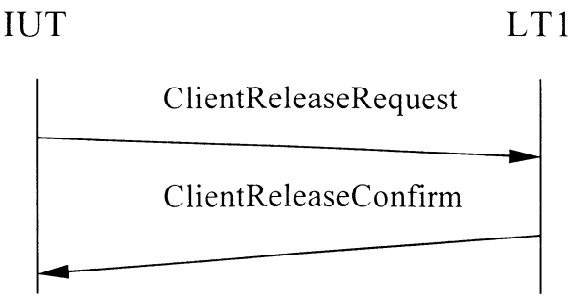
Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT1 waits for a 'ClientReleaseRequest' message.

2) IUT sends a 'ClientReleaseRequest' message.

3) LT1 sends a 'ClientReleaseConfirm' message.



Test Verdict:

Pass the test if a 'ClientReleaseConfirm' message with the sessionId set to the value of the sessionId of a 'ClientReleaseRequest' message and the reason code set to indicate the session released by the SRM is received from the LT1 and all resources are released by the IUT.

5.2.1.8.6 Test Case 6 - Releasing a Session : Network rejects

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the IUT sends a 'ClientReleaseRequest' that is rejected by the LT1, the IUT terminates the session.

Test Preamble:

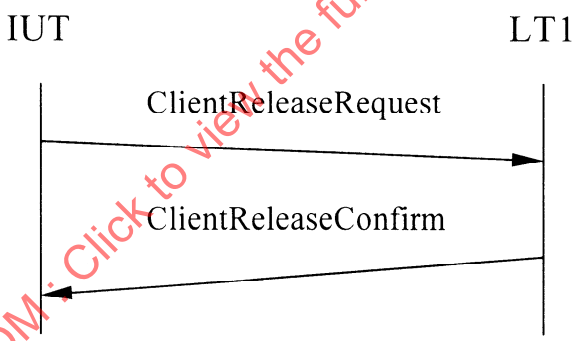
Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT1 waits for a 'ClientReleaseRequest' message.

2) IUT sends a 'ClientReleaseRequest' message.

3) LT1 sends a 'ClientReleaseConfirm' message with the response set to indicate that the request is rejected.



Test Verdict:

Pass the test if all resources for the session are released.

5.2.1.8.7 Test Case 7 - Asking a Session Status

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, when the IUT sends a 'ClientStatusRequest' message, the information of the requested type is obtained.

Test Preamble:

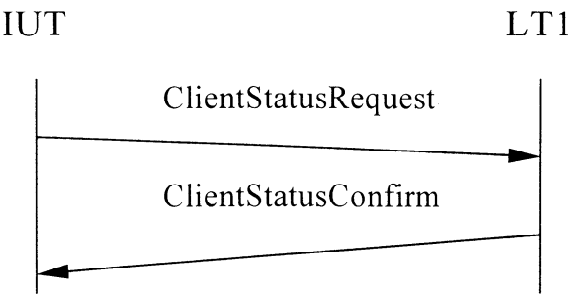
Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 waits for a 'ClientStatusRequest' message.

2) IUT sends a 'ClientStatusRequest' message.

3) LT1 sends a 'ClientStatusConfirm' message.



Test Verdict:

Pass the test if the requested status information is obtained from the LT1 through a 'ClientStatusConfirm' message.

5.2.1.8.8 Test Case 8 - Adding additional resources to an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientAddResourceIndication' message, additional resources are added to the session.

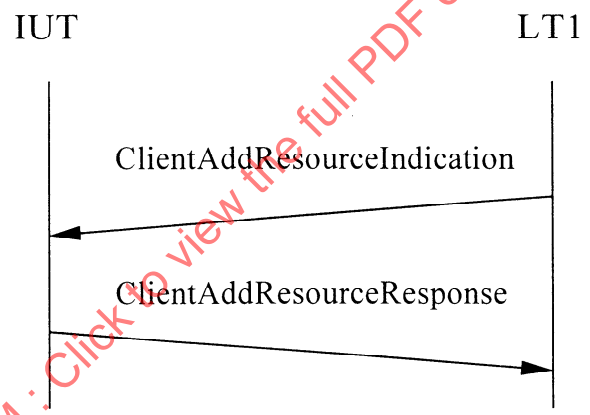
Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientAddResourceIndication' message.

2) IUT sends a 'ClientAddResourceResponse' message.



Test Verdict:

Pass the test if the IUT adds the resources to the session.

5.2.1.8.9 Test Case 9 - Deleting resources from an existing Session

Test Purpose:

Verify that after establishing a Session between the SUT and the LTs, when the LT1 sends a 'ClientDeleteResourceIndication' message, the resources are deleted from an existing session.

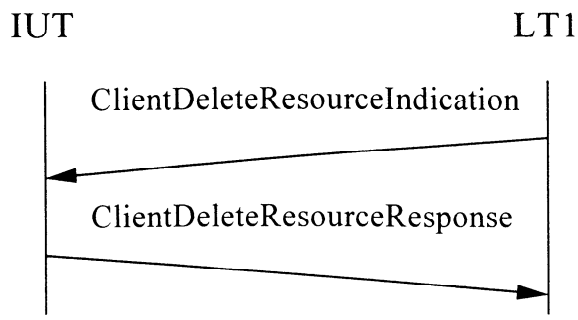
Test Preamble:

Establish a Session between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientDeleteResourceIndication' message.

2) IUT sends a 'ClientDeleteResourceResponse' message.



Test Verdict:

Pass the test if the IUT deletes the resources from the session.

5.2.1.8.10 Test Case 10 - Asking a Client Status : Network initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT responds to an 'ClientStatusIndication' message from LT1.

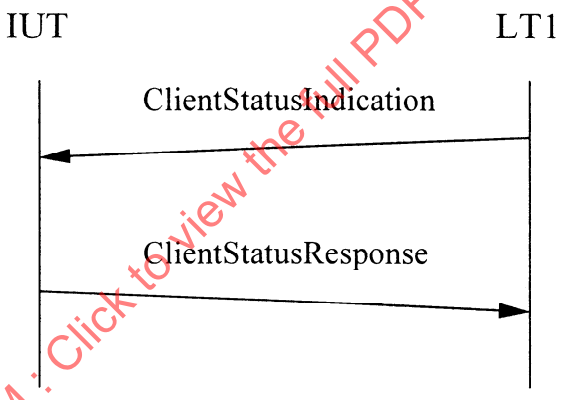
Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientStatusIndication' message.

2) IUT sends a 'ClientStatusResponse' message.



