



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

**CCITT**

COMITÉ CONSULTIVO  
INTERNACIONAL  
TELEGRÁFICO Y TELEFÓNICO

**Q.775**

(11/1988)

SERIE Q: CONMUTACIÓN Y SEÑALIZACIÓN

Especificaciones del sistema de señalización N.º 7 –  
Parte Aplicación de Capacidades de Transacción (PACT)

---

**DIRECTRICES PARA LA UTILIZACIÓN DE  
CAPACIDADES DE TRANSACCIÓN**

Reedición de la Recomendación Q.775 del CCITT  
publicada en el Libro Azul, Fascículo VI.9 (1988)

---

## NOTAS

1 La Recomendación Q.775 del CCITT se publicó en el fascículo VI.9 del Libro Azul. Este fichero es un extracto del Libro Azul. Aunque la presentación y disposición del texto son ligeramente diferentes de la versión del Libro Azul, el contenido del fichero es idéntico a la citada versión y los derechos de autor siguen siendo los mismos (véase a continuación).

2 Por razones de concisión, el término «Administración» se utiliza en la presente Recomendación para designar a una administración de telecomunicaciones y a una empresa de explotación reconocida.

## DIRECTRICES PARA LA UTILIZACIÓN DE CAPACIDADES DE TRANSACCIÓN

(Melbourne, 1988)

### 1 Introducción

#### 1.1 Generalidades

El objeto de esta Recomendación es proporcionar directrices a los posibles usuarios de capacidades de transacción (usuarios CT). Se facilitan a título ilustrativo solamente, ejemplos que indican la manera en que una aplicación puede utilizar la PACT, pero no la manera en que la PACT debe emplearse en todos los casos. La base técnica de este documento la constituyen las Recomendaciones Q.771 a Q.774, que deben considerarse como referencia primaria en caso de desacuerdo.

El objeto principal de la PACT es proporcionar un soporte para aplicaciones interactivas en un entorno distribuido. La PACT se basa en las Recomendaciones X.219 y X.229 (ESOD) mejoradas para proporcionar los servicios que necesitan los usuarios CT. Las interacciones entre las entidades de aplicación distribuida se modelan mediante operaciones. Una operación se invoca mediante una entidad (de origen), en tanto que la otra entidad (de destino) intenta la ejecución de la operación y, posiblemente devuelve el resultado de esta tentativa.

La semántica de una operación (representada por su nombre y sus parámetros) no reviste interés para la PACT, la cual proporciona facilidades independientes de cualquier operación particular. Cuando defina una aplicación, el usuario CT deberá:

- 1) seleccionar operaciones;
- 2) seleccionar facilidades de la PACT para soportar esas operaciones. Tales facilidades comprenden el tratamiento de operaciones individuales y la capacidad de disponer de cierto número de operaciones conexas asociadas a una asociación entre usuarios CT, denominada diálogo;
- 3) definir el enunciado de la aplicación.

En esta Recomendación se describe el proceso de selección para la definición y utilización de las operaciones. Las operaciones que aparecen en lo que sigue son ficticias y se muestran con fines de ilustración solamente. Se describen, asimismo, las facilidades ofrecidas por la PACT para el tratamiento de una operación o de una secuencia de operaciones en un diálogo. La definición de secuencias específicas de operaciones pertenece a la definición del protocolo de aplicación, que cae fuera del ámbito de esta Recomendación. No obstante, en el capítulo 4 se proporciona una breve indicación de qué tipo de información debería contener una especificación de aplicación.

Los servicios de PACT son accesibles a los usuarios CT por conducto de primitivas. Estas primitivas modelan el interfaz entre la PACT y sus usuarios, pero no limitan la realización de este interfaz.

#### 1.2 Entorno

La PACT define el protocolo de extremo a extremo entre usuarios CT que pueden ubicarse en una red del sistema de señalización N.º 7 y/o en otra red que soporte protocolos de PACT.

Se han considerado dos grandes categorías de usuarios (véase el § 1.3.2 de la Recomendación Q.771). Aquí sólo se examina la primera categoría es decir, la de usuarios sensibles al tiempo real que no necesitan intercambiar grandes cantidades de datos. Se considera que, para este tipo de usuarios, los protocolos definidos por las capas 4 a 6 de la ISA en las Recomendaciones de la serie X, producirían un número excesivo de elementos de servicio, por lo que no se utilizan. Se especifica un servicio básico fundamentándose en un servicio de red sin conexión. Otras categorías de usuario pueden necesitar una red con conexión y servicio de capa superior.

En consecuencia, la PACT no puede soportar todas las clases de aplicaciones y algunas de éstas necesitarán apoyarse en servicios más elaborados, tales como los especificados en las Recomendaciones de la serie X. Además de expresar lo que puede efectuar la PACT, esta Recomendación indica lo que el enfoque sin conexión es incapaz de realizar, a fin de ayudar al proyectista de aplicaciones a elegir la manera de llevar a efecto una aplicación.

## 2 Operaciones

### 2.1 Definición

Un usuario CT de origen invoca una operación para pedir a un usuario CT de destino que realice una acción determinada.

A cada operación se le asocia una clase, que indica si el destino debe o no comunicar la consecuencia satisfactoria (resultado) o insatisfactoria (error), o ambas, o ninguna. La consecuencia se comunica en un resultado.

Además de la clase, la definición de una operación incluye un valor de temporización que indica cuándo debe completarse la operación. Este valor no se comunica al usuario CT distante, suponiéndose que en ambos extremos la aplicación posee una comprensión común de las operaciones en curso.

Una **operación** se define mediante:

- su código de operación y el tipo de los parámetros asociados con la petición de operación;
- su clase;
- si la clase requiere un informe de éxito, los posibles resultados correspondientes a las ejecuciones fructuosas se definen mediante una lista de parámetros;
- si la clase requiere un informe de fallo, los posibles resultados correspondientes a situaciones en que el usuario CT distante no pudo ejecutar completamente la operación. Cada una de estas situaciones se identifica mediante una causa de error específica. La lista de estas causas de error forma parte de la definición de la operación. Puede añadirse información de diagnóstico a la causa de error: si está presente, forma parte de la definición;
- la lista de las posibles operaciones enlazadas, si se admiten réplicas consistentes en operaciones enlazadas para esa operación. Las operaciones enlazadas deberán describirse separadamente;
- un valor de temporizador que indique el intervalo en el cual debe completarse la operación. Este valor de temporización se utiliza para gestionar el ID de componente asociado a la invocación de la operación.

### 2.2 Ejemplos

#### 2.2.1 Operaciones simples

*Nota* – La invocación de la operación, así como un informe de resultado infructuoso deberá caber en un mensaje. Los informes de éxito pueden segmentarse utilizando Devolución de Resultado-No Último (RE-NL) y Devolución de Resultado-Último (RE-L).

*Clase 1 (se indican tanto el éxito como el fallo):*

Traducir un número con franquicia a un número de abonado llamado; devolver el número llamado si puede efectuarse la traducción. En otro caso indicar por qué no es así; tiempo atribuido: 2 segundos.

La no recepción de una respuesta cuando expire el temporizador indica una situación anormal (por ejemplo, puede haberse perdido la invocación de la operación); se informa al usuario CT local (cancelación de la operación por la PACT).

*Clase 2 (sólo se notifican los fallos):*

Realizar una prueba de rutina y enviar una respuesta solamente en el caso en que exista algún fallo; tiempo atribuido: 1 minuto.

En el caso de una operación de clase 2, si no se recibe ningún resultado cuando expira el temporizador, se informa al usuario CT. Esto se interpreta como una realización fructuosa, aun cuando se pierda la invocación. Cuando se seleccione la clase 2 deberá contemplarse este aspecto.

*Clase 3 (sólo se notifica el éxito):*

Realizar una prueba: esto corresponde a un punto de vista pesimista en que se considera el fallo como opción por defecto que no requiere ninguna respuesta.

