



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

Q.755.2

(09/97)

SERIE Q: CONMUTACIÓN Y SEÑALIZACIÓN

Especificaciones del sistema de señalización N.º 7 –
Gestión del sistema de señalización N.º 7

**Respondedor de pruebas de capacidades de
transacción**

Recomendación UIT-T Q.755.2

(Anteriormente Recomendación del CCITT)

RECOMENDACIONES DE LA SERIE Q DEL UIT-T
CONMUTACIÓN Y SEÑALIZACIÓN

SEÑALIZACIÓN EN EL SERVICIO MANUAL INTERNACIONAL	Q.1–Q.3
EXPLOTACIÓN INTERNACIONAL SEMIAUTOMÁTICA Y AUTOMÁTICA	Q.4–Q.59
FUNCIONES Y FLUJOS DE INFORMACIÓN PARA SERVICIOS DE LA RDSI	Q.60–Q.99
CLÁUSULAS APLICABLES A TODOS LOS SISTEMAS NORMALIZADOS DEL UIT-T	Q.100–Q.119
ESPECIFICACIONES DE LOS SISTEMAS DE SEÑALIZACIÓN N.º 4 Y N.º 5	Q.120–Q.249
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN N.º 6	Q.250–Q.309
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN R1	Q.310–Q.399
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN R2	Q.400–Q.499
CENTRALES DIGITALES	Q.500–Q.599
INTERFUNCIONAMIENTO DE LOS SISTEMAS DE SEÑALIZACIÓN	Q.600–Q.699
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN N.º 7	Q.700–Q.849
Generalidades	Q.700
Parte transferencia de mensajes	Q.701–Q.709
Parte control de la conexión de señalización	Q.711–Q.719
Parte usuario de telefonía	Q.720–Q.729
Servicios suplementarios de la RDSI	Q.730–Q.739
Parte usuario de datos	Q.740–Q.749
Gestión del sistema de señalización N.º 7	Q.750–Q.759
Parte usuario de la RDSI	Q.760–Q.769
Parte aplicación de capacidades de transacción	Q.770–Q.779
Especificaciones de las pruebas	Q.780–Q.799
Interfaz Q3	Q.800–Q.849
SISTEMA DE SEÑALIZACIÓN DIGITAL DE ABONADO N.º 1	Q.850–Q.999
RED MÓVIL TERRESTRE PÚBLICA	Q.1000–Q.1099
INTERFUNCIONAMIENTO CON SISTEMAS MÓVILES POR SATÉLITE	Q.1100–Q.1199
RED INTELIGENTE	Q.1200–Q.1999
RED DIGITAL DE SERVICIOS INTEGRADOS DE BANDA ANCHA (RDSI-BA)	Q.2000–Q.2999

Para más información, véase la Lista de Recomendaciones del UIT-T.

RECOMENDACIÓN UIT-T Q.755.2

RESPONDEDOR DE PRUEBAS DE CAPACIDADES DE TRANSACCIÓN

Resumen

En la versión de 1993 de la Recomendación Q.755, que se refiere a probadores de protocolos, sólo figuraba el probador MTP. Esa Recomendación ha sido dividida en una serie de Recomendaciones.

La presente Recomendación define el respondedor de pruebas de capacidades de transacción (TC, *transaction capabilities*), y es una añadidura a la Recomendación Q.755 de la versión de 1993.

El respondedor de pruebas de TC que figura en el sistema sometido a prueba permite utilizar sucesiones de pruebas abstractas (ATS, *abstract test suites*) para probar la funcionalidad TC definida de ese sistema con independencia de las funcionalidades empleadas por los usuarios TC realmente presentes.

Orígenes

La Recomendación UIT-T Q.755.2 ha sido preparada por la Comisión de Estudio 11 (1997-2000) del UIT-T y fue aprobada por el procedimiento de la Resolución N.º 1 de la CMNT el 12 de septiembre de 1997.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución N.º 1 de la CMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

NOTA

En esta Recomendación, la expresión "Administración" se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT no ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

© UIT 1998

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

1	Alcance	1
2	Referencias.....	1
3	Símbolos y abreviaturas.....	2
4	Generalidades.....	3
5	Respondedor de pruebas de TC	4
5.1	Arquitectura	4
5.2	ASE del usuario probador de TC.....	6
	5.2.1 Principios generales.....	6
	5.2.2 Operaciones y errores	6
5.3	Protocolo de gestión de las pruebas de TC.....	7
	5.3.1 Unidades de datos del protocolo de gestión de las pruebas.....	7
	5.3.2 Estructura de las instrucciones.....	7
	5.3.3 Sintaxis abstracta	8
	5.3.4 Procedimientos	8
5.4	Operaciones y errores del respondedor de pruebas de TC.....	12
5.5	Unidades de datos del protocolo de gestión de las pruebas.....	14
	Anexo A – Ejemplo de utilización del respondedor de pruebas de TC	16
	Anexo B – Establecimiento de bucles utilizando el respondedor de pruebas de TC.....	19

Recomendación Q.755.2

RESPONDEDOR DE PRUEBAS DE CAPACIDADES DE TRANSACCIÓN

(Ginebra, 1997)

1 Alcance

En la presente Recomendación se define el respondedor de pruebas de capacidades de transacción (TC, *transaction capabilities*), que se ha de utilizar como ayuda para verificar las TC del sistema de señalización N.º 7 de la UIT.

Esta Recomendación se basa en la Recomendación Q.750 en lo que respecta a las consideraciones sobre arquitectura de la relación entre el respondedor de pruebas y la gestión del SS N.º 7 (OMAP) [9].

Se define aquí un respondedor de pruebas sencillo y flexible que permite utilizar sucesiones de pruebas abstractas (ATS, *abstract test suites*) para capacidades de transacción con independencia de los usuarios TC que residen efectivamente en un sistema sometido a prueba (SUT, *system under test*).

No se formula ninguna hipótesis sobre la implementación real de la interfaz entre esta función y las TC.