Test Verdict:

Pass the test if the IUT sends a 'ClientStatusResponse' message on receipt of a 'ClientStatusIndication' message.

5.2.1.8.11 Test Case 11 - Resetting for system recovery : Client initiates

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the IUT correctly initiates a system recovery.

Test Preamble:

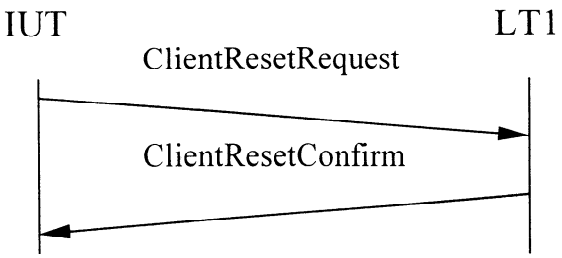
Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 waits for a 'ClientResetRequest' message.

2) IUT sends a 'ClientResetRequest' message

3) LT1 sends a 'ClientResetConfirm' message.



Test Verdict:

Pass the test if all active sessions on the IUT are cleared and all timers reset and all resources released by the IUT when the 'ClientResetConfirm' message is received.

5.2.1.8.12 Test Case 12 - Resetting for system recovery : Network initiates reset to a Client

Test Purpose:

Verify that after establishing a network connection between the SUT and the LTs, the LT1 initiates to clear all sessions with the IUT by sending a 'ClientResetIndication' message.

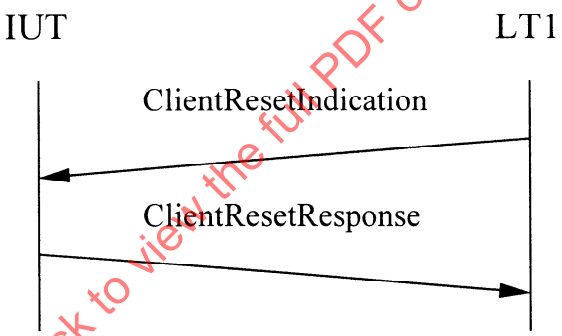
Test Preamble:

Establish a network connection between the SUT and the LTs.

Test Procedure:

1) LT1 sends a 'ClientResetIndication' message.

2) IUT sends a 'ClientResetResponse' message.



Test Verdict:

Pass the test if all active sessions on the IUT are cleared and all timers reset and all resources released by the IUT before the 'ClientResetConfirm' message is sent.

5.2.2 Test Coverage

5.2.2.1 DSM-CC U-N Configuration

DSM-CC U-N Configuration Sequences:	Test Case Group	DSM-CC U-N Configuration Messages:	Reference to ISO/IEC 13818-6
User Initiated Configuration Sequence	1	UNConfigRequest	3.3.1
		UNConfigConfirm	3.3.2
Network Initiated Configuration Sequence	1	UNConfigIndication	3.3.3
		UNConfigResponse	3.3.4
Broadcasting of Configuration messages	1	UNConfigIndication	3.3.3
Mixed User/Network Initiated Configuration Sequence	1	UNConfigRequest	3.3.1
		UNConfigConfirm	3.3.2

1. The Test Cases are included in subclause 5.2.1.2.1

5.2.2.2 DSM-CC U-N Download

DSM-CC U-N Download Scenarios:	Test Case Group	DSM-CC U-N Download Messages:	Reference to ISO/IEC 13818-6
Flow-Controlled Download	2	DownloadInfoRequest	7.3.1
		DownloadInfoResponse	7.3.2
		DownloadDataBlock	7.3.3
		DownloadDataRequest	7.3.4
Non-Flow-Controlled Download	2	DownloadInfoRequest	7.3.1
		DownloadInfoResponse	7.3.2
		DownloadDataBlock	7.3.3
Data Carousel	2	DownloadInfoResponse	7.3.2
		DownloadDataBlock	7.3.3

2. The Test Cases are included in subclause 5.2.1.2.2.

5.2.2.3 DSM-CC U-N SDB-CCP

DSM-CC U-N SDB-CCP Scenarios:	Test Case Group	DSM-CC U-N SDB-CCP Messages:	Reference to ISO/IEC 13818-6
Client Initiated Program Select Command Sequence	1	SDBProgramSelectRequest	10.2.3.1
		SDBProgramSelectConfirm	10.2.3.2
SDB Server Initiated Program Select Command Sequence	1	SDBProgramSelectIndication	10.2.3.3
		SDBProgramSelectResponse	10.2.3.4

1. The test case is defined in subclause 5.2.1.2.3.

5.2.2.4 DSM-CC U-N Pass-Thru

DSM-CC U-N Pass-Thru Scenarios:	Test Case Group	DSM-CC U-N Pass-Thru Messages:	Reference to ISO/IEC 13818-6
Pass-Thru message scenario	2	PassThruRequest	12.2.2.1
		PassThruIndication	12.2.2.2
Pass-Thru receipt message scenario	2	PassThruReceiptRequest	12.2.2.3
		PassThruReceiptConfirm	12.2.2.4
		PassThruReceiptIndication	12.2.2.5
		PassThruReceiptResponse	12.2.2.6

2. The test case is defined in subclause 5.2.1.2.4.

5.2.2.5 DSM-CC U-N Session (SRM)

DSM-CC U-N Session Scenarios:	Test Case Group	DSM-CC U-N Session Messages:	Reference to ISO/IEC 13818-6
Client Session Set-Up Command Sequence	1	ClientSessionSetUpRequest	4.2.4.1
		ClientSessionSetUpConfirm	4.2.4.2
		ServerSessionSetUpIndication	4.2.4.3
		ServerSessionSetUpResponse	4.2.4.4
		ClientSessionProceedingIndication	4.2.11.1
		ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4
		ClientConnectRequest	4.2.12.1
		ServerConnectIndication	4.2.12.2
Client Session Release Command Sequence	1	ClientReleaseRequest	4.2.5.1
		ClientReleaseConfirm	4.2.5.2
		ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Client Initiated Status Command Sequence	1	ClientStatusRequest	4.2.9.1
		ClientStatusConfirm	4.2.9.2
		ServerStatusIndication	4.2.9.7
		ServerStatusResponse	4.2.9.8
Server Continuous Feed Session Command Sequence	1	ServerContinuousFeedSessionRequest	4.2.8.1
		ServerContinuousFeedSessionConfirm	4.2.8.2
Server Add Resource Command Sequence	1	ClientAddResourceIndication	4.2.6.1
		ClientAddResourceResponse	4.2.6.2
		ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4