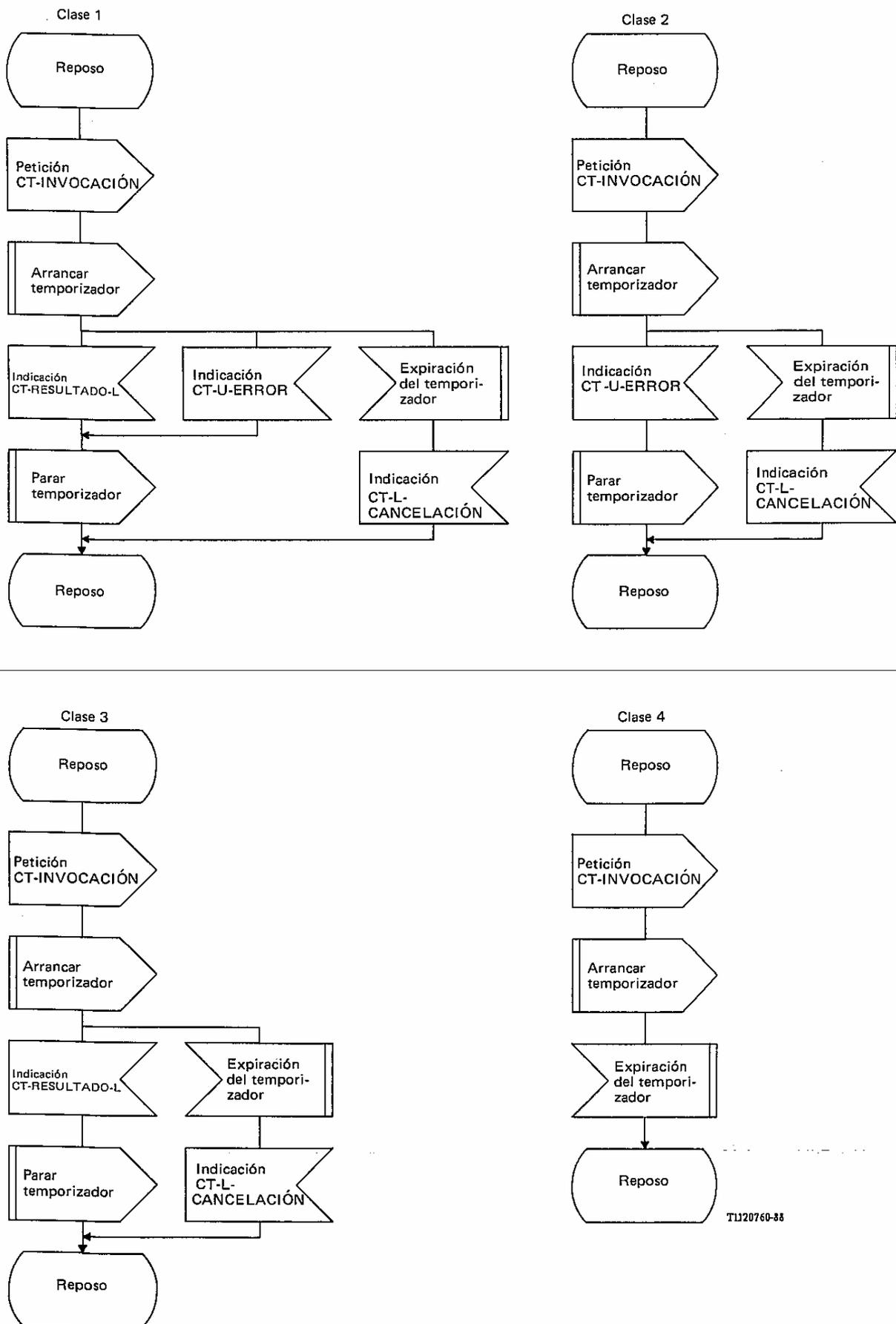
Se informa al usuario CT sobre la expiración del temporizador: el usuario CT deberá interpretar esta información como un fallo de la operación (aunque esto es considerado normal por el CT, que estima que la operación ha terminado). Al seleccionar la clase 3 deberá contemplarse este aspecto.

*Clase 4 (no se notifica ni el éxito ni el fallo):*

Enviar un aviso, sin esperar una respuesta ni acuse de recibo de ningún tipo.

En este caso, no se desprende ningún resultado de la invocación de la operación. El usuario CT confía en la PACT y en la red para la entrega de la invocación. La notificación de la expiración del temporizador es un asunto local.

En los diagramas de la figura 1/Q.775 se ilustran posibles secuencias de primitivas tal como son contempladas por el usuario CT que origina una operación.



TI120760-88

FIGURA 1/Q.775

*Comparación con las clases de operación de ESOD (Recomendación X.219)*

El ESOD establece 5 clases de operaciones: las clases 2 a 5, denominadas clases asíncronas, son idénticas a las clases 1 a 4 de la PACT. La clase 1 del ESOD es una clase síncrona que no tiene correspondencia en la PACT, donde se consideran intercambios dúplex de componentes. Sin embargo, un usuario CT puede decidir que el funcionamiento será asíncrono (véase el § 3.2.1).

**2.2.2 Operaciones más complicadas**

**Operaciones con resultados segmentados**

Un resultado fructuoso puede dividirse en varios segmentos, cada uno de los cuales se indica al originador de la aplicación mediante una primitiva. Los usuarios CT pueden emplear esta facilidad, que utiliza la primitiva CT-RESULTADO-NL, para remediar la ausencia de segmentación en las capas subyacentes. El último segmento se indica mediante la primitiva CT-RESULTADO-L.

La notificación de un error no puede segmentarse.

Salvo en el caso de situaciones anormales, las respuestas se entregan al usuario CT distante en el orden en que el usuario CT de origen las ha transferido a la PACT.

En el caso de un resultado segmentado, la CT no puede identificar un segmento específico.

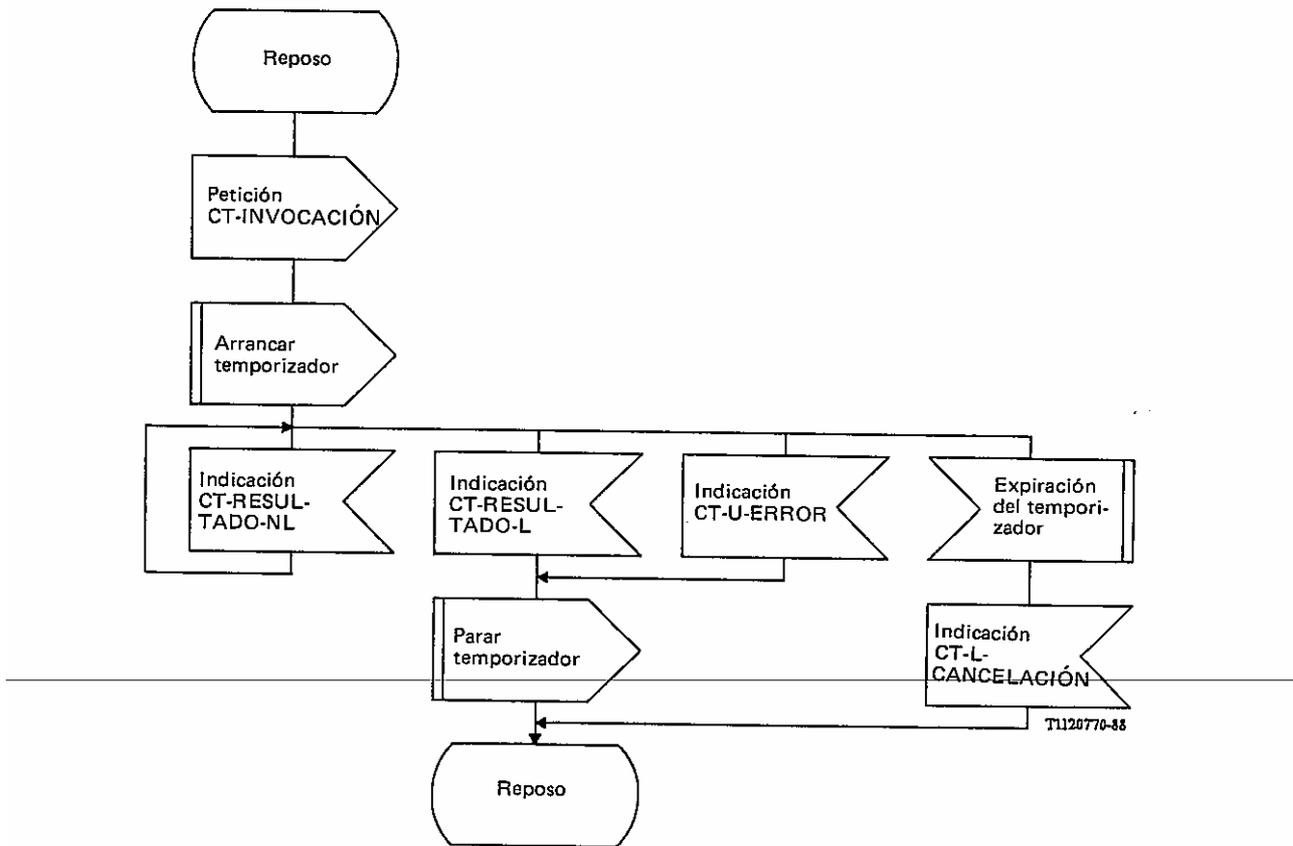
Ejemplo E1: Una operación solicita la ejecución de una prueba. El resultado de una ejecución correcta se segmenta en tres partes P1, P2 y P3, que se devuelven al originador.

Una posible secuencia de primitivas para el ejemplo E1 se indica en el cuadro 1/Q.775.

CUADRO 1/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (Prueba) Petición CT-RESULTADO-NL (P1)
Indicación CT-RESULTADO-NL (P1)	Petición CT-RESULTADO-NL (P2)
Indicación CT-RESULTADO-NL (P2)	Petición CT-RESULTADO-L (P3)
Indicación CT-RESULTADO-L (P3)	
Tiempo	

El diagrama de la figura 2/Q.775 ilustra posibles secuencias de primitivas contempladas por el originador de una operación (clase 1), con resultados segmentados.



*Nota* — El valor de la temporización es especificado por el usuario CT en el momento de la invocación. Un resultado no final no reanuda la temporización.