El disponer de un respondedor de pruebas de TC normalizado tiene las siguientes ventajas:

- permite escribir una única sucesión de pruebas abstractas aplicables a cualquier implementación TC que resida en un sistema en el que también exista un respondedor de pruebas;
- permite probar todas las funcionalidades TC definidas, con independencia del subconjunto de funcionalidades realmente utilizadas por los usuarios TC que están disponibles cuando un equipo se somete a prueba;
- sirve para aislar averías durante las pruebas, ya que la respuesta característica al mensaje o componente TC será independiente de la ejecución propiamente dicha de una operación de usuario TC real;
- permite probar la pila TC antes de entregarla a un cliente para que sustente una o más aplicaciones específicas de usuario TC;
- el respondedor de pruebas puede utilizarse incluso para efectuar pruebas de interfuncionamiento pila a pila independientemente de cualquier aplicación específica de usuario TC.

2 Referencias

Las siguientes Recomendaciones del UIT-T y otras referencias contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y otras referencias son objeto de revisiones por lo que se preconiza que los usuarios de esta Recomendación investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y otras referencias citadas a continuación. Se publica periódicamente una lista de las Recomendaciones UIT-T actualmente vigentes.

- [1] Recomendaciones UIT-T X.680-X.683 (1994), *Tecnología de la información – Notación de sintaxis abstracta uno*.
- [2] Recomendación UIT-T X.690 (1994) | ISO/CEI 8825-1:1995, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación básica, de las reglas de codificación canónica y de las reglas de codificación distinguida*.
- [3] Recomendaciones UIT-T Q.711-Q.714 (1993), *Especificaciones del sistema de señalización N.º 7: Parte control de la conexión de señalización*.
- [4] Recomendaciones Q.771-Q.775 del CCITT (1988), *Especificaciones del sistema de señalización N.º 7: Parte aplicación de capacidades de transacción (PACT)*.
- [5] Recomendaciones UIT-T Q.771-Q.775 (1993), *Descripción funcional de capacidades de transacción*.
- [6] Recomendación UIT-T X.290 (1995), *Metodología y marco de las pruebas de conformidad de interconexión de sistemas abiertos de las Recomendaciones sobre los protocolos para aplicaciones del UIT-T – Conceptos generales*.
ISO | CEI 9646-1:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts*.
- [7] Recomendación UIT-T I.320 (1993), *Modelo de referencia de protocolo de la red digital de servicios integrados*.
- [8] Recomendación I.321 del CCITT (1991), *Modelo de referencia de protocolo de la RDSI-BA y su aplicación*.
- [9] Recomendación UIT-T Q.750 (1993), *Visión de conjunto de la gestión del sistema de señalización N.º 7*.

3 Símbolos y abreviaturas

En esta Recomendación se utilizan las siguientes siglas.

AE	Entidad de aplicación (<i>application entity</i>)
APDU	Unidad de datos del protocolo de aplicación (<i>application protocol data unit</i>)
ASE	Elemento de servicio de aplicación (<i>application service element</i>)
ASN.1	Notación de sintaxis abstracta uno (<i>abstract syntax notation one</i>)
ATS	Sucesión de pruebas abstractas (<i>abstract test suite</i>)
DPC	Código de punto de destino (<i>destination PC</i>)
GPC	Código de punto de generador (<i>generator PC</i>)
IUT	Implementación sometida a prueba (<i>implementation under test</i>)
LME	Entidad de gestión de nivel (<i>level management entity</i>)
LT	Probador inferior (<i>lower tester</i>)
MIB	Base de información de gestión (<i>management information base</i>)
MSU	Unidad de señalización de mensaje (<i>message signal unit</i>)
MT	Probador MTP (<i>MTP tester</i>)
MTP	Parte transferencia de mensaje (SS N.º 7) [(SS No. 7) <i>message transfer part</i>]

OPC	Código de punto de origen (<i>originating PC</i>)
PC	Código de punto (<i>point code</i>)
PDU	Unidad de datos de protocolo (<i>protocol data unit</i>)
SAP	Punto de acceso al servicio (<i>service access point</i>)
SCCP	Parte control de la conexión de señalización (SS N.º 7) [(SS No. 7) <i>signalling connection control part</i>]
SP	Punto de señalización (<i>signalling point</i>)
SS N.º 7	Sistema de señalización N.º 7 (<i>signalling system No. 7</i>)
SUT	Sistema sometido a prueba (<i>system under test</i>)
TC	Capacidades de transacción (<i>transaction capabilities</i>)
TMP	Protocolo de gestión de las pruebas (<i>test management protocol</i>)
TPC	Código de punto de reversión (<i>turn-around PC</i>)
TTCN	Notación combinada arborescente y tabular (<i>tree and tabular combined notation</i>)
UT	Probador superior (<i>upper tester</i>)

4 Generalidades

Los probadores de protocolos pueden utilizarse como ayuda para verificar las capacidades de transacción (TC) del sistema de señalización N.º 7 (SS N.º 7, *signalling system No. 7*), sea al realizar una prueba de validación de una implementación o una prueba de compatibilidad entre dos implementaciones. La función principal del probador es la simulación de una parte o aplicación de usuario común, como puede verse a partir de las TC para la generación de tráfico de prueba.

En las Recomendaciones I.320 [7] e I.321 [8] se especifica el modelo de referencia de protocolo RDSI que debe utilizarse para la RDSI de banda estrecha (RDSI-BE) y la RDSI de banda ancha (RDSI-BA). Se identifican el plano de usuario (plano U), el plano de control (plano C) y el plano de gestión (plano M). Los principios de estratificación de capas se aplican en cada uno de estos planos. El plano U proporciona la transferencia del flujo de información de usuario con los controles conexos. El plano C trata la información sobre la llamada y el control de la conexión. El plano M se divide en dos partes, a saber: las funciones de gestión de capa y las funciones de gestión de plano. La gestión de plano desempeña funciones de gestión relacionadas con un sistema considerado en su conjunto, coordina todos los planos y no tiene estructura de capas. La gestión de capa contiene entidades de gestión de capa (LME, *layer management entities*). Cada uno de estos planos ofrece funciones de gestión relacionadas con recursos y parámetros ubicados en su propia entidad de protocolo. La gestión de capa trata los flujos de información sobre operaciones y mantenimiento. La interfaz entre las capas adyacentes dentro de un plano y entre las LME y su capa conexas debe definirse en términos de primitivas de servicio. La interfaz entre las LME y el plano de gestión de plano no tiene que especificarse y depende de la aplicación.