Server Session Delete Resource Command Sequence	1	ClientDeleteResourceIndication	4.2.7.1
		ClientDeleteResourceResponse	4.2.7.2
		ServerDeleteResourceRequest	4.2.7.3
		ServerDeleteResourceConfirm	4.2.7.4
Server Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
		ServerReleaseRequest	4.2.5.5
		ServerReleaseConfirm	4.2.5.6
Server Continuous Feed Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
		ServerReleaseRequest	4.2.5.5
		ServerReleaseConfirm	4.2.5.6
Server Status Command Sequence	1	ClientStatusIndication	4.2.9.3
		ClientStatusResponse	4.2.9.4
		ServerStatusRequest	4.2.9.5
		ServerStatusConfirm	4.2.9.6
Server Session Forward Command Sequence	1	ClientSessionSetUpRequest	4.2.4.1
		ClientSessionSetUpConfirm	4.2.4.2
		ServerSessionSetUpIndication	4.2.4.3
		ServerSessionSetUpResponse	4.2.4.4
Server Session Transfer Command Sequence	1	ClientSessionTransferIndication	4.2.13.1
		ClientSessionTransferResponse	4.2.13.2
		ServerSessionTransferRequest	4.2.13.3
		ServerSessionTransferConfirm	4.2.13.4
		ServerSessionTransferIndication	4.2.13.5
		ServerSessionTransferResponse	4.2.13.6
		ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4
		ClientConnectRequest	4.2.12.1
		ServerConnectIndication	4.2.12.2
Client Initiated Transferred Session Release Command Sequence	1	ClientReleaseRequest	4.2.5.1
		ClientReleaseConfirm	4.2.5.2
		ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Server Initiated Transferred Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
		ServerReleaseRequest	4.2.5.5

		ServerReleaseConfirm	4.2.5.6
		ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Network Initiated Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
		ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Network Initiated Continuous Feed Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
		ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Network Initiated Client Status Command Sequence	1	ClientStatusIndication	4.2.9.3
		ClientStatusResponse	4.2.9.4
Network Initiated Server Status Command Sequence	1	ServerStatusIndication	4.2.9.7
		ServerStatusResponse	4.2.9.8
Client Initiated Reset Command Sequence	1	ClientResetRequest	4.2.10.1
		ClientResetConfirm	4.2.10.2
		ServerResetIndication	4.2.10.7
Server Initiated Reset Command Sequence	1	ServerResetRequest	4.2.10.5
		ServerResetConfirm	4.2.10.6
		ClientResetIndication	4.2.10.3
Network Initiated Reset Command Sequence	1	ClientResetIndication	4.2.10.3
		ClientResetResponse	4.2.10.4
		ServerResetIndication	4.2.10.7
		ServerResetResponse	4.2.10.8

1. The test case is defined in subclause 5.2.1.2.5.1

5.2.2.6 DSM-CC U-N Session (Server)

DSM-CC U-N Session Scenarios:	Test Case Group	DSM-CC U-N Session Messages:	Reference to ISO/IEC 13818-6
Client Session Set-Up Command Sequence	1	ServerSessionSetUpIndication	4.2.4.3
		ServerSessionSetUpResponse	4.2.4.4
		ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4
		ServerConnectIndication	4.2.12.2
Client Session Release Command Sequence	1	ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8

Server Continuous Feed Session Command Sequence	1	ServerContinuousFeedSessionRequest	4.2.8.1
		ServerContinuousFeedSessionConfirm	4.2.8.2
Server Add Resource Command Sequence	1	ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4
Server Session Delete Resource Command Sequence	1	ServerDeleteResourceRequest	4.2.7.3
		ServerDeleteResourceConfirm	4.2.7.4
Server Session Release Command Sequence	1	ServerReleaseRequest	4.2.5.5
		ServerReleaseConfirm	4.2.5.6
Server Continuous Feed Session Release Command Sequence	1	ServerReleaseRequest	4.2.5.5
		ServerReleaseConfirm	4.2.5.6
Server Status Command Sequence	1	ServerStatusRequest	4.2.9.5
		ServerStatusConfirm	4.2.9.6
Server Session Forward Command Sequence	1	ServerSessionSetUpIndication	4.2.4.3
		ServerSessionSetUpResponse	4.2.4.4
Server Session Transfer Command Sequence	1	ServerSessionTransferRequest	4.2.13.3
		ServerSessionTransferConfirm	4.2.13.4
		ServerSessionTransferIndication	4.2.13.5
		ServerSessionTransferResponse	4.2.13.6
		ServerAddResourceRequest	4.2.6.3
		ServerAddResourceConfirm	4.2.6.4
		ServerConnectIndication	4.2.12.2
Network Initiated Session Release Command Sequence	1	ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Network Initiated Continuous Feed Session Release Command Sequence	1	ServerReleaseIndication	4.2.5.7
		ServerReleaseResponse	4.2.5.8
Network Initiated Server Status Command Sequence	1	ServerStatusIndication	4.2.9.7
		ServerStatusResponse	4.2.9.8
Server Initiated Reset Command Sequence	1	ServerResetRequest	4.2.10.5
		ServerResetConfirm	4.2.10.6
		ClientResetIndication	4.2.10.3
Network Initiated Reset Command Sequence	1	ServerResetIndication	4.2.10.7
		ServerResetResponse	4.2.10.8

1. The test case is defined in subclause 5.2.1.2.5.2

5.2.2.7 DSM-CC U-N Session (Client)

DSM-CC U-N Session Scenarios:	Test Case Group	DSM-CC U-N Session Messages:	Reference to ISO/IEC 13818-6
Client Session Set-Up Command Sequence	1	ClientSessionSetUpRequest	4.2.4.1
		ClientSessionSetUpConfirm	4.2.4.2
		ClientSessionProceedingIndication	4.2.11.1
		ClientConnectRequest	4.2.12.1
Client Session Release Command Sequence	1	ClientReleaseRequest	4.2.5.1
		ClientReleaseConfirm	4.2.5.2
Client Initiated Status Command Sequence	1	ClientStatusRequest	4.2.9.1
		ClientStatusConfirm	4.2.9.2
Server Continuous Feed Session Command Sequence	1	ServerContinuousFeedSessionRequest	4.2.8.1
		ServerContinuousFeedSessionConfirm	4.2.8.2
Server Add Resource Command Sequence	1	ClientAddResourceIndication	4.2.6.1
		ClientAddResourceResponse	4.2.6.2
Server Session Delete Resource Command Sequence	1	ClientDeleteResourceIndication	4.2.7.1
		ClientDeleteResourceResponse	4.2.7.2
Network Initiated Session Release Command Sequence	1	ClientReleaseIndication	4.2.5.3
		ClientReleaseResponse	4.2.5.4
Network Initiated Client Status Command Sequence	1	ClientStatusIndication	4.2.9.3
		ClientStatusResponse	4.2.9.4
Client Initiated Reset Command Sequence	1	ClientResetRequest	4.2.10.1
		ClientResetConfirm	4.2.10.2
Server Initiated Reset Command Sequence	1	ServerResetRequest	4.2.10.5
		ServerResetConfirm	4.2.10.6
		ClientResetIndication	4.2.10.3
Network Initiated Reset Command Sequence	1	ClientResetIndication	4.2.10.3
		ClientResetResponse	4.2.10.4