FIGURA 2/Q.775

*Operaciones enlazadas*

Otro elemento del método de operación básico es la capacidad de enlazar una invocación de operación con otra invocación de operación.

Normalmente esta facilidad abarca situaciones en que el destino de la operación original (con la que se efectúa el enlace) requiere información adicional, a fin de procesar esta operación: tal es el caso cuando se utilizan facilidades de menú (las facilidades de menú permiten a un usuario efectuar una secuencia de elecciones, cada una de las cuales depende de las anteriores).

Ejemplo E2: La operación consiste en la ejecución de una prueba con varias opciones. Antes de ejecutarse la prueba, se ofrecen esas opciones al originador de la misma para que efectúe una selección entre ellas (usuario CT A). Se entrelazan dos operaciones: la operación 1 es la prueba y la operación 2 es la de selección de la opción: el usuario CT A responde en primer lugar a la operación 2 antes de que el usuario CT B pueda efectuar la prueba con la opción u opciones indicadas.

Una posible secuencia de primitiva para el ejemplo 2 se indica en el cuadro 2/Q.775.

CUADRO 2/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (Prueba) Comienza la operación 1 Petición CT-INVOCACIÓN (Selección de opción, Clase = 1) Comienza la operación 2
Indicación CT-INVOCACIÓN (Selección de opción) Petición CT-RESULTADO-L (Opciones)	Indicación CT-RESULTADO-L (Opciones) Finaliza la operación 2 Petición CT-RESULTADO-L (Resultado de la prueba)
Indicación CT-RESULTADO-L (Resultado de la prueba)	Finaliza la operación 1
Tiempo	

No hay ningún límite al número de invocaciones de operación que puede enlazarse con una invocación de operación dada.

Debe observarse que, cuando una operación B está enlazada con otra operación A, éstas no necesitan estar entrelazadas. La única condición es que la invocación de B tenga lugar antes de que se notifique la consecuencia de A. Sin embargo, la operación B no tiene por qué terminar con anterioridad a la operación A.

### 2.3 Facilidades relacionadas con los componentes ofrecidos a los usuarios CT

#### 2.3.1 Invocación

Hasta ahora, se han considerado las operaciones desde el punto de vista estático. La invocación introduce un aspecto dinámico: debe diferenciarse la invocación específica de una operación de otras posibles invocaciones concurrentes de la misma operación o de otra cualquiera.

Cada activación particular de una operación se identifica mediante un ID de componente. Este ID de componente debe ser inequívoco. Es seleccionado por el usuario CT que origina la invocación de la operación y es transferido al usuario CT de destino que lo reflejará en su respuesta o respuestas: en consecuencia, correlaciona las respuestas a una invocación con la propia invocación.

El usuario CT tiene libertad para asignar cualquier valor al ID de componente (índice, dirección . . .).

El ID de componente asociado con una invocación puede reutilizarse cuando se ha recibido el único o último segmento de un resultado, o cuando la PACT indique situaciones anormales. Sin embargo, no debe reatribuirse el valor inmediatamente a otra activación de operación ya que la reatribución inmediata impediría el tratamiento correcto de algunas situaciones (véase más adelante).

Se denomina periodo de congelación al periodo en que se libera un ID de componente, pero en que éste no puede reatribuirse.

Como los ID de componente reciben su valor dinámicamente en el momento en que se invoca la operación, dicho valor no puede aparecer en la especificación de los protocolos de aplicación. Se trata más bien de indicar un valor «lógico», que debe sustituirse por un valor real en el momento de la ejecución, a fin de identificar una operación en un flujo único.

Teniendo en cuenta los ID de componente, la secuencia de primitivas para el ejemplo E2 anterior, sería la siguiente tal como se indica en el cuadro 3/Q.775

CUADRO 3/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-INVOCACIÓN (2, 1, Selección de opción, Clase = 1)
Indicación CT-INVOCACIÓN (2, 1, Selección de opción) Petición CT-RESULTADO-L (2, Opciones)	Indicación CT-RESULTADO-L (2, Opciones) Petición CT-RESULTADO-L (1, Resultado de la prueba)
Indicación CT-RESULTADO-L (1, Resultado de la prueba)	
Tiempo	

donde el primer parámetro de una primitiva indica un ID de invocación. Cuando deban estar presentes ambos parámetros, el segundo de ellos es el ID enlazado. Esto no es más que un convenio de notación.

2.3.2 Cancelación (por el usuario CT)

El usuario CT que pide la invocación de una operación puede parar la actividad asociada al ID de componente correspondiente, por cualquier motivo que considere apropiado. Sin embargo, en principio, deberá reservarse la cancelación para situaciones anormales. El método normal de terminación de una operación es la recepción de un resultado o la finalización basada en la expiración de un temporizador.

La cancelación tiene solamente efecto local: no impide que el usuario CT distante envíe respuestas a una operación cancelada. Cuando se reciban tales respuestas, serán rechazadas por la PACT como se indica en el ejemplo que sigue, que representa una secuencia de primitivas para el ejemplo E1 definidas anteriormente, en el caso en que el usuario CT A cancele la prueba tras la recepción del primer segmento del resultado.

En el cuadro 4/Q.775, el usuario CT A no recibe la parte P2: la PACT detecta una situación de rechazo antes de la entrega, por lo que en el lado A se rechaza todo intento del usuario CT B de enviar más respuestas.

CUADRO 4/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-RESULTADO-NL (1, P1)
Indicación CT-RESULTADO-NL (1, P1) Decisión de cancelación: Petición CT-CANCELACIÓN (1) Indicación CT-RECHAZO-L (1, Código de problema) <p style="text-align: right;">....</p>	Petición CT-RESULTADO-NL (1, P2) <p style="text-align: right;">....</p>
Tiempo	

2.3.3 Rechazo (por el usuario CT)

Un usuario CT puede decidir el rechazo de una componente por cualquier motivo que considere apropiado, por ejemplo, error de protocolo de aplicación, ausencia de un parámetro en una operación o en una respuesta, etc.

La PACT contempla en la Recomendación Q.773 cierto número de casos identificados por la lista de códigos de problema. En cualquiera de estos casos, que corresponden a situaciones en las que no se ha formateado correctamente una operación o una respuesta, el usuario CT puede hacer uso de la facilidad de rechazo. Alternativamente, puede decidir la devolución de una indicación de fallo (componente de error) que proporciona una información más detallada del error y del diagnóstico.

El rechazo de una invocación de operación o de un resultado afecta a la totalidad de la operación, por lo que para esta invocación no se aceptarán más respuestas. El rechazo de una operación enlazada no afecta a la operación conexas.

Este hecho se ilustra en el cuadro 5/Q.775 en la que, en el ejemplo E2, el usuario CT A no espera el proceso de selección de opción (puede tratarse de una característica facultativa) y rechaza la operación con el código de problema «operación enlazada inesperada». El usuario CT B, puede decidir entonces la ejecución de la prueba adoptando una opción por defecto.

CUADRO 5/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-INVOCACIÓN (2, 1, Selección de opción, Clase = 1)
Indicación CT-INVOCACIÓN (2, 1, Selección de opción) Petición CT-RECHAZO-U (2, Código de problema)	
	Indicación CT-RECHAZO-U (2, Código de problemas) Petición CT-RESULTADO-L (1, Resultado de la prueba)
Indicación CT-RESULTADO-L (1, Resultado de la prueba)	
Tiempo	

Cuando se rechaza una invocación de operación, el usuario CT puede decidir una nueva invocación (por ejemplo, en caso de corrupción del componente de invocación); esto podría ser una nueva invocación (nuevo ID de invocación). El usuario puede tomar también la decisión de abortar el diálogo. Un diálogo muy simple (una pregunta y una respuesta) puede no definir ningún tipo de mecanismo de recuperación, salvo cuando se trate de una operación de importancia esencial (por ejemplo, actualización de una base de datos).

#### 2.4 *Situaciones anormales relacionadas con los componentes*

##### 2.4.1 *Pérdida del componente*

La PACT presupone una probabilidad de pérdida de mensajes en la red muy pequeña. Si esta probabilidad es excesiva para una aplicación, deberá utilizarse el método de servicio de red con conexión. Si algunas informaciones de protocolo requieren una calidad de servicio mejorada (por ejemplo, información de tasación) la aplicación deberá establecer sus propios mecanismos para lograr una mayor fiabilidad para esta información.

Pérdida de una invocación de operación:

En el cuadro 6/Q.775 se ilustra el caso, correspondiente al ejemplo E1, en que no se recibe una respuesta a la prueba antes de la expiración del temporizador.

CUADRO 6/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	
Límite de tiempo: Indicación CT-CANCELACIÓN-L (1)	
Tiempo	

Quando se pierde una operación de clase 1, se informa al usuario CT de cuando expira el temporizador asociado con la operación. Cuando se pierde una operación de clase 1 con un solo resultado, la PACT no puede indicar si se ha perdido la invocación de operación o la respuesta. Si la aplicación debe establecer una discriminación entre estos dos casos, deberá hacerlo en el protocolo de aplicación (por ejemplo empleando una marca temporal o mediante acuse de recibo de la invocación de operación, antes de responder a la misma).