Para el SS N.º 7, la **entidad de gestión de nivel** se define por analogía con la LME que figura en las Recomendaciones I.320 e I.321. Esto se hace para tener en cuenta las distintas posiciones de los límites entre los niveles inferiores del SS N.º 7 y los de la OSI (por ejemplo, la parte superior de la MTP es el nivel 3 en el SS N.º 7 y la SCCP es el nivel 4, pero ambos estarán dentro de la capa 3 si se aplica estrictamente el modelo de la OSI). Para el SS N.º 7, la sigla "LME" significa "entidad de gestión de nivel".

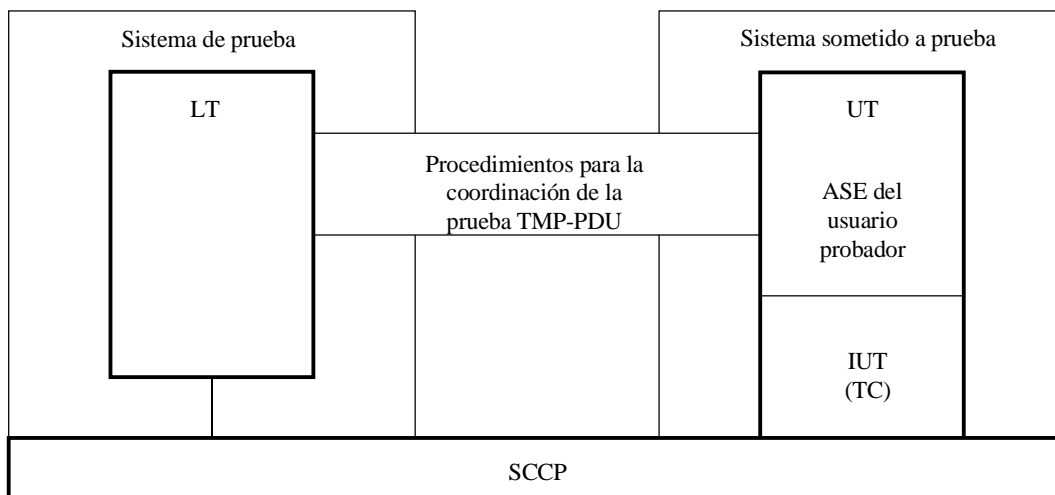
La comunicación entre el respondedor de pruebas de TC ubicado en un sistema sometido a prueba (SUT) y un sistema de prueba depende de la utilización de un ASE particular, denominado ASE del "usuario probador de TC". El respondedor de pruebas de TC desempeña la función de suministrador del ASE mientras que el sistema de prueba desempeña la función de consumidor del ASE.

El ASE del usuario probador de TC puede implementarse como el único componente de una entidad de aplicación (AE, *application entity*) con un número de subsistema particular o puede combinarse con otros elementos de la capa de aplicación, en cuyo caso se selecciona utilizando cualquier nombre de contexto de aplicación que comience con el valor siguiente:

{itu-t recommendation q 755 ac(5)}

La SCCP [3], asigna un número de subsistema que puede utilizarse para el direccionamiento de una AE que contiene el respondedor de prueba (por ejemplo, cuando sólo están disponibles las facilidades TC indicadas en el *Libro Azul*).

Según la metodología de las pruebas de conformidad de la OSI [6], el conjunto de unidades de datos transmitidos entre las instancias del respondedor de pruebas y el sistema de prueba pueden considerarse como un protocolo de gestión de las pruebas (TMP, *test management protocol*) "dentro de banda", que puede utilizarse para sustentar métodos de prueba coordinados entre el respondedor de prueba que actúa como un probador superior (UT, *upper tester*) y la funcionalidad de probador inferior (LT, *lower tester*) del sistema de prueba (véase la figura 2).



T1185390-97

Figura 2/Q.755.2 – Disposición de la arquitectura de prueba

El uso de este respondedor de pruebas no requiere ningún acceso a la interfaz de servicio TC en el SUT, ni impone ninguna restricción a su implementación. El respondedor de pruebas no aporta medios de comprobación del comportamiento de una implementación TC en respuesta al comportamiento no válido del usuario TC local.

El carácter "dentro de banda" del TMP implica que, aunque está sometido a prueba, el servicio TC es lo bastante fiable como para transportar esas PDU y entregarlas a un usuario. Para superar las posibles dificultades debidas a un servicio TC faltante o no fiable, los procedimientos del ASE del usuario probador se definen de modo que cada TMP-PDU pueda recibirse en distintos tipos de mensajes o componentes.

El objetivo principal del TMP es permitir el envío de una serie de instrucciones del sistema de prueba al respondedor de pruebas a fin de provocar algún comportamiento en la implementación sometida a prueba (IUT, *implementation under test*) de las TC. Cada instrucción es una primitiva del servicio TC que el respondedor de pruebas debe pasar a las TC sometidas a prueba o bien una indicación de que el respondedor de pruebas debe esperar un evento ulterior.

5.2 ASE del usuario probador de TC

5.2.1 Principios generales

El ASE del usuario probador de TC puede construirse en base a las versiones de 1988 [4] o de 1993 [5] de las TC¹. Este ASE tiene la posibilidad de tratar simultáneamente varios diálogos. Comprende el conocimiento de siete operaciones, dos de las cuales pueden ser invocadas por el consumidor del ASE (es decir, el sistema de prueba), y otras cinco pueden ser invocadas por el suministrador del ASE (es decir, el sistema sometido a prueba). Las dos operaciones invocadas por el respondedor de pruebas se definen como operaciones de clase 1. Sin embargo, esto no tiene consecuencias en el comportamiento del respondedor de pruebas².

Las unidades de datos del protocolo de gestión de las pruebas (TMP-PDU, *test management protocol data units*) pueden transportarse en los argumentos de las operaciones, en los parámetros resultado de las operaciones, en los parámetros error y, cuando está disponible, en el parámetro información de usuario de la porción de diálogo.