1. The test case is defined in subclause 5.2.1.2.5.1

5.2.3 DSM-CC User-to-User

This subclause provides the tables to outline the test cases specified for DSM-CC U-U Core and Extended Interfaces. These test cases are specified for testing the implementation of objects at the server side. Each test case is specified in terms of the Test Purpose, a preamble, the test procedures, and the test verdict.

5.2.3.1 DSM-CC UU Core Consumer Interfaces

Test Case No.	Test Case names	DSM-CC UU API	Reference to ISO/IEC 13818-6
1	Closing a reference to an object	DSM_Base_close()	5.5.1.1
2	Stopping a Stream	DSM_Stream_pause()	5.5.1.3.5
3	Resuming a Stream	DSM_Stream_resume()	5.5.1.3.6
4	Obtaining the status of a Stream	DSM_Stream_status()	5.5.1.3.7
5	Resetting a Stream State Machine	DSM_Stream_reset()	5.5.1.3.8
6	Jumping a Stream	DSM_Stream_jump()	5.5.1.3.9
7	Playing a Stream	DSM_Stream_play()	5.5.1.3.10
8	Reading a File	DSM_File_read()	5.5.1.4.3
9	Writing a File	DSM_File_write()	5.5.1.4.4
10	Listing bindings in a Directory	DSM_Directory_list()	5.5.1.5.3
11	Getting more bindings from a Directory	CosNaming_BindingIterator_ next_one() or _next_n()	
12	Discarding any Server-side storage associated with the Iterator	CosNaming_BindingIterator_ destroy()	
13	Resolving an IOR	DSM_Directory_resolve()	5.5.1.5.4
14	Opening a Directory	DSM_Directory_open()	5.5.1.5.13
15	Closing a reference to a Directory	DSM_Directory_close()	5.5.1.5.14
16	Getting the attribute values bound to a given name	DSM_Directory_get()	5.5.1.5.15
17	Attaching to a ServiceGateway domain of Services	DSM_Session_attach()	5.5.1.6.3
18	Detaching from a ServiceGateway domain of Services	DSM_Session_detach()	5.5.1.6.4
19	Obtaining the object reference of the ServiceGateway for the current session.	DSM_First_root()	5.5.1.8.2
20	Obtaining the object reference of the current sessions Primary Service.	DSM_First_service()	5.5.1.8.3

5.2.3.1.1 Test Case 1 - Closing a reference to an object

Test Purpose:

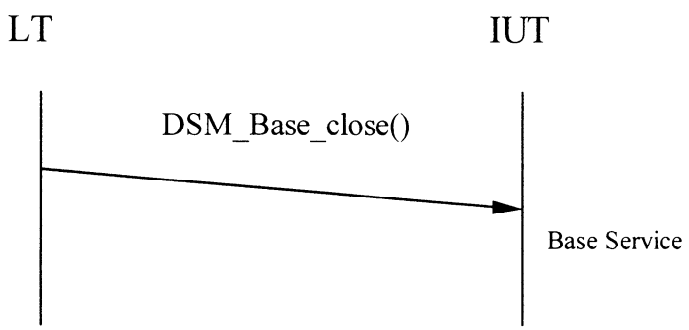
Verify that after getting an object reference of a Service Object Implementation using 'DSM-CC Directory resolve' operation, when the LT sends a 'DSM-CC Base close' message, a persistent Object Reference is deleted.

Test Preamble:

Get an object reference of a Service Object Implementation using 'DSM-CC Directory resolve' operation.

Test Procedure:

1) LT sends a 'DSM-CC Base close' message.



Test Verdict:

Pass the test if the object reference is deleted, therefore the LT can't communicate with the object any more.

5.2.3.1.2 Test Case 2 - Stopping a Stream

5.2.3.1.2.1 Case (2-1)

Test Purpose:

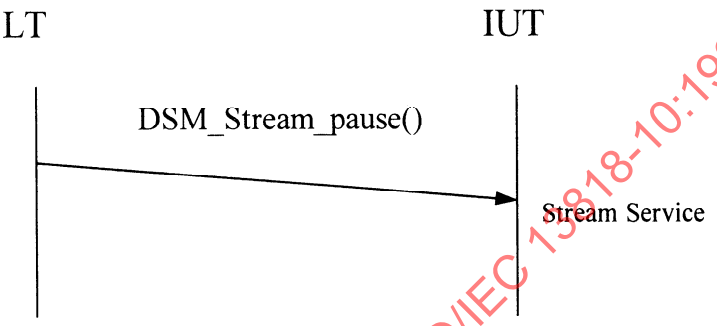
Verify that after initiating a stream, when the LT sends 'DSM-CC Stream pause' message, a stream is stopped.

Test Preamble:

Play a stream (that is, the Stream State Machine is in Transport_Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream pause' message.



Test Verdict:

Pass the test if the stream is stopped at the rStop and the Stream State Machine is in Transport_Pause_Mode, or the exception is raised, if there is any.

5.2.3.1.2.2 Case (2-2)

Test Purpose:

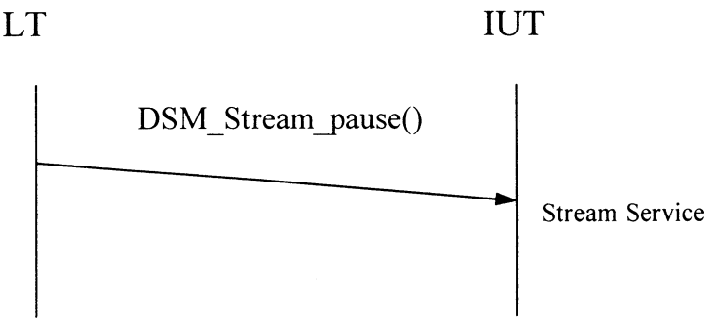
Verify that when the Stream State Machine is in Transport_Pause_Mode and the LT sends 'DSM-CC Stream pause' message, a stream is stopped.