Para una operación de clase 2, la pérdida se considerará como un éxito (tanto si se perdió la invocación como el informe de fallo). Teniendo en cuenta la probabilidad de pérdida, esto puede ser aceptable para operaciones que no sean fundamentales (por ejemplo, mediciones estadísticas).

En el caso de una operación de clase 3, el tratamiento que se da a la pérdida es el mismo que el de un fallo de operación, tanto si se ha perdido la invocación como el informe de éxito.

Para una operación de clase 4, la pérdida no será visible por la PACT.

*Pérdida de un resultado:*

- La PACT no detecta nunca la pérdida de un resultado que no sea final.
- La pérdida de un resultado final se indicará, eventualmente, al usuario CT cuando se alcance el límite de tiempo, pero no puede siempre interpretarse de forma inequívoca como pérdida de respuesta. Si no se ha recibido ningún resultado no final, puede deberse a que se haya perdido la invocación.

*Pérdida de una operación enlazada:*

La pérdida de una operación enlazada surte el mismo efecto que la pérdida de una operación no enlazada. No tiene ningún efecto sobre la operación con la que se realiza el enlace.

*Pérdida de un componente de rechazo:*

Este caso tendrá una frecuencia de aparición muy reducida por lo que ninguna aplicación intentará recuperarse de tal situación. Si la pérdida del rechazo afecta a una invocación de operación, cuando expire la temporización de la operación, el usuario CT que invocó la operación estimará que dicha invocación (o la respuesta) se ha perdido y reaccionará en consecuencia. Si se trata de una respuesta, el originador de la respuesta considerará que era correcta y la detección de la pérdida corresponderá al originador de la operación.

#### 2.4.2 Duplicación de un componente

Como las duplicaciones de mensajes son muy raras en la red SS7, no es necesario que en las normas que regulan las aplicaciones del sistema N.º 7 se definan escenarios complicados para prever estas situaciones. Sin embargo, para cualquier aplicación en que la duplicación fuera inaceptable, debería definirse su propio mecanismo de detección de duplicación o utilizarse un servicio con conexión.

*Invocación de operación duplicada*

Quando se duplica una invocación de operación (por el proveedor del servicio), el usuario CT de destino (B) puede o no detectar la operación:

- El usuario CT B detecta la duplicación: lo mejor que puede hacer en este caso es ignorar la duplicación. El rechazo podría ser interpretado por el usuario CT distante como un rechazo de la invocación original.
- El usuario CT B no detecta la duplicación: esto puede ocurrir cuando existe una relación principal/subordinado entre A y B, y B ejecuta la operación con desconocimiento del contexto.

Suponiendo, en el ejemplo E1, que se trata del segundo caso, una secuencia posible podría ser la indicada en el cuadro 7/Q.775.

CUADRO 7/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)  Indicación CT-RESULTADO-NL (1, P1) Indicación CT-RESULTADO-NL (1, P1) A detecta una situación anormal y la rechaza: Petición CT-RECHAZO-U (1, Código de problema) CT detecta una situación anormal y rechaza P2: Indicación CT-RECHAZO-L (1, Código de problema)	Indicación CT-INVOCACIÓN (1, Prueba) Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-RESULTADO-NL (1, P1) Petición CT-RESULTADO-NL (1, P1)  Petición CT-RESULTADO-NL (1, P2) Indicación CT-RECHAZO-U (1, Código de problema)
	Indicación CT-RECHAZO-U (1, Código de problema)
Tiempo	

En esta secuencia, el usuario CT B considera dos invocaciones de prueba independientes y responde a cada una de ellas. Se acepta el primer resultado P1. El usuario CT A detecta que P1 se ha recibido por segunda vez y lo rechaza. Esto concluye la operación y origina el rechazo del resultado P2 cuando se reciba (rechazado por la PACT). En consecuencia, en el lado B ambas actividades concluirán con la recepción de los rechazos.

*Duplicación de un resultado no final*

La PACT no puede detectar la duplicación de un resultado no final, por lo que lo entregará dos veces al usuario CT. La detección de esta situación queda para la aplicación.

*Duplicación de un resultado final*

En caso de duplicación de un resultado final la PACT puede detectar la situación. El segundo resultado final será considerado como anormal (la operación ha finalizado con el primer resultado «final») y la PACT la rechaza.

En el cuadro 8/Q.775 se indica una secuencia correspondiente al ejemplo E1 en el que está duplicado (por la red) el tercer segmento del resultado.

CUADRO 8/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)  Indicación CT-RESULTADO-NL (1, P1) Indicación CT-RESULTADO-NL (1, P2) Indicación CT-RESULTADO-L (1, P3) Duplicación de P3: Indicación CT-RECHAZO-L (1, Código de problema)	Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-RESULTADO-NL (1, P1) Petición CT-RESULTADO-NL (1, P2) Petición CT-RESULTADO-L (1, P3)
	Indicación CT-RECHAZO-R (1, Código de problema)
Tiempo	

Comentario: Podría parecer que una solución más elegante sería que la PACT eliminase los duplicados en todos los casos. Sin embargo, debe observarse que:

- 1) esto exigiría otro grado de complejidad en la PACT, que contradice las características básicas de la PACT en el método sin conexión;
- 2) esto corresponde a una situación que es sumamente rara, al menos en la red del sistema de señalización N.º 7.

Para afrontar estas situaciones cuando lo requiera una aplicación, es mejor utilizar el método de servicio de red con conexión, ya que en el mismo puede detectarse y manejarse la duplicación en las capas inferiores.

#### 2.4.3 *Desorden de los componentes*

El orden de los resultados segmentados no es importante para la PACT. Si dicho orden es importante para el usuario CT, deberán definirse mecanismos apropiados en el protocolo de aplicación (por ejemplo, introduciendo un método de numeración para identificar las respuestas intermedias en un parámetro de estas respuestas, o empleando un servicio con conexión).

Debido a una secuencia errónea, puede llegar un resultado no final antes que un resultado final. Cuando esto suceda, la PACT rechazará el resultado no final.

En la secuencia ilustrada en el cuadro 9/Q.775 se indica lo que ocurre en el ejemplo E1 cuando se recibe la última parte del resultado antes que la segunda: se informa a ambos usuarios CT.

CUADRO 9/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (1, Prueba) Petición CT-RESULTADO-NL (1, P1)
Indicación CT-RESULTADO-NL (1, P1)	Petición CT-RESULTADO-NL (1, P2)
Indicación CT-RESULTADO-L (1, P3) Resultado con secuencia errónea: Rechazo Indicación CT-RECHAZO-L (1, Código de problema)	Petición CT-RESULTADO-L (1, P3)
	Indicación CT-RECHAZO-R (1, Código de problema)
Tiempo	

Si se recibe una invocación de operación enlazada tras el resultado final de la operación con la que se efectúa el enlace (como consecuencia de una secuencia errónea), se rechaza la operación enlazada.

La PACT presupone una probabilidad de secuencia errónea muy reducida. Si la red soporte no es satisfactoria a este respecto, deberá considerarse la utilización del servicio de red con la conexión.

#### 2.4.4 Rechazo de un componente por la PACT

Cuando la PACT recibe un componente (invocación de operación o respuesta) que no se ha formateado correctamente o se ha recibido fuera de contexto (por ejemplo, respuesta sin invocación de operación previa), un principio general es proceder a su rechazo, lo que significa que:

- 1) se informa, en primer lugar, de la situación al destinatario del componente que falla. La PACT proporciona toda la información disponible sobre la naturaleza del componente rechazado;
- 2) como reacción, el usuario CT puede tomar la decisión de abortar, continuar o finalizar el diálogo. En los dos últimos casos, cuando el usuario CT notifique su decisión a la PACT, se informará del rechazo al usuario CT par.

En las secciones anteriores han aparecido los posibles casos de rechazo por la PACT. Siempre que se reconozca el ID de componente, el rechazo por la PACT produce la terminación de la operación. Una posible recuperación consistirá en una nueva invocación de la operación terminada. Solamente se informará al usuario CT local cuando no pueda identificarse la componente rechazada y entonces la reacción apropiada puede ser la de abortar el diálogo.

#### 2.4.5 Expiración del temporizador de operación

Cuando la PACT informa al usuario CT sobre la expiración del temporizador (indicación CT-L-CANCELACIÓN), ello indica que no puede recibirse más información sobre la invocación de la operación (en especial, ningún rechazar). Si la entidad par envía todavía información en relación con esta invocación, se descartará dicha información cuando se reciba, siempre que no se haya reatribuido el ID de componente de la operación cancelada. Se evita, generalmente, la reatribución prematura de los valores de ID de componentes, ajustando

correctamente los valores del temporizador. A fin de compensar las incertidumbres del tiempo total necesario para enviar información desde un usuario CT a otro, sin tener en cuenta el caso más desfavorable absoluto (que es además, en general, el más improbable) se necesita el establecimiento de un mecanismo dependiente de la realización, que evite la reatribución prematura de los ID de componente.

La indicación de expiración de un temporizador corresponde a una situación anormal en el caso de las operaciones de clase 1 solamente. En ese caso, el usuario CT es consciente de que se ha perdido la invocación o la respuesta. Si no surgen efectos secundarios indeseables, puede producirse otra invocación de la misma operación tras la expiración del temporizador. Esto se indica, en el ejemplo E1, mediante la secuencia ilustrada en el cuadro 10/Q.775.