No existe una relación específica entre una TMP-PDU y el tipo de mensaje o componente utilizado para llevarla.

NOTA – La interfaz de servicio superior del ASE del usuario probador de TC no está normalizada. No obstante, esa interfaz podría hacerse accesible localmente para provocar un determinado escenario de prueba (por ejemplo, en las pruebas de interfuncionamiento) o habilitar al respondedor de pruebas para que comunique a alguna función de gestión la expiración del temporizador T-prueba.

5.2.2 Operaciones y errores

El módulo ASN.1 usuario probador de TC definido en 5.4 contiene la especificación de las operaciones que pueden ser invocadas por el sistema de prueba o el respondedor de pruebas durante la comunicación.

Los valores locales asignados a las siguientes operaciones y errores se consideran valores por defecto. Los valores locales reales utilizados para estas operaciones y errores deberían tratarse como parámetros de configuración.

class1SupplierOperation
class2SupplierOperation
class3SupplierOperation
class4SupplierOperation
localConsumerError
localSupplierError

¹ Corresponde al diseñador de la sucesión de pruebas preparar los casos de prueba de tal modo que el respondedor de pruebas no pida funcionalidades de la versión de 1993 a una implementación TC de 1988.

² El hecho de que el respondedor de prueba comunique o no un resultado de estas operaciones depende de las instrucciones recibidas en las TMP-PDU. Esto no infringe ninguna regla por lo que se refiere a las TC dado que la clase de una operación no afecta a las TC ubicadas en el lado en que se ejecuta la operación.

5.3 Protocolo de gestión de las pruebas de TC

5.3.1 Unidades de datos del protocolo de gestión de las pruebas

Existen tres tipos de unidades de datos del protocolo de gestión de las pruebas (TMP-PDU)³:

Test Init (Iniciación de la prueba)

La PDU testInit pide al respondedor de pruebas que inicie una sesión de prueba y transporta un conjunto de instrucciones que el respondedor de pruebas debe ejecutar secuencialmente. Esta PDU sólo puede ser enviada por el sistema de prueba.

Test Continue (Continuación de la prueba)

La PDU testContinue transporta un conjunto de instrucciones adicionales que el respondedor de pruebas debe ejecutar secuencialmente. Esta PDU sólo puede ser enviada por el sistema de prueba.

Test Data Echo (Eco de los datos de la prueba)

La PDU testDataEcho transmite los datos de usuario reproduciendo en eco los datos de usuario recibidos con una determinada instrucción. Esta PDU sólo puede ser enviada por el respondedor de pruebas.

Las PDU testInit y testContinue pueden transportarse en el argumento de las operaciones invocados por el sistema de prueba, en el parámetro resultado o el parámetro error devuelto por el sistema de prueba en respuesta a una operación invocada por el respondedor de pruebas o en el parámetro información de usuario de la porción de diálogo de los mensajes enviados por el sistema de prueba.

La PDU testDataEcho puede transportarse en el argumento de las operaciones invocadas por el respondedor de pruebas, en el parámetro resultado o en el parámetro error devuelto por el sistema sometido a prueba en respuesta a una operación invocada por el sistema de prueba o en el parámetro información de usuario de la porción de diálogo de los mensajes enviados por el respondedor de pruebas.

5.3.2 Estructura de las instrucciones

Cada instrucción enviada desde el sistema de prueba en una PDU testInit o test Continue indica al respondedor de pruebas que debe esperar un evento externo de las TC locales o que debe enviar una determinada primitiva de petición a las TC locales.

Las instrucciones que indican al respondedor de pruebas que tiene que esperar un evento externo indican también si dicho evento deberá producirse en un diálogo particular o si eso no tiene importancia.

Las instrucciones dadas al respondedor de pruebas para que emita una primitiva comprenden hasta tres elementos:

- 1) una indicación del tipo de primitiva de petición que debe pasar a las TC locales;
- 2) la referencia al diálogo en el que debería efectuarse la acción. Si esta información no se da de manera explícita, el respondedor de pruebas supone que la instrucción se refiere al diálogo en el que se ha recibido la TMP-PDU. El sistema de prueba elige siempre la referencia al diálogo. A diferencia de los identificadores (Id) de transacción y de diálogo, la referencia al diálogo es una referencia común compartida por ambas partes de un diálogo.

³ La necesidad de que exista una PDU "Fin de prueba" queda en estudio.

El respondedor de pruebas debe mantenerse al tanto de la relación biunívoca que existe entre una referencia al diálogo y el Id de diálogo de las TC locales convenida entre las TC y el ASE del usuario probador para ese diálogo;

- 3) facultativamente, un parámetro con los datos de usuario que se han de reproducir en eco como datos de usuario asociados a la primitiva de servicio que ha de enviarse.

5.3.3 Sintaxis abstracta

El módulo ASN.1 TC-TMP definido en 5.5 contiene la especificación de las unidades de datos del protocolo de gestión de las pruebas.

Cuando las TMP-PDU se transportan en el parámetro información de usuario de la porción de diálogo se utiliza el siguiente nombre de sintaxis abstracta como una referencia directa para identificar el conjunto de valores de datos, cada uno de los cuales es un valor del tipo

TMP-Protocol.TMP-PDU:

`{itu-t recommendation q 755 as(4) tmp-pdus (1) version1(1)}`

Las reglas de codificación aplicables son las reglas de codificación básicas definidas en la Recomendación X.690 [2].

5.3.4 Procedimientos

5.3.4.1 Procedimiento en el lado sistema de prueba

El ASE del usuario probador tiene capacidades limitadas de detección de errores y recuperación tras error. El diseñador de la sucesión de pruebas abstractas debe asegurar que la secuencia de las instrucciones enviadas al respondedor de pruebas corresponde a un comportamiento válido del usuario TC y no plantea conflictos con los procedimientos automáticos de este ASE.

Corresponde al diseñador de la sucesión de pruebas elegir la forma en que se transporta una TMP-PDU al respondedor de pruebas (es decir, qué mensaje y qué componente, en su caso, etc.). El sistema de prueba puede enviar una TMP-PDU en cualquier tipo de mensaje y componente con la única limitación de que una PDU testInit sólo puede ser enviada en un mensaje comienzo o en un mensaje unidireccional.