Test Preamble:

Test Case 2-1 (that is, the Stream State Machine is in Transport_Pause_Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream pause' message.



Test Verdict:

Pass the test if the Stream State Machine remains in Transport_Pause_Mode and the rStop value is updated to the new rStop, or the exception is raised, if there is any.

5.2.3.1.2.3 Case (2-3)

Test Purpose:

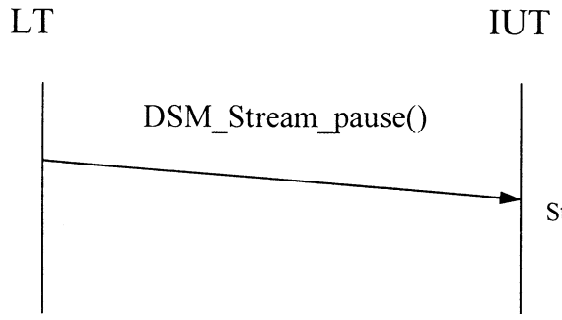
Verify that when the Stream State Machine is in Search_Transport_Pause_Mode or Search_Transport_Mode and the LT sends 'DSM-CC Stream pause' message, a stream is stopped.

Test Preamble:

The Stream State Machine is in Search_Transport_Pause_Mode or Search_Transport_Mode.

Test Procedure:

1) LT sends a 'DSM-CC Stream pause' message.



Test Verdict:

Pass the test if the Stream is stopped at the rStop and the Stream State Machine has transition to Search_Transport_Pause_Mode, or the exception is raised, if there is any.

5.2.3.1.2.4 Case (2-4)

Test Purpose:

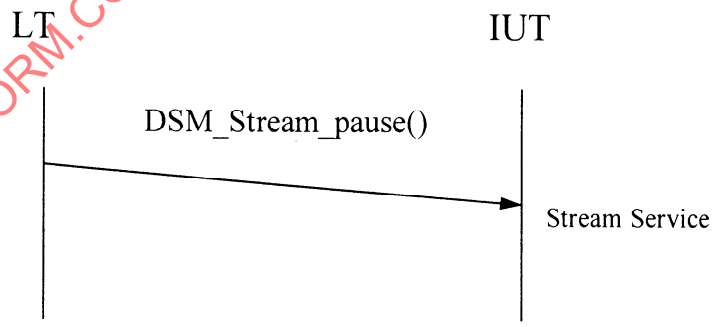
Verify that when the Stream State Machine is in Pause_Search_Transport_Mode and the LT sends 'DSM-CC Stream pause' message, a stream is stopped.

Test Preamble:

The Stream State Machine is in Pause_Search_Transport_Mode.

Test Procedure:

1) LT sends a 'DSM-CC Stream pause' message.



Test Verdict:

Pass the test if the Stream is stopped at the rStop and the Stream State Machine has transition to Transport_Pause_Mode, or the exception is raised, if there is any.

5.2.3.1.3 Test Case 3 - Resuming a Stream

5.2.3.1.3.1 Case (3-1)

Test Purpose:

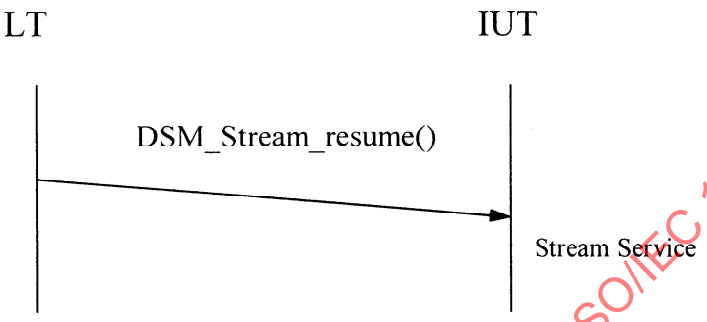
Verify that when the Stream State Machine is in Open_Mode and the LT sends 'DSM-CC Stream resume' message, a stream is started.

Test Preamble:

Establish a session between the SUT and the LT and get an object reference of a given stream (that is, the Stream State Machine is in Open Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream resume' message.



Test Verdict:

Pass the test if the correct MPEG-2 stream is received at the rStart and the Stream State Machine is first in Search_Transport_Mode and then in Transport_Mode, or the exception is raised, if there is any.

5.2.3.1.3.2 Case (3-2)

Test Purpose:

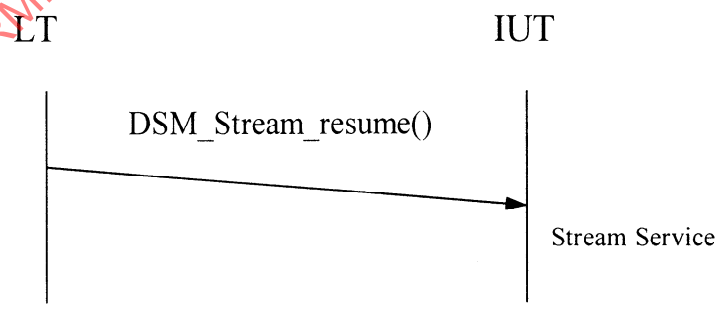
Verify that when the Stream State Machine is in Search_Transport_Mode, Pause_Mode, or Search_Transport_Pause_Mode and the LT sends 'DSM-CC Stream resume' message, a stream is started.

Test Preamble:

The Stream State Machine is in Search_Transport_Mode, Pause_Mode, or Search_Transport_Pause_Mode.

Test Procedure:

1) LT sends a 'DSM-CC Stream resume' message.



Test Verdict:

Pass the test if the Stream State Machine has transition to Search_Transport_Mode immediately and the stream is started when AppNPT arrives at the rStart and the Stream State Machine has transition to Transport_Mode, or the exception is raised, if there is any.

5.2.3.1.3.3 Case (3-3)

Test Purpose:

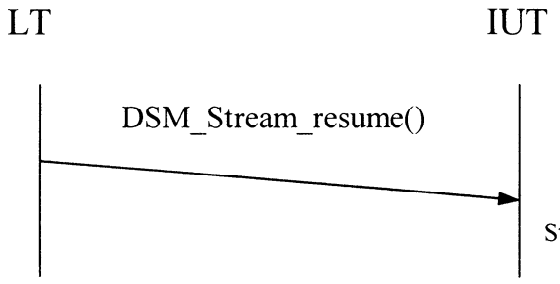
Verify that when the Stream State Machine is in Transport_Pause_Mode and the LT sends 'DSM-CC Stream resume' message, a stream is started.