CUADRO 10/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Prueba, Clase = 1)	Indicación CT-INVOCACIÓN (1, Prueba)
Expiración del temporizador: Indicación CT-CANCELACIÓN-L (1) Petición CT-INVOCACIÓN (2, Prueba, Clase = 1)	
Tiempo	

Para una operación de clase 2, la expiración del temporizador indica que, para esta invocación, no se ha recibido o no se aceptará ningún fallo. Se trata de una indicación definida de éxito (para la clase 2). En caso de fallo, se aplica a la clase 3 una situación similar. La indicación de expiración del temporizador para una operación de clase 4 es una decisión local.

### 3 Diálogos

Siempre que se emita una de las primitivas de tratamiento de operación contempladas en el § 2, se cursa una petición a la PACT, pero no se envía nada al usuario CT distante, hasta que se emita una primitiva de petición de transmisión. A continuación se tratará de estas primitivas y de su relación con la primitivas de tratamiento de operación.

#### 3.1 Agrupación de componentes en un mensaje

El efecto de la emisión por parte de un usuario CT de una primitiva de tratamiento de componente (a menos que dicha primitiva tenga solamente efecto local) es la creación de un componente para incluirlo en un mensaje. El mensaje no se transmite hasta que el usuario CT lo solicite.

Obsérvese que puede también generarse un componente como resultado de un rechazo de la PACT: en este caso, dicho componente se coloca en el siguiente mensaje de diálogo, salvo si éste se aborta.

Pueden agruparse varios componentes y enviarse al extremo distante como un único mensaje para ahorrar elementos de transmisión, siempre que no se exceda el tamaño máximo de un mensaje. Esto se efectúa bajo control del usuario CT, el cual especifica, de forma explícita, cuando desea la emisión de uno o varios componentes.

El ejemplo E3 ilustrado en la figura 11/Q.775 muestra el comienzo de un diálogo con un centro de servicio de red, en el que un conmutador solicita instrucciones (operación 1) y recibe una petición para conectar la llamada a una dirección de destino dada, y una petición para enviar información (por ejemplo, anuncios o un mensaje para su representación) a la parte llamante. Ambos componentes están contenidos en un solo mensaje.

CUADRO 11/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (1, Proporcionar-instrucciones, Clase = 1) Petición CT-COMENZO (Parámetros de control)	
	Indicación CT-COMIENZO (Parámetros de control) Indicación CT-INVOCACIÓN (1, Proporcionar-Instrucciones) Petición CT-INVOCACIÓN (2, 1, Conectar llamada) Petición CT-RESULTADO-L (1, Enviar información) Petición CT-CONTINUACIÓN (Parámetros de control)
Indicación CT-CONTINUACIÓN (Parámetros de control) Indicación CT-INVOCACIÓN (2, 1, Conectar llamada ) Indicación CT-RESULTADO-L (1, Enviar información)	
Tiempo	

CT-COMIENZO y CT-CONTINUACIÓN son primitivas de transmisión que se describen a continuación en el § 3.2.

Puede haber más de una primitiva de transmisión para cada componente, si bien la separación de primitivas permite la agrupación de componentes dentro de un mensaje. Además, la información contenida en los parámetros de las primitivas de transmisión (por ejemplo, información de direccionamiento), se aplica a todos los componentes incluidos en el mensaje.

En el lado de origen, la primitiva que solicita la transmisión aparece detrás de una primitiva de tratamiento de componente. Esto indica que la transmisión de los componentes precedentes se efectuará inmediatamente y evita indicar los componentes específicos que han de transmitirse con una primitiva de transmisión dada, permitiendo el uso de primitivas de transmisión sin ningún componente asociado.

En el lado de destino aparece, en primer término, la primitiva de petición de transmisión que contiene información de control necesaria a la PACT para la entrega de cada componente (si existe) del mensaje. El parámetro «último componente» indica al usuario CT cuál es el último componente del mensaje. Los componentes se entregan al usuario CT de destino en el mismo orden en que el usuario CT de origen las transfirió a la PACT.

### 3.2 *Facilidades de tratamiento de diálogo*

Cuando dos usuarios CT cooperan en una aplicación se necesita, generalmente más de una invocación de operación. Debe identificarse el flujo resultante de componentes de forma que:

- 1) puedan relacionarse las componentes del mismo flujo;
- 2) puedan identificarse flujos correspondientes a diversas instancias de la misma aplicación y permitirles que discurren en paralelo.

Cada uno de estos flujos se identifica al usuario CT mediante un diálogo y el parámetro ID de diálogo correspondiente. La facilidad de tratamiento de diálogo ofrecida para este fin es el diálogo estructurado.

Cuando sólo se necesita un mensaje para completar una aplicación distribuida, puede emplearse el mensaje unidireccional del diálogo no estructurado. El originador no espera un informe de la consecuencia de la operación (es decir, sólo puede invocar operaciones de clase 4), pero puede recibir un informe de error de protocolo, si lo hay.

### 3.2.1 *Diálogo estructurado*

#### 3.2.1.1 *Generalidades*

La utilización de diálogos permite la coexistencia de varios flujos de componentes entre dos usuarios CT. Se utiliza el parámetro ID de diálogo en las primitivas de tratamiento de operación y tratamiento de transmisión (diálogo) para determinar qué componente o componentes pertenecen a cada diálogo.

El parámetro ID de diálogo está representado (por convenio) por el primer parámetro de estas primitivas que comienza con la letra D. Cada usuario CT posee su propia referencia para un diálogo determinado. Aquí se representan las referencias locales (las utilizadas en el interfaz). La PACT efectúa la correspondencia entre estas referencias locales y las referencias de protocolo incluidas en los mensajes.

Para el tratamiento de diálogos en circunstancias normales, se han definido tres primitivas que indican el comienzo del diálogo (CT-COMIENZO) su continuación (CT-CONTINUACIÓN) o su finalización (CT-FINALIZACIÓN). Puede utilizarse cada una de estas primitivas para solicitar la transmisión de 0, 1 o varios componentes. Estos componentes pueden contener información relativa a una o varias operaciones.

En el cuadro 12/Q.775 se representa, para el ejemplo E2, una secuencia posible en la que el diálogo comienza con una petición de prueba y finaliza cuando se ha enviado el resultado de la prueba.

CUADRO 12/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (D1, 1, Prueba, Clase = 1) Petición CT-COMIENZO (D1, Dirección)	
	Indicación CT-COMIENZO (D2, Dirección) Indicación CT-INVOCACIÓN (D2, 1, Prueba)
	Petición CT-INVOCACIÓN (D2, 2, 1, Selección de opción, Clase = 1) Petición CT-CONTINUACIÓN (D2)
Indicación CT-CONTINUACIÓN (D1) Indicación CT-INVOCACIÓN (D1, 2, 1, Selección de opción) Petición CT-RESULTADO-L (D1, 2, Opciones) Petición CT-CONTINUACIÓN (D1)  Indicación CT-FINALIZACIÓN (D1, normal) Indicación CT-RESULTADO-L (D1, 1, Resultado de la prueba)	Indicación CT-CONTINUACIÓN (D2) Indicación CT-RESULTADO (D2, 2, Opciones) Petición CT-RESULTADO-L (D2, 1, Resultado de la prueba) Petición CT-FINALIZACIÓN (D2)
Tiempo	

Nota – D1 y D2 son referencias locales para el mismo diálogo y se corresponden con los ID de transacción que aparecen en los mensajes.

En los mensajes de un diálogo se permite cualquier agrupación de componentes: la PACT no comprueba, por ejemplo, que un mensaje con el que finaliza un diálogo no incluye invocaciones de operación de clase 1. Se presupone el intercambio dúplex completo de componentes: si un usuario CT desea introducir algunas restricciones, por ejemplo el funcionamiento en un modo síncrono, según se define en el ESOD, deberá introducir por sí mismo los procedimientos necesarios.

### 3.2.1.2 *Intercambio de mensajes*

La transmisión de mensajes se efectúa con la calidad de servicio correspondiente a los servicios de la capa subyacente. La PACT no proporciona mecanismos de control de flujo ni de recuperación de errores.

- La primera primitiva de tratamiento de diálogo de un diálogo deberá indicar el comienzo del mismo (CT-COMIENZO). No podrán enviarse mensajes ulteriores desde el lado en que se origina el diálogo, hasta que se reciba un mensaje en sentido opuesto, que indica la continuación del diálogo.
- Si un usuario CT trata de enviar un gran número de mensajes en un breve intervalo de tiempo, ningún mecanismo de control del flujo de la PACT podrá evitarlo.
- Como opción puede solicitarse la entrega secuencial PCCS de clase 1 indicada por el parámetro calidad de servicio. Obsérvese que esta opción puede no estar disponible de extremo a extremo, en casos de interfuncionamiento con una red que no la proporcione.