Corresponde también al diseñador de la sucesión de pruebas abstractas asegurar que la TMP-PDU se entrega al ASE del usuario probador (es decir, que debería transmitirse en un mensaje o componente TC válido).

Comienzo de un caso de prueba: Al comenzar cada caso de prueba, el sistema de prueba envía una PDU testInit para asegurarse de que no sigue activo ningún recurso asociado a un caso de prueba anterior. Esta PDU incluye una secuencia de instrucciones que debe ejecutar el respondedor de pruebas y, facultativamente, el valor de un temporizador de seguridad (T-prueba). Si no se indica ningún valor de temporizador, se selecciona un valor de la implementación⁴.

Continuación de un caso de prueba: Si no se envían todas las instrucciones en la PDU testInit, el sistema de prueba puede enviar más instrucciones al respondedor de pruebas utilizando la PDU testContinue que contiene también una secuencia de instrucciones que debe ejecutar el respondedor de pruebas. Un mensaje TC puede transportar más de una PDU testContinue.

⁴ Los implementadores de las pruebas pueden elegir un valor muy amplio si el respondedor de pruebas se utiliza para cursar tráfico de fondo o para efectuar una prueba de la carga.

Finalización de un caso de prueba: No existe ninguna instrucción específica ni ninguna TMP-PDU para finalizar un caso de prueba. Corresponde al diseñador de la prueba escribir el epílogo del caso de prueba de forma que queden cerrados todos los diálogos activos.

5.3.4.2 Procedimiento en el lado respondedor de pruebas

5.3.4.2.1 Reglas genéricas

El respondedor de pruebas rechaza cualquier nombre de contexto de aplicación cuyo valor de identificador de objeto no comience con la siguiente raíz:

{itu-t recommendation q 755 ac(5)}

Cuando el respondedor de pruebas acepta un diálogo de la versión 1993 (es decir, un diálogo conforme a [5]), utiliza el nombre de contexto de aplicación recibido en el diálogo siguiente que trata la primitiva que él emite.

Cuando el respondedor de pruebas rechaza un diálogo de la versión 1993, propone el siguiente nombre de contexto de aplicación:

{itu-t recommendation q 755 ac(5) testing-ac (1) version1 (1)}

Cuando pide a las TC que envíen un mensaje comienzo o un mensaje unidireccional, el respondedor de pruebas utiliza como dirección de destino la dirección de origen del mensaje en el que se recibió la PDU testInit. Si el respondedor de pruebas tiene que proporcionar un nombre de contexto de aplicación, utiliza siempre el siguiente valor:

{itu-t recommendation q 755 ac(5) testing-ac (1) version1 (1)}

Si el respondedor de pruebas detecta que ha sido alcanzado con una dirección de destino que no es la que conoce a partir de los datos de configuración, obliga a las TC a insertar la dirección que él conoce en el primer mensaje continuación hacia atrás enviado al sistema de prueba.

Durante un diálogo, se aplican las siguientes reglas:

- si el primer Id (identificador) de invocación utilizado por el contestador de prueba tiene el valor 0, se incrementa este valor cada vez que el respondedor de pruebas invoca una operación;
- si la invocación recibida por el respondedor de pruebas hace que se produzca un rechazo de usuario, el respondedor de pruebas pide automáticamente a las TC que devuelvan un componente rechazo (por ejemplo, rechaza cualquier operación que no esté definida en el módulo usuario probador de TC) con el problema de rechazo adecuado;
- el respondedor de pruebas pide siempre la opción rechazo al enviar una primitiva de petición TC-COMIENZO. En los demás casos, la fijación de este parámetro depende de la implementación;
- al pedir el envío de resultados parciales, el respondedor de pruebas solicita siempre la opción secuenciación. En los demás casos, la fijación de este parámetro depende de la implementación;
- el valor del temporizador proporcionado por el respondedor de pruebas al invocar una operación es un parámetro de la configuración.

5.3.4.2.2 Tratamiento de las TMP-PDU

Si se recibe un evento sin ninguna TMP-PDU, el respondedor de pruebas no efectúa ninguna operación⁵.

Si se recibe un evento con varias TMP-PDU, éstas se procesan secuencialmente comenzando con las que están incluidas en el campo información de usuario de la porción de diálogo (si existe).

Al recibir una PDU testInit, el respondedor de pruebas libera todos los recursos activos (por ejemplo, cierra cualquier diálogo activo, libera las referencias a diálogos, etc.)⁶ y arranca el temporizador T-prueba. A continuación, las instrucciones se ejecutan secuencialmente y el respondedor de pruebas espera el próximo evento o la expiración del temporizador T-prueba.

Al recibir una PDU testContinue, el respondedor de pruebas ejecuta secuencialmente las instrucciones específicas y espera el próximo evento o la expiración del temporizador T-prueba.

Al recibir una PDU testDataEcho, el respondedor de pruebas no efectúa ninguna operación.

Según las reglas genéricas, la recepción de cualquier otro valor de datos llevará a:

- un procedimiento de rechazo por el usuario TC, si el valor de PDU no válido se recibe en un argumento de operación, en un parámetro resultado o en un parámetro error;
- un procedimiento de aborto por el usuario TC, si se recibe en el campo de información de usuario de la porción de diálogo.

Cuando el temporizador T-prueba expira, todos los recursos asociados a la prueba son liberados automáticamente.

El respondedor de pruebas interpreta y ejecuta secuencialmente cada instrucción recibida del sistema de prueba, como se describe en 5.3.4.2.3 y 5.3.4.2.4.

5.3.4.2.3 Ejecución de la instrucción de espera

El respondedor de pruebas espera un evento entrante de la IUT. Una vez recibido dicho evento, ejecuta la próxima instrucción (si hay alguna) o espera la TMP-PDU siguiente⁷.

Conceptualmente, un evento entrante corresponde a una o más primitivas de indicación relacionadas con las TC. El segundo caso corresponde a una situación en la que el respondedor de pruebas recibe una primitiva de tratamiento de diálogo que indica la presencia de componentes. En tal caso, el respondedor de pruebas consume todas las primitivas de tratamiento de componentes hasta que el parámetro "último componente" toma el valor "VERDADERO". No obstante, esto no impone ninguna restricción a la implementación real de la interfaz entre el respondedor de pruebas y las TC.