Test Preamble:

Stop a stream (that is, the Stream State Machine is in Transport_Pause_Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream resume' message.



Test Verdict:

Pass the test if the stream is started at the new rStart and the Stream State Machine is first in Pause_Search_Transport_Mode and then in Search_Transport_Mode when AppNPT arrives at rStop and finally in Transport_Mode, or the exception is raised, if there is any.

5.2.3.1.3.4 Case (3-4)

Test Purpose:

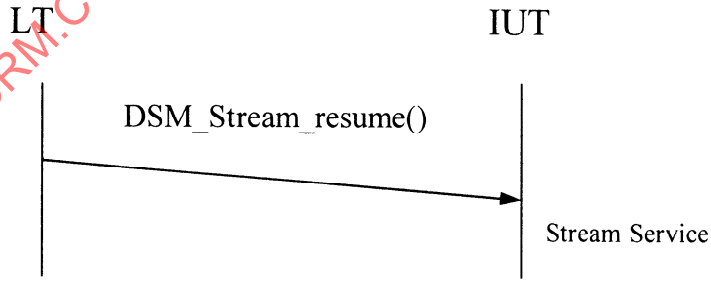
Verify that when the Stream State Machine is in Transport_Mode and the LT sends 'DSM-CC Stream resume' message, a stream is started.

Test Preamble:

Starts a stream (that is, the Stream State Machine is in Transport_Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream resume' message.



Test Verdict:

Pass the test if the stream is started at the new rStart and the Stream State Machine is first in Prc_Search_Transport_Mode and then in Search_Transport_Mode and finally in Transport_Mode, or the exception is raised, if there is any.

5.2.3.1.3.5 Case (3-5)

Test Purpose:

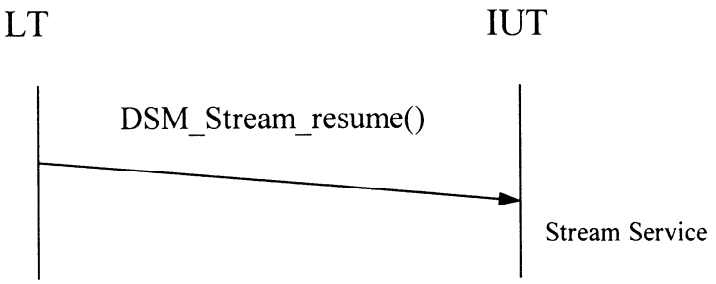
Verify that during the Stream State Machine is in Pause_Search_Transport_Mode when the LT sends 'DSM-CC Stream resume' message, a stream is started.

Test Preamble:

The Stream State Machine is in Pause_Search_Transport_Mode.

Test Procedure:

1) LT sends a 'DSM-CC Stream resume' message.



Test Verdict:

Pass the test if the stream is started at the new rStart and the Stream State Machine remains in Pause_Search_Transport_Mode and then in Search_Transport_Mode when AppNPT arrives at rStop and finally in Transport_Mode, or the exception is raised, if there is any.

5.2.3.1.4 Test Case 4 - Obtaining the status of a Stream

Test Purpose:

Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC Stream status' message, the current state of a stream is sent.

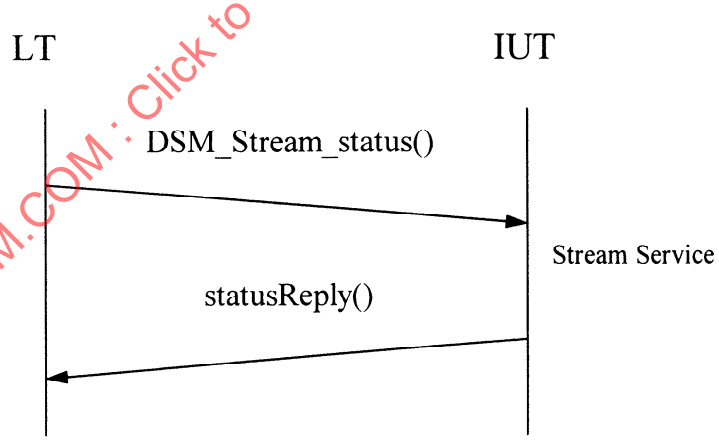
Test Preamble:

Establish a session between the SUT and the LT and get an object reference of a given stream.

Test Procedure:

1) LT sends a 'DSM-CC Stream status' message.

2) LT waits for the current mode of the stream returned from the IUT.



Test Verdict:

Pass the test if the current mode of the stream is received from the IUT. In the case of this test case Open_Mode Stream Status will be returned, or the MPEG_DELIVERY exception is raised, if there is.

5.2.3.1.5 Test Case 5 - Resetting a Stream State Machine

Test Purpose:

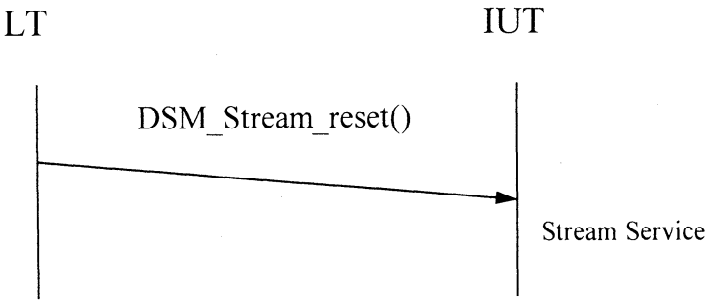
Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC Stream reset' message, the Stream State Machine is to be in Open_Mode.

Test Preamble:

Establish a session between the SUT and the LT.

Test Procedure:

1) LT sends a 'DSM-CC Stream reset' message.



Test Verdict:

Pass the test if the current mode of the Stream State Machine is in Open_Mode.

5.2.3.1.6 Test Case 6 - Jumping a Stream

Test Purpose:

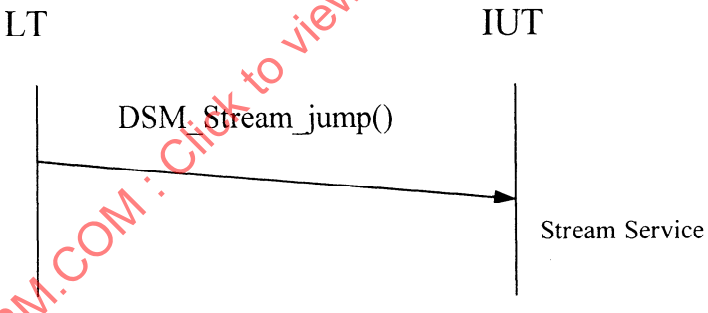
Verify that after playing a stream, when the LT sends 'DSM-CC Stream jump' message, the stream is resumed at the rStart when it reaches the rStop.