### 3.2.1.3 *Finalización del diálogo*

La PACT no impone ninguna restricción a la capacidad de un usuario CT para solicitar la finalización de un diálogo. Por este motivo, pueden perderse mensajes si en la aplicación no se adoptan precauciones relativas a la finalización del diálogo. En particular, si el protocolo de aplicación permite a ambos usuarios CT la emisión de primitivas CT-FINALIZACIÓN aproximadamente al mismo tiempo, y si esas primitivas desencadenan la transmisión de componentes, es probable que algunos de esos componentes (si no todos) no se entreguen a sus respectivos usuarios CT de destino.

Corresponde a la aplicación la definición, en caso de necesidad, de sus propias reglas relativas al derecho de terminar un diálogo. La PACT no las comprobará. En caso de un diálogo terminado, se descarta todo mensaje recibido si solicita la finalización del diálogo, y en otro caso hace que el diálogo se aborte en la entidad distante.

Las diferencias entre las tres formas de finalización de un diálogo son las siguientes:

#### *Finalización concertada*

Una aplicación típica es el acceso a una base de datos distribuida en la que el usuario solicitante (usuario CT A) no sabe dónde está ubicada la información que busca. El usuario CT A difunde una petición a cada ubicación en la que podría encontrarse la información solicitada y recibirá oportunamente, una respuesta del usuario CT que posee esa información. La finalización concertada evita el envío de mensajes desde otro destino con el contenido: «No tengo esa información». Solamente el destino que responde continuará el diálogo (si así se desea); los restantes destinos finalizarán, por convenio, el diálogo localmente. El originador de las preguntas finalizará también de forma local el diálogo con los destinos que no respondan, cuando reciba la respuesta a su pregunta. Obsérvese que se trata de un convenio entre aplicaciones: la PACT no comprueba su cumplimiento ni ello se indica en el protocolo PACT.

En el ejemplo E4 del cuadro 13/Q.775 se ilustra esta situación con dos destinos (B1 y B2), iniciándose dos diálogos (D1, D2) y (D3, D4). La información solicitada está disponible en B1, el cual decide la continuación del diálogo.

CUADRO 13/Q.775

USUARIO CT A	USUARIO CT B1	USUARIO CT B2
Petición CT-INVOCACIÓN (D1, 1, Pregunta) Petición CT-COMIENZO (D1, Dirección) Petición CT-INVOCACIÓN (D3, 1, Pregunta) Petición CT-COMIENZO (D3, Dirección)	Indicación CT-COMIENZO (D2, Dirección) Indicación CT-INVOCACIÓN (D2, 1, Pregunta)	Indicación CT-COMIENZO (D4, Dirección) Indicación CT-INVOCACIÓN (D4, 1, Pregunta) – B2 no posee la información: Petición CT-FINALIZACIÓN (D4, local)
Indicación CT-CONTINUACIÓN (D1) Indicación CT-RESULTADO-L (D1, 1, Respuesta) D1 prosigue D3 termina localmente	Petición CT-RESULTADO-L (D2, 1, Respuesta) Petición CT-CONTINUACIÓN (D2)	
Petición CT-FINALIZACIÓN (D3, local)		
Tiempo		

Puede también emplearse la finalización concertada cuando un usuario CT desea enviar información y no espera respuesta de ningún tipo.

*Finalización básica*

Cuando un usuario CT emite la primitiva de petición CT-FINALIZACIÓN provoca la transmisión de todos los componentes pendientes al extremo distante. La PACT no comprueba que todas las invocaciones de operación han recibido una respuesta cuando se solicita la finalización del diálogo, ni se notifica al usuario CT de que algunas invocaciones de operación pendientes no han recibido el resultado final.

En el extremo receptor, se considera terminado el diálogo cuando se han entregado al usuario CT todos los componentes recibidos en el mensaje que indica la finalización.

Ejemplo: el diálogo finaliza cuando la prueba del ejemplo E1, cuadro 14/Q.775, recibe una respuesta.

CUADRO 14/Q.775

USUARIO CT A	USUARIO CT B
<p style="text-align: center;">.....</p> <p>Indicación CT-FINALIZACIÓN (D1)            Indicación CT-RESULTADO-NL (D1, 1, P1)            Indicación CT-RESULTADO-NL (D1, 1, P2)            Indicación CT-RESULTADO (D1, 1, P3)            Finalización del diálogo para A</p>	<p style="text-align: center;">.....</p> <p>Petición CT-RESULTADO-NL (D2, 1, P1)            Petición CT-RESULTADO-NL (D2, 1, P2)            Petición CT-RESULTADO-L (D2, 1, P3)            Petición CT-FINALIZACIÓN (D2, normal)            Finalización del diálogo para B</p>
Tiempo	

*Aborto por la parte del usuario CT*

La facilidad de aborto permite al usuario CT detener el diálogo en cualquier momento. Un caso típico es el abandono del servicio por parte del usuario. Las diferencias principales entre estos casos y la finalización normal, son las siguientes:

- los componentes cuya transmisión está pendiente no se envían a la entidad par;
- en el momento de producirse el aborto, puede indicarse la información entre partes pares y entregarla al usuario CT distante.

La secuencia ilustrada en el cuadro 15/Q.775 muestra un abandono por el usuario en el ejemplo E2.

3.2.1.4 *Situaciones anormales relacionadas con el mensaje*

Estas se consideran con independencia de sus efectos en la subcapa componente.

*Pérdida del mensaje*

La PACT no proporciona protección contra la pérdida de mensajes. Se identifican tres casos:

- 1) Con el mensaje comienza un nuevo diálogo: el diálogo existirá en el lado de origen solamente, no permitiéndose ningún mensaje en ningún sentido. Oportunamente, un mecanismo de la PACT, dependiente de la realización, finalizará el diálogo en el extremo de origen.
- 2) El mensaje es la continuación de un diálogo existente: no se detecta la pérdida. La PACT reaccionará (o no) a la pérdida de los componentes incluidos, según se indica en el § 2.4.1 anterior.
- 3) Con el mensaje finaliza un diálogo: la PACT reaccionará oportunamente si este mensaje contiene una respuesta a una operación de clase 1. En otro caso, un mecanismo dependiente de la realización podrá finalizar el diálogo en el extremo de destino.

CUADRO 15/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (D1, 1, Prueba, Clase = 1) Petición CT-COMIENZO (D1, Dirección)	
	Indicación CT-COMIENZO (D2, Dirección) Indicación CT-INVOCACIÓN (D2, 1, Prueba) Petición CT-INVOCACIÓN (D2, 2, 1, Selección de opción, Clase = 1) Petición CT-CONTINUACIÓN (D2)
Indicación CT-CONTINUACIÓN (D1) Indicación CT-INVOCACIÓN (D1, 2, 1, Selección de opción) Abandono del usuario: Petición CT-U-ABORTO (D1, Causa)	
	Indicación CT-U-ABORTO (D2, Causa)
Tiempo	

*Duplicación de mensajes*

La duplicación de un mensaje COMIENZO provoca la apertura de dos transacciones, como se indica más adelante: cada una de estas transacciones tiene su propio ID local y el mismo ID de destino. El usuario CT detecta a su debido tiempo que algo es incorrecto, por lo que se abortan ambos diálogos.

La secuencia ilustra una duplicación del mensaje COMIENZO en el ejemplo E2:

CUADRO 16/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (D1, 1, Prueba, Clase = 1) Petición CT-COMIENZO (D1, Dirección)	
	Indicación CT-COMIENZO (D2, Dirección) Indicación CT-INVOCACIÓN (D2, 1, Prueba) COMIENZO duplicado: Indicación CT-COMIENZO (D3, Dirección)
Indicación CT-CONTINUACIÓN (D1) Indicación CT-INVOCACIÓN (D1, 2, 1, Selección de opción)	Indicación CT-INVOCACIÓN (D3, 1, Prueba) Respuesta al primer comienzo Petición CT-INVOCACIÓN (D2, 2, 1, Selección de opción, Clase = 1) Petición CT-CONTINUACIÓN (D2) Respuesta al segundo Comienzo Indicación CT-INVOCACIÓN (D3, 2, 1, Selección de opción, Clase = 1) Petición CT-CONTINUACIÓN (D3)
Indicación CT-CONTINUACIÓN (D1) Indicación CT-INVOCACIÓN (D1, 2, 1, Selección de opción) El usuario CT considera que esta invocación es anormal y puede rechazarla o abortar uno de los diálogos: Petición CT-U-ABORTO (D1, Causa)	
	Indicación CT-U-ABORTO (D3, Causa)
Tiempo	

En ese momento, existe todavía un diálogo (con el ID local D2) en el lado del usuario CT B pero no en el lado del usuario CT A. El usuario CT B recibirá una indicación de la PACT cuando expire la temporización de la operación 2 del diálogo D2 sin respuesta (Indicación CT-L-CANCELACIÓN) y puede entonces decidir abortar D2. Obsérvese que la situación sería más difícil de detectar si el usuario CT B no invocase una operación de clase 1.

La PACT no detecta la duplicación del mensaje de CONTINUACIÓN.

Cuando se duplica un mensaje de FINALIZACIÓN, el segundo mensaje se recibe con un ID que no corresponde a un diálogo activo: la PACT reacciona descartando el mensaje duplicado.