Si la instrucción va acompañada de una referencia a diálogo explícita, se ignoran las primitivas cuyo identificador de diálogo no corresponde a esta referencia (el respondedor de pruebas espera el evento siguiente). En los demás casos, no se verifica el valor del identificador de diálogo.

⁵ Esto incluye la recepción de una primitiva de indicación TC-NOTIFICACIÓN que puede contener una TMP-PDU devuelta.

⁶ La forma en que se liberan los recursos es una cuestión que concierne a la implementación. No obstante, no debería provocar el envío de ningún mensaje externo (por ejemplo, mensaje FINALIZACIÓN o ABORTO).

⁷ El respondedor de pruebas espera siempre eventos. La instrucción de espera sólo se requiere cuando es preciso esperar explícitamente un evento antes de ejecutar una instrucción que ha sido enviada previamente.

5.3.4.2.4 Ejecución de otras instrucciones

El respondedor de pruebas interpreta el tipo de servicio del modo siguiente:

- v1988uniReq: El respondedor de pruebas pide a la IUT que envíe un mensaje unidireccional de la versión 1988.
- v1993uniReq: El respondedor de pruebas pide a la IUT que envíe un mensaje unidireccional de la versión 1993.
- v1988beginReq: El respondedor de pruebas pide a la IUT que envíe un mensaje comienzo de la versión 1988.
- v1993beginReq: El respondedor de pruebas pide a la IUT que envíe un mensaje comienzo de la versión 1993 que incluya una porción de diálogo.
- continueReq: El respondedor de pruebas pide a la IUT que envíe un mensaje continuación para el diálogo que corresponde al valor de la referencia al diálogo incluido en la instrucción.
- basicEndReq: El respondedor de pruebas pide a la IUT que envíe un mensaje finalización para el diálogo que corresponde al valor de la referencia al diálogo incluido en la instrucción.
- localEndReq: El respondedor de pruebas pide a la IUT que termine localmente el diálogo que corresponde al valor de la referencia al diálogo incluido en la instrucción.
- uAbortReq: El respondedor de pruebas pide a la IUT que aborte el diálogo que corresponde al valor de la referencia al diálogo incluido en la instrucción. Si el diálogo se ha establecido utilizando el procedimiento de la versión 1993, el motivo del aborto se fija en "específico del usuario".
- class1invokeReq: El respondedor de pruebas pide a la IUT que invoque la operación de clase 1 definida en el módulo usuario probador de TC.
- class2invokeReq: El respondedor de pruebas pide a la IUT que invoque la operación de clase 2 definida en el módulo usuario probador de TC.
- class3invokeReq: El respondedor de pruebas pide a la IUT que invoque la operación de clase 3 definida en el módulo usuario probador de TC.
- class4invokeReq: El respondedor de pruebas pide a la IUT que invoque la operación de clase 4 definida en el módulo usuario probador de TC.
- linkedInvokedReq: El respondedor de pruebas pide a la IUT que invoque la operación de clase 1 definida en el módulo usuario probador de TC como operación vinculada a la operación pendiente más antigua.
- resultNReq: El respondedor de pruebas pide a la IUT que envíe un componente no último como resultado en respuesta a la operación pendiente más antigua.
- resultLReq: El respondedor de pruebas pide a la IUT que envíe un componente último como resultado en respuesta a la operación pendiente más antigua.
- uErrorReq: El respondedor de pruebas pide a la IUT que envíe un componente de error en respuesta a la operación pendiente más antigua. El código de error es el único código de error permitido por la definición de la operación.
- uCancelReq: El respondedor de pruebas pide a la IUT que cancele la operación pendiente más antigua.

- `uRejectReq`: El respondedor de pruebas pide a la IUT que envíe un componente de rechazo que indique "problema de invocación: limitación de recursos" para la operación pendiente más antigua.

NOTA 1 – En este contexto, un "mensaje de la versión 1988" es un mensaje definido en [4], un "mensaje de la versión 1993" es un mensaje definido en [5].

NOTA 2 – Para el módulo usuario probador de TC, véase 5.4.

5.3.4.2.5 Eco de datos

Cuando se pide un servicio de tratamiento de diálogo de las TC 1993 (véase [5]), y si está presente un elemento "a reproducir en eco" en la instrucción asociada recibida del sistema de prueba, el respondedor de pruebas incluye una o más TMP-PDU `testDataEcho` (eco de datos de prueba) como valor del parámetro de información de usuario. Cada valor "testDataEcho" es idéntico al valor "toBeEchoed" (a reproducir en eco) recibido. Durante la fase de establecimiento del diálogo o cuando se utiliza el modo diálogo no estructurado, el número de valores "testDataEcho" proporcionados por el respondedor de pruebas es un parámetro de configuración. Una vez establecido el diálogo, sólo se puede enviar un valor.

Cuando se pide un servicio de tratamiento de componente de las TC, el respondedor de pruebas incluye una TMP-PDU como parámetro de usuario (es decir, un argumento de operación, parámetro de resultado, parámetro de error, etc.) si está presente un elemento "a reproducir en eco" en la instrucción asociada recibida del sistema de prueba. Si se incluye, el valor "testDataEcho" es idéntico al valor "toBeEchoed" recibido.

Si el respondedor de pruebas recibe alguna información desconocida en el parámetro información de usuario de una primitiva de tratamiento de diálogo, la entrega inalterada en el parámetro información de usuario de la siguiente primitiva de tratamiento de diálogo que emita.