Test Preamble:

Play a stream (that is, the Stream State Machine is in Transport_Mode).

Test Procedure:

1) LT sends a 'DSM-CC Stream jump' message.



Test Verdict:

Pass the test if the stream is resumed at the rStart when the stream reaches the rStop, or the exception is raised, if there is any.

5.2.3.1.7 Test Case 7 - Playing a Stream

Test Purpose:

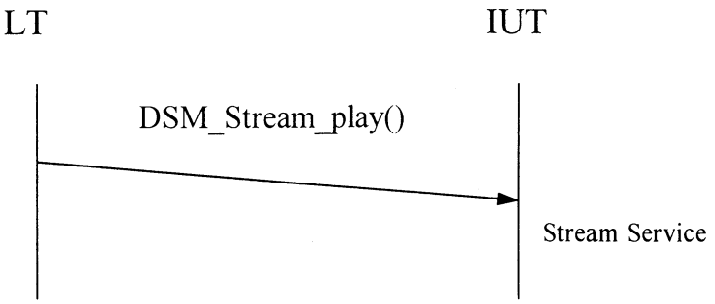
Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC Stream play' message, a stream from the rStart until the rStop is played.

Test Preamble:

Establish a session between the SUT and the LT and get an object reference of a given stream.

Test Procedure:

1) LT sends a 'DSM-CC Stream play' message.



Test Verdict:

Pass the test if the stream is played from the rStart until the rStop and the final Stream State Machine is in Pause_Mode, or the exception is raised, if there is any.

5.2.3.1.8 Test Case 8 - Reading a File

Test Purpose:

Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC File read' message to access to a file, a random access to a file is provided.

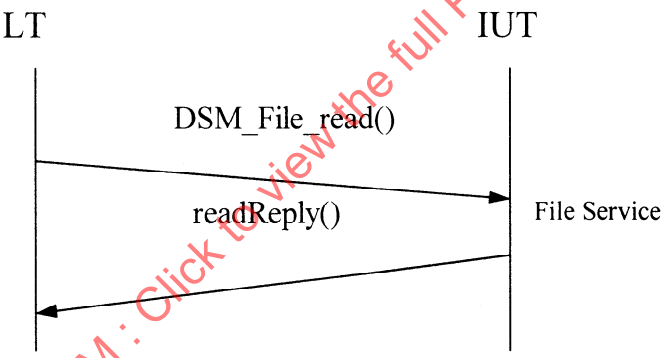
Test Preamble:

Establish a session between the SUT and the LT and get an object reference of a given file.

Test Procedure:

1) LT sends a 'DSM-CC File read' message using the obtained File object reference.

2) LT waits for the file read reply message from the IUT.



Test Verdict:

Pass the test if a random access to an opened file is provided, using the obtained File reference from a previous 'DSM-CC Directory resolve' operation or the exception is raised, if there is any.

5.2.3.1.9 Test Case 9 - Writing a File

Test Purpose:

Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC File write' message, a mechanism to write data to a file starting at a designated offset is provided.

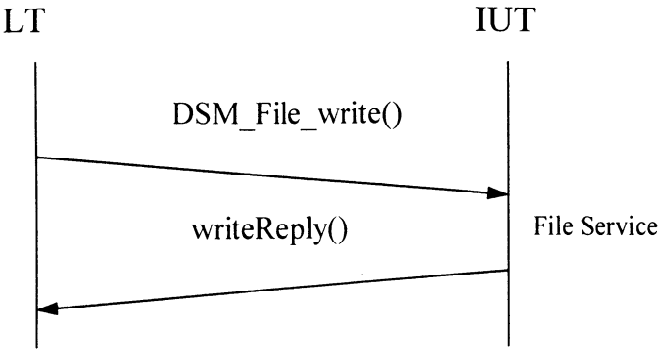
Test Preamble:

Establish a session between the SUT and the LT and get an object reference of a given file.

Test Procedure:

1) LT sends a 'DSM-CC File write' message with the obtained File object reference.

2) LT waits for the file write reply from the IUT.



Test Verdict:

Pass the test if writing data to a file starting at a designated offset is done or the exception is raised, if there is any.

5.2.3.1.10 Test Case 10 - Listing bindings in a Directory

Test Purpose:

Verify that after establishing a session between the SUT and the LT, when the LT sends 'DSM-CC Directory list' message, a list of bindings in a Directory is returned.

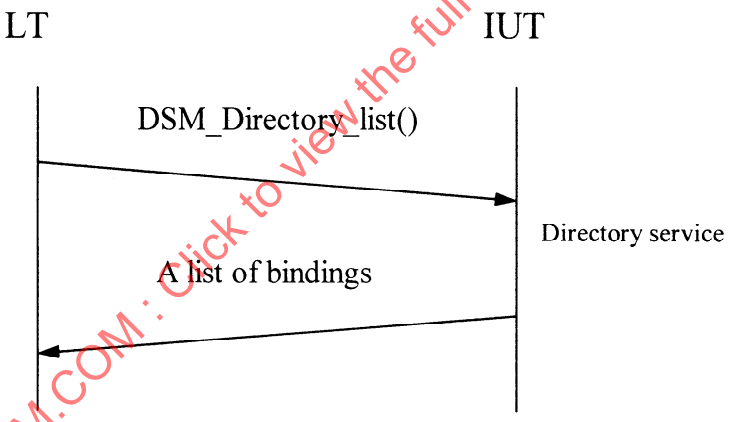
Test Preamble:

Establish a session between the SUT and the LT.

Test Procedure:

1) LT sends a 'DSM-CC Directory list' message.

2) LT waits for the list of bindings in the Directory from the IUT.



Test Verdict:

Pass the test if the requested list of bindings in the Directory is received and in the case that there are any additional bindings, a BindingIterator is returned.

5.2.3.1.11 Test Case 11 - Getting more bindings from a Directory

Test Purpose:

Verify that after listing bindings in the Directory, when the LT sends 'DSM-CC BindingIterator next_one or next_n' message to get more bindings from the Directory, more bindings in a Directory are returned.