### *Secuencia errónea de mensajes*

Cuando la secuencia errónea de los mensajes no afecta ni al principio ni al final de un diálogo, no es detectada por la PACT, lo cual puede provocar una secuencia errónea de componentes a la que la PACT reaccionará como se indica en el § 2.5.3 anterior.

Cuando un mensaje, que indica la continuación de diálogo, llega tras un mensaje que indica el final del mismo diálogo, no será entregado y provocará que la PACT aborte el diálogo. El usuario CT detectará probablemente la pérdida al recibir una indicación prematura de finalización del diálogo. Si la aplicación necesita recuperarse de este caso, deberá iniciarse un nuevo diálogo.

### *Adulteración de mensajes*

Al recibir un mensaje adulterado, la PACT reacciona como se indica en la Recomendación Q.774.

En el cuadro 17/Q.775 se indica la secuencia de primitivas cuando la PACT decide abortar el diálogo tras recibir un mensaje adulterado en el ejemplo E2.

CUADRO 17/Q.775

USUARIO CT A	USUARIO CT B
Petición CT-INVOCACIÓN (D1, 1, Prueba, Clase = 1) Petición CT-COMIENZO (D1, Dirección)	
	Indicación CT-COMIENZO (D2, Dirección) Indicación CT-INVOCACIÓN (D2, 1, Prueba) Petición CT-INVOCACIÓN (D2, 2, 1, Selección de opción, Clase = 1) Petición CT-CONTINUACIÓN (D2)
Mensaje adulterado: Indicación CT-ABORTO (D1, Causa)	Indicación CT-ABORTO (D2, Causa)
Tiempo	

### 3.2.1.5 *Relaciones entre el tratamiento de diálogos y el tratamiento de operaciones*

Según el momento en que solicite la finalización del diálogo, se dispondrá o no de las facilidades PACT asociadas con una operación hasta la finalización del diálogo. A continuación se dan orientaciones sobre cuándo puede solicitarse la finalización del diálogo. Si no se respetan, la PACT no rechazará la petición de finalización del diálogo.

Se han examinado anteriormente los problemas que pueden surgir por colisión de mensajes que soliciten la finalización del diálogo.

No se solicitará la finalización normal cuando:

- existan, para el diálogo, invocaciones de operaciones pendientes;
- el protocolo de aplicación anticipe que podrían rechazarse las respuestas transmitidas con la petición de terminación.

Además, una petición de terminación de diálogo no deberá desencadenar la transmisión de invocaciones de operación, ya que para tales operaciones no podría recibirse ninguna respuesta.

Muchas aplicaciones podrían no definir escenarios de recuperación como contestación a una respuesta rechazada. Esto justifica la transmisión de respuestas o de operaciones de clase 4 en un mensaje que indica la finalización de un diálogo. Las demás aplicaciones deberán usar el servicio de red con conexión o finalizar el diálogo con un mensaje que no contenga componentes y que se enviará solamente cuando ya no pueda recibirse una indicación de rechazo.

### 3.2.2 *Diálogo no estructurado*

Un mensaje unidireccional contiene, bien únicamente invocaciones de operación de clase 4, o informes de errores de protocolo en esas invocaciones. Pueden transmitirse en un mensaje unidireccional, siempre y cuando no se exceda el tamaño máximo del mensaje.

## **4 Elementos del servicio de aplicación y entidades de aplicación**

### 4.1 *Introducción*

Este texto constituye un complemento del material precedente que proporciona orientaciones sobre la utilización de la CT describiendo qué necesidades han de incluirse en la especificación de una entidad de aplicación (EA). Su contenido se basa en las Recomendaciones X.219 y X.229 del CCITT y requiere ulterior estudio.

En el § 3.2.3.6 de la Recomendación Q.700 del CCITT, se describe cómo se estructuran los elementos del servicio de aplicación (ESA) y las entidades de aplicación (EA) y cómo se direcciona una EA en el sistema de señalización N.º 7.

En esta sección se indica esa arquitectura teniendo en cuenta la descomposición funcional de una aplicación, y se describe cómo deberían definirse los ESA, EA, operaciones y errores.

### 4.2 *Descomposición de la funcionalidad*

Las funciones de proceso de aplicación se comunican a través de una o más entidades de aplicación (EA). La combinación de dos EA pares y su interacción se denomina contexto de aplicación. Una EA consta de comunicaciones para una o más funciones de una aplicación. Cada función de comunicación constituye una ESA que es un conjunto integrado de acciones que puede utilizarse en más de una EA. La propia PACT es una ESA utilizada por otras ESA y común a varias EA (véase el § 3.2.3.6 de la Recomendación Q.700). Una ESA identifica una o más operaciones y especifica cómo se utilizan esas operaciones, esto es, qué operaciones puede invocar cada entidad par, y en qué orden. Las operaciones pueden seleccionarse de una o más bibliotecas.

Una ESA proporciona un servicio al usuario de la ESA. Una ESA es utilizada por dos EA complementarias: el consumidor del servicio y el proveedor del servicio. El consumidor del servicio es el extremo que inicia la comunicación EA-EA. Así pues, un usuario ESA es, generalmente, asimétrico.

Dentro de una ESA, el mecanismo para proporcionar el servicio ESA consiste en la invocación de operaciones por el solicitante o el proveedor del servicio. Cada operación proporciona una parte del servicio de forma intrínsecamente asimétrica, puesto que es invocada por una EA y ejecutada por la EA par. Una ESA incluye, generalmente, más de una operación. El usuario de una ESA no está, en general, limitado a la invocación o realización de operaciones, sino que puede tanto invocar como realizar la misma operación u operaciones diferentes. Además, el usuario de una ESA puede existir en un par de nodos de forma que cualquiera de los nodos puede solicitar el mismo servicio al otro nodo. Es decir que las EA de los nodos pueden ser simétricas, al invocar y ejecutar las mismas operaciones.

*Nota* – Las primitivas que proporcionan un interfaz de servicio normalizado para el acceso a las ESA dentro de las EA se estudiarán en fecha ulterior.

En la figura 3/Q.775, se indica la descomposición de esta funcionalidad y se dan ejemplos.

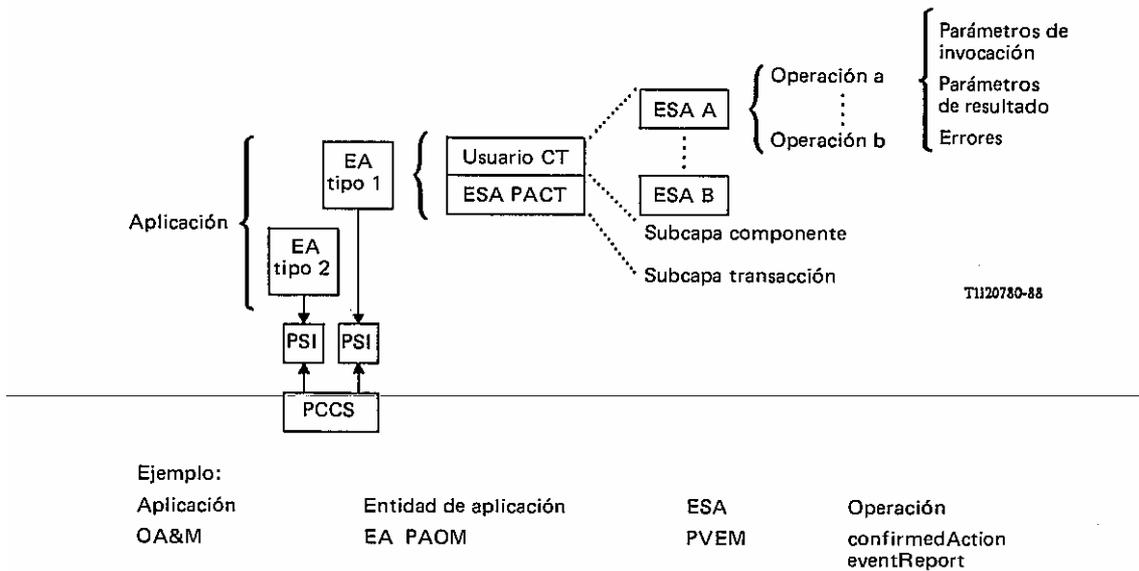


FIGURA 3/Q.775  
Descomposición de la funcionalidad

#### 4.3 Especificación de una EA

En el § 3.2.3.6 de la Recomendación Q.700 del CCITT, se describe cómo se comunican dos procesos de aplicación del sistema de señalización N.º 7, a través de entidades de aplicación, y se indica asimismo la estructura de una EA.

El diseñador de la aplicación ha de proporcionar para cada tipo EA una definición que deberá contener:

- una descripción general de los servicios soportados por la combinación de las dos EA pares y que comunican mediante un diálogo (según la terminología de la Recomendación X.229, esto corresponde al «contexto de aplicación»);
- una definición del protocolo de aplicación completo entre los dos EA pares mediante la:
  - identificación de cada ESA que constituye la EA, y la
  - indicación de cuál de las EA pares inicia el servicio,
- limitaciones especiales que aseguren que las EA pares con versiones diferentes sean compatibles.