5.4 Operaciones y errores del respondedor de pruebas de TC

El siguiente módulo define el ASE del usuario probador de TC en función de las operaciones que pueden ser invocadas por el consumidor (sistema de prueba) y por el proveedor (sistema sometido a prueba).

```
TC-Testing-User {itu-t recommendation q 755 modules(0) testing-user(1) version1(1)}
```

```
DEFINITIONS ::=
```

```
BEGIN
```

```
IMPORTS
```

```
OPERATION, ERROR
```

```
FROM TCAPMessages {itu-t recommendation q 773 modules(2) messages(1) version2(2)}
```

```
APPLICATION-SERVICE-ELEMENT
```

```
FROM Notation-Extensions {joint-iso-ccitt remote-operations(4) notation-extension(2)}
```

```
TMP-PDU
```

```
FROM TC-TMP {itu-t recommendation q 755 modules(0) tmp(2) version1(1)}
```

```
;
```

```
-- application-context-names
```

```
ac-id OBJECT IDENTIFIER ::= {itu-t recommendation q 755 ac(5) }
```

```
testing-ac-id OBJECT IDENTIFIER ::= {ac-id testing-ac(1) version1(1)}
```

```

-- ase
Testing-User-ASE APPLICATION-SERVICE-ELEMENT
CONSUMER INVOKES {
    localConsumerOperation,
    globalConsumerOperation
}
-- consumer is the test system
SUPPLIER INVOKES {
    class1SupplierOperation,
    class2SupplierOperation,
    class3SupplierOperation,
    class4SupplierOperation,
    globalSupplierOperation
}
-- supplier is the test responder

::= {itu-t recommendation q 755 ase(3) testing-user(1) version1(1)}

```

```

LocalConsumerOperation ::= OPERATION
ARGUMENT                TMP-PDU
RESULT                  TMP-PDU
ERRORS                  {localSupplierError}

```

```

GlobalConsumerOperation ::= OPERATION
ARGUMENT                TMP-PDU
RESULT                  TMP-PDU
ERRORS                  {globalSupplierError}

```

```

Class1SupplierOperation ::= OPERATION
ARGUMENT                TMP-PDU
RESULT                  TMP-PDU
ERRORS                  {localConsumerError}
LINKED                  {localConsumerOperation}

```

```

Class2SupplierOperation ::= OPERATION
ARGUMENT                TMP-PDU
ERRORS                  {localConsumerError}
LINKED                  {localConsumerOperation}

```

```

Class3SupplierOperation ::= OPERATION
ARGUMENT                TMP-PDU
RESULT                  TMP-PDU

```

```

Class4SupplierOperation ::= OPERATION
ARGUMENT                TMP-PDU

```

```

GlobalSupplierOperation ::= OPERATION
ARGUMENT                TMP-PDU
RESULT                  TMP-PDU
ERRORS                  {globalConsumerError}
LINKED                  {globalConsumerOperation}

```

```

ConsumerError ::= ERROR
PARAMETER          TMP-PDU

```

```

SupplierError ::= ERROR
PARAMETER          TMP-PDU

```

```

localConsumerOperation LocalConsumerOperation ::= localValue : 0

globalConsumerOperation GlobalConsumerOperation ::=
    globalValue : {itu-t recommendation q 755 operations(1) consumer(1)}

class1SupplierOperation Class1SupplierOperation ::= localValue : 1
class2SupplierOperation Class2SupplierOperation ::= localValue : 2
class3SupplierOperation Class3SupplierOperation ::= localValue : 3
class4SupplierOperation Class4SupplierOperation ::= localValue : 4
globalSupplierOperation Class1SupplierOperation ::=
    globalValue : {itu-t recommendation q 755 operations(1) supplier(2)}

localConsumerError ConsumerError ::= localValue : 1
globalConsumerError ConsumerError ::=
    globalValue : {itu-t recommendation q 755 errors(2) consumer(1)}

localSupplierError SupplierError ::= localValue : 2
globalSupplierError SupplierError ::=
    globalValue : {itu-t recommendation q 755 errors(2) supplier(2)}

```

END

5.5 Unidades de datos del protocolo de gestión de las pruebas

El siguiente módulo define las unidades de datos del protocolo de gestión de las pruebas.

```

TC-TMP {itu-t recommendation q 755 modules(0) tmp(2) version1(1)}

```

DEFINITIONS IMPLICIT TAGS ::=

BEGIN

```

TMP-PDU ::= CHOICE
    {
        testInit           [0] TestInit,
        testContinue      [1] CommandSequence,
        testDataEcho     [2] UserData
    }

```

CommandSequence ::= SEQUENCE SIZE(0..maxNbOfCommands) OF TestCommand

maxNbOfCommands INTEGER ::= 30

```

TestInit ::= SEQUENCE
    {
        timeout           INTEGER (1..127) OPTIONAL-- T-Test (unit is 30 sec)
        commands         CommandSequence,
        ...
    }

```

```

UserData ::= CHOICE{
    simple           OCTET STRING (SIZE(0..maxUserDataLength)),
    complex        [0] ABSTRACT-SYNTAX.&Type
}

```

maxUserDataLength INTEGER ::= 2048

```

TestCommand ::= CHOICE
    {
        wait          [0] DialogueReference,
        action        [1] ActionInfo
    }

DialogueReference ::= CHOICE
{
    unspecified      NULL,
    dialogue         INTEGER (0..255)
}

ActionInfo ::= SEQUENCE
{
    service          ServiceType,
    dialogueReference DialogueReference  DEFAULT unspecified : NULL,
    to-be-echoed    UserData  OPTIONAL,
    ...
}

ServiceType ::= ENUMERATED
    {
        v1988uniReq (10),
        v1993uniReq (11),
        v1988beginReq (12),
        v1993beginReq (13),
        continueReq (14),
        basicEndReq (15),
        localEndReq (16),
        uAbortReq (17),
        class1invokeReq (21),
        class2invokeReq (22),
        class3invokeReq (23),
        class4invokeReq (24),
        linkedInvokeReq (25),
        resultNReq (26),
        resultLReq (27),
        uErrorReq (28),
        uCancelReq (29),
        uRejectReq (30),
        ...
    }

-- abstract syntax name for TMP-PDUs

tmp-pdus-as OBJECT IDENTIFIER ::= {itu-t recommendation q 755 as(4) tmp-pdus(1) version1(1)}

END

```

ANEXO A

Ejemplo de utilización del respondedor de pruebas de TC

Este anexo ilustra la forma en que pueden ponerse en aplicación los objetivos de la prueba definidos en la Recomendación Q.787 utilizando el respondedor de pruebas de TC.