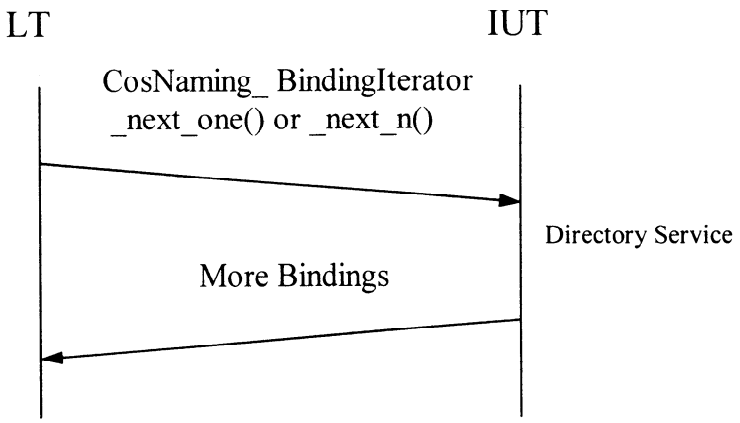
Test Preamble:

List bindings in the Directory (Test Case 10).

Test Procedure:

1) LT sends a 'CosNaming BindingIterator next_one or next_n' message.

2) LT waits for more lists of bindings in the Directory returned from the IUT.



Test Verdict:

Pass the test if more bindings in the Directory are received by LT from IUT in the case that there are any additional bindings or False is returned by IUT in the case that there are no additional bindings.

5.2.3.1.12 Test Case 12 - Discarding any Server-side storage associated with an Iterator

Test Purpose:

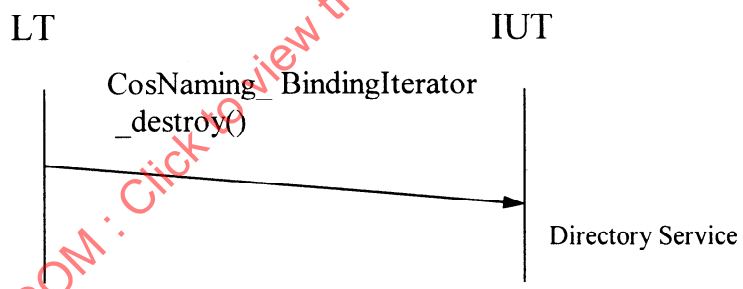
Verify that after listing bindings in a Directory, when the LT sends 'DSM-CC BindingIterator destroy' message, any Server-side storage associated with an Iterator is discarded and the Iterator is no longer valid to access.

Test Preamble:

List bindings in a Directory (Test Case 10).

Test Procedure:

1) LT sends a 'CosNaming BindingIterator destroy' message.



Test Verdict:

Pass the test if any Server-side storage associated with the Iterator is discarded and the Iterator is no longer valid to access.

5.2.3.1.13 Test Case 13 - Resolving an IOR 2

Test Purpose:

Verify that after getting a list of services and assets using 'DSM-CC Directory list()' operation, when the LT sends 'DSM-CC Directory resolve' message, the object reference bound to a given name for a Service Object Implementation instance is returned.

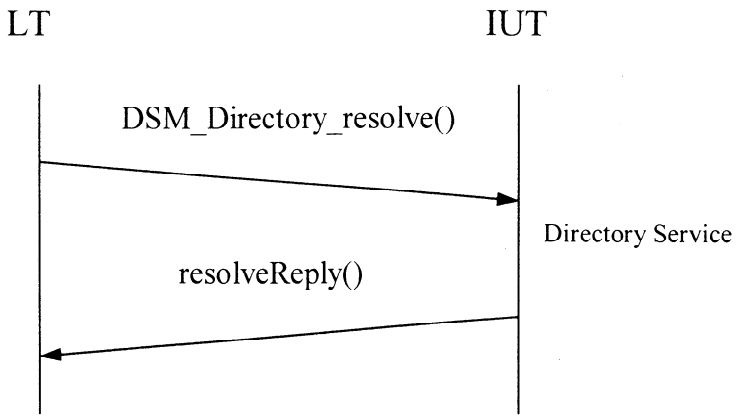
Test Preamble:

Get lists of services and assets using 'DSM-CC Directory list()' operation,

Test Procedure:

1) LT sends a 'DSM-CC Directory resolve' message.

2) LT waits for the object reference bound to a given name.



Test Verdict:

Pass the test if the object reference bound to a given name is received or the exception is raised, if there is any.

5.2.3.1.14 Test Case 14 - Opening a Directory

Test Purpose:

Verify that after getting lists of services and assets using 'DSM-CC Directory list()' operation, when the LT sends 'DSM-CC Directory open' message to find the objects associated with the names in the given path, a resolve of object references from names at the specified nodes in the path are provided.

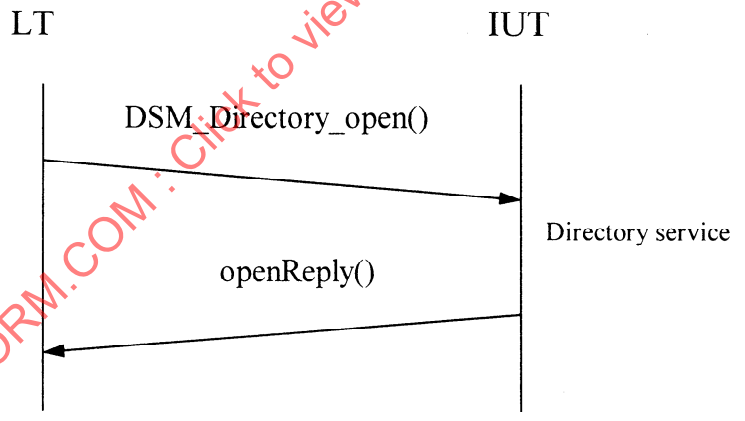
Test Preamble:

List bindings in the directory of services using DSM_Directory_list().

Test Procedure:

1) LT sends a 'DSM-CC Directory open' message.

2) LT waits for a resolve of object references bound to given names.



Test Verdict:

Pass the test if a number of object references associated with given names are provided or the exception is raised, if there is any.

5.2.3.1.15 Test Case 15 - Closing a reference to a Directory object

Test Purpose:

Verify that after resolving a reference of a Directory using 'DSM-CC Directory resolve()' operation, when the LT sends 'DSM-CC Directory close' message to quit a reference to a directory, a reference to a directory is closed.

Test Preamble:

Resolve a reference of a Directory using 'DSM-CC Directory resolve()' operation.