a) Prueba 2.1.6: Funciones válidas, cancelación por el usuario

Interfaz respondedor de pruebas/TC	Flujo de mensajes	Sistema de prueba
	COMIENZO [Invocación de (operación de consumidor local, 1)] <-----	
Indicación-TC-Comienzo Indicación TC-Invocación (1)		
Petición-TC-Invocación (0) Petición-TC-Continuación	CONTINUACIÓN [Invocación de (operación de suministrador de clase 1, 0)] ----->	
Petición-TC-U-Cancelación (0)		
	CONTINUACIÓN [devolución de resultado, último (0)] <-----	
Indicación-TC-L-Rechazo (0)		
Petición-TC-Finalización		
	FINALIZACIÓN [Rechazo (0)]	

El argumento localConsumerOperation (operación de consumidor local) se define del modo siguiente:

```

testInit : {
    timeOut 30,
    commands
    {
        action : {service classIInvokeReq},
        action : {service continueReq},
        action : {service uCancelReq},
        wait : {unspecified : NULL},
        action : {service basicEndReq}
    }
}
    
```

b) **Prueba 2.1.2.1.1: Funciones válidas, operaciones vinculadas, operación original de clase 1, IUT como emisor**

Interfaz respondedor de pruebas/TC	Flujo de mensajes	Sistema de prueba
	COMIENZO [Invocación de (operación de consumidor local, 1)] <-----	
Indicación-TC-Comienzo Indicación-TC-Invocación (1)		
Petición-TC-Invocación (0) Petición-TC-Continuación		
	CONTINUACIÓN [Invocación de (operación de suministrador de clase 1, 0)] ----->	
	CONTINUACIÓN [Invocación de (operación de consumidor local, 2)] <-----	
Indicación-TC-Continuación Indicación-TC-Invocación (2)		
TC-Resultado-L (2) Petición-TC-Continuación		
	CONTINUACIÓN [Devolución de resultado, último (2)]	
	FINALIZACIÓN [Devolución de resultado, último (0)]	
Indicación-TC-Finalización TC-Resultado-L (0)		

El primer argumento de invocación localConsumerOperation (operación de consumidor local) se define del modo siguiente:

```

testInit : {
    timeOut 30,
    commands {
        action : {service classInvokeReq},
        action : {service continueReq},
        wait : {unspecified : NULL}
    }
}

```

El segundo argumento de invocación localConsumerOperation (operación de consumidor local) se define del modo siguiente:

```
testContinue : {
    action : {service resultLReq},
    action : {service continueReq},
    wait   : {unspecified : NULL},
}
}
```

c) Prueba 1.1.2.2.1.1-3: Funciones válidas, liberación después del mensaje de continuación, aborto de la IUT por el usuario TR

Interfaz respondedor de pruebas/TC	Flujo de mensajes	Sistema de prueba
	COMIENZO [Invocación de (operación de consumidor local, 1)] <-----	
Indicación-TC-Comienzo (0) Indicación-TC-Invocación (1)		
Petición-TC-Comienzo (1)		
	COMIENZO ----->	
	CONTINUACIÓN <-----	
Indicación-TC-Continuación (1)		
Petición-TC-U-Aborto (1)		
	ABORTO ----->	
Petición-TC-Finalización (local,0)		

El argumento de invocación localConsumerOperation (operación de consumidor local) se define del modo siguiente:

```
testInit : {
    timeOut 30,
    commands {
        action : {service v1988beginReq, dialogueReference value :1},
        wait   : {dialogueReference value :1},
        action : {service uAbortReq, dialogueReference value :1},
        action : {service localEndReq, dialogueReference value : 0}
    }
}
```


ANEXO B

Establecimiento de bucles utilizando el respondedor de pruebas de TC

Este anexo ilustra la utilización del respondedor de pruebas de TC para el establecimiento de bucles. En este caso de prueba, tanto el sistema de prueba como el respondedor de pruebas abren de manera continua diálogos que son finalizados inmediatamente por la otra entidad.

Interfaz respondedor de pruebas/TC	Flujo de mensajes	Sistema de prueba
	COMIENZO (OTID = X1, testInit) <-----	
Indicación-TC-Comienzo (0)		
Petición-TC-Comienzo (1)		
	COMIENZO (OTID = Y1) ----->	
Petición-TC-Finalización (0)	FINALIZACIÓN (DTID = X1) ----->	
	FINALIZACIÓN (DTID = Y1) <-----	
	COMIENZO (OTID = X2, testContinue) <-----	
Indicación-TC-Comienzo (2)		
Petición-TC-Comienzo (3)		
	COMIENZO (OTID = Y2) ----->	
Petición-TC-Finalización (2)	FINALIZACIÓN (DTID = X2) ----->	
	FINALIZACIÓN (DTID = Y2) <-----	
...		

El primer mensaje COMIENZO enviado del probador al respondedor de pruebas incluye una PDU testInit definida del modo siguiente:

```

testInit : {
    timeOut 30,
    commands {
        action : {service v1988beginReq, dialogueReference value: 1},
        action : {service basicEndReq, dialogueReference value: 0},
        wait   : {dialogueReference value: 1}
    }
}
    
```

Los mensajes COMIENZO posteriores incluyen una PDU testContinue definida del modo siguiente:

```
testContinue : {  
    action : {service v1988beginReq, dialogueReference value: i+1},  
    action : {service basicEndReq, dialogueReference value: i},  
    wait : {dialogueReference value : i+1}  
}
```

donde i aumenta desde 1 al número N de bucles deseados.

Cuando $i=N$, se puede terminar el caso de prueba enviando una PDU testContinue, sin la instrucción v1988beginReq. En este ejemplo hay como máximo dos diálogos al mismo tiempo. No obstante, a efectos de la prueba de carga, el caso de prueba podría escribirse de nuevo de manera que el probador no envíe ningún mensaje FINALIZACIÓN y las TMP-PDU (excepto la que termina el caso de prueba) no contengan ninguna "basicEndRequest" (petición de finalización básica) en la lista de instrucciones.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedios
Serie I	Red digital de servicios integrados
Serie J	Transmisiones de señales radiofónicas, de televisión y de otras señales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos y comunicación entre sistemas abiertos
Serie Z	Lenguajes de programación