Deberá estudiarse ulteriormente la especificación formal del contexto de aplicación utilizando el macro de CONTEXTO-APLICACIÓN de la Recomendación X.229.

Como cada EA constituye un solo dominio de codificación para los valores de operación y de código de error (direccionados por el número de subsistema PCCS en un entorno de servicio de red sin conexión) cada valor de operación o de código de error debe ser único dentro de la EA (véase el § 4.5).

#### 4.4 Especificación de una ESA

La definición de una ESA forma parte de la etapa 3 de la metodología de descripción del servicio definida en la Recomendación I.220.

La descripción de la ESA deberá proporcionar:

- Una descripción general de la ESA y de sus procedimientos.
- Los flujos de información entre las entidades que se comunican para soportar el servicio, basados en la etapa 2, con las adiciones y mejoras que sean necesarias como parte del diseño del protocolo.
- Una descripción detallada del protocolo ESA. Comprende la secuencia en que pueden invocarse, las operaciones y la reacción ante situaciones anormales. La definición deberá indicar cómo interfuncionan las versiones del protocolo. Deberá especificarse el comienzo, la continuación y la finalización del diálogo. En esta sección deberá describirse la interacción entre la ESA y la subcapa componente de la PACT expresada en términos del interfaz de la primitiva.
- Diagramas LED.

En la Recomendación X.229 (ESOD), se define un macro de ELEMENTO-SERVICIO-APLICACIÓN que puede utilizarse para especificar formalmente una ESA. Identifica qué operaciones están contenidas en la EA y cómo se invocan. Se estudiará ulteriormente la utilización de este macro en el sistema de señalización N.º 7.

#### 4.5 *Especificación de operaciones y errores*

##### 4.5.1 *Información necesaria para la especificación de operaciones y errores*

Para especificar una operación, deberán definirse los siguientes elementos:

- Nombre de la operación.
- Código de la operación. Puede ser local o global. Véase el § 4.5.2.
- Clase de la operación. Valor comprendido entre 1 y 4 según se define en el § 2.2.1.
- Parámetros que acompañan a la invocación de la operación (parámetros de entrada). Puede solicitarse información esencial adicional que complete la que proporcionan los parámetros de la invocación original empleando operaciones enlazadas.
- Parámetros que pueden devolverse como consecuencia de una realización satisfactoria (devolución de resultado), cuando la operación comunique un éxito (parámetros de resultado positivo). La forma en que se transfieren estos parámetros (en un solo componente o en varios), no forma parte de la descripción de la operación.
- Códigos de error y parámetros asociados que pueden devolverse como consecuencia de un resultado infructuoso (devolución de error) o de la ejecución de la operación cuando esta operación comunica un fallo (parámetros de resultado negativo). Al comunicar un fallo, debe estar presente un código de error, definiéndose todos los valores posibles como parte de la descripción de la operación.
- Operaciones enlazadas admitidas (véase el § 2.2.2).
- Valor del temporizador para la finalización de la operación.

La descripción de la operación consta de un cuadro que indica estos 8 elementos, junto con una breve descripción de lo que realiza la operación. Debe incluirse una definición formal, utilizando los macros OPERACIÓN y ERROR del anexo A a la Recomendación Q.773, para indicar inequívocamente qué parámetros son obligatorios, cuales son facultativos con los valores por defecto que procedan, y qué parámetros, conjuntos o secuencias de parámetros son legales como entradas, salida positiva y salida negativa. Las definiciones de los macros de OPERACIÓN y ERROR proceden de las definiciones de la PACT (anexo A a la Recomendación Q.773) y deben incorporarse a la definición de las ESA, a fin de definir las operaciones y los errores.

La sintaxis de MACRO OPERACIÓN (extracto del anexo A a la Recomendación Q.773) es la siguiente:

OPERATION MACRO ::=

BEGIN

TYPE NOTATION ::=           Parameter Result Errors Linked Operations

VALUE NOTATION ::=        value{ VALUE CHOICE{  
                                  localValue INTEGER,  
                                  globalValue OBJECT IDENTIFIER } }

Parameter ::=               “PARAMETER” NamedType | empty

Result ::=                  “RESULT” ResultType | empty

```

ResultType ::=          NamedType | empty
Errors ::=             "ERRORS" "{"ErrorNames" | empty
LinkedOperations ::=   "LINKED" "{"LinkedOperationNames" | empty
ErrorNames ::=        ErrorList | empty
ErrorList ::=         Error | ErrorList "," Error
Error ::=             value (ERROR) -- referenciará un valor de error
                    | type   -- referenciará un tipo de error si no se
                    -- especifica ningún valor de error

LinkedOperationNames ::= OperationList | empty
OperationList ::=     Operation | OperationList "," Operation
Operation ::=         value (OPERATION) -- referenciará un valor de operación
                    | type   -- referenciará un tipo de operación si no se
                    -- especifica ningún valor de operación

NamedType ::=         identifiertype | type
END
ERROR MACRO ::=
BEGIN
TYPE NOTATION ::=     Parameter
VALUE NOTATION ::=    value (VALUE CHOICE{
                        localValue INTEGER,
                        globalValue OBJECT IDENTIFIER })
Parameter ::=        "PARAMETER" NamedType | empty
NamedType ::=        identifiertype | type
END

```

En el § 4.5.2 se explica la utilización de los valores global y local.

Como ejemplo, se describe aquí en notación formal (abreviada) y en forma de cuadro, la operación de comprobación GCU 2 utilizada para comprobar si una llamada entrante es compatible con las características GCU de la parte llamada.

#### 4.5.2 *Ejemplo de la descripción de una operación*

(Nota – Los puntos de este ejemplo están numerados en forma arbitraria.)

##### 3.4.3.1 *Descripción de operaciones*

###### 3.4.3.1.1 *Comprobación GCU 1*

Esta operación se emplea entre la central de origen de una llamada y un punto especializado para la comprobación de la validación GCU del usuario llamante.

###### 3.4.3.1.2 *Comprobación GCU 2*

Esta operación se emplea entre la central de destino de una llamada y un punto especializado para la comprobación de la validación GCU del usuario llamado.

###### 3.4.3.2 *Parámetros de operaciones y resultados*

3.4.3.2.1 Comprobación GCU 1

Comprobación GCU 1	Temp. = x seg.	Clase = 1	Código = 0000001
Parámetros con invocación		Fac./Oblig.	Referencia
CallingUserIndex		F	3.4.3.3.1
CUGCallIndicator		O	3.4.3.3.2
CallingPartyNumber		O	3.4.3.3.3
Parámetros con devolución de resultado			
CUGInterlockCode		F	3.4.3.3.5
CUGCallIndicator		O	3.4.3.3.2
Operaciones enlazadas			
No aplicable			
Errores			
UnsuccessfulCheck			3.4.3.3.7

```

cUGCheck1    OPERATION
PARAMETER    SEQUENCE{ callingUserIndex OPTIONAL, cUGCallIndicator,
                  callingPartyNumber }
RESULT       SEQUENCE{ cUGInterlockCode OPTIONAL, cUGCallIndicator }
ERRORS       { unsuccessfulCheck }
 ::= 1
    
```

### 3.4.3.2.2 Comprobación GCU 2

Comprobación GCU 2	Temp. = x seg.	Clase = 1	Código = 00000010
Parámetros con invocación		Fac./Oblig.	Referencia
CGUInterlockCode		O	3.4.3.3.5
CUGCallIndicator		O	3.4.3.3.2
CallingPartyNumber		O	3.4.3.3.4
Parámetros con devolución de resultado			
CalledUserIndex		F	3.4.3.3.6
CUGCallIndicator		O	3.4.3.3.2
Operaciones enlazadas			
No aplicable			
Errores			
<del>UnsuccessfulCheck</del>			3.4.3.3.7

```

cUGCheck2      OPERATION
  PARAMETER    SEQUENCE{ cUGInterlockCode, cUGCallIndicator,
                        calledPartyNumber }
  RESULT       SEQUENCE{ calledUserIndex OPTIONAL, cUGCallIndicator }
  ERRORS       { unsuccessfulCheck }
  ::= 2
  
```

### 3.4.3.3 Codificación de parámetros

3.4.3.3.1 CallingUserIndex es el índice local en el usuario llamante, para identificar un GCU determinado al que éste pertenece.

CallingUserIndex		Código = 10000001
Contenido	Significado	
Cadena de caracteres AI5	Un carácter AI5 representa una cifra del valor del índice GCU	

```

callingUserIndex ::= [1] IMPLICIT LocalIndex
LocalIndex ::= IA5 STRING
-- El número máximo de cifras es cuatro.
  
```

3.4.3.3.2 GCUCallIndicator indica si la llamada está pedida o designada como llamada GCU y si se pide o autoriza el acceso de salida.

CUGCallIndicator		Código = 10000010
Contenido	Significado	
00000000	Llamada no GCU	
00000001	Llamada no GCU	
00000010	Llamada GCU con acceso de salida	
00000011	Llamada GCU sin acceso de salida	

```
cUGCallIndicator ::= [2] IMPLICIT CallIndicator
CallIndicator ::= INTEGER{
    nonCUGCall (0),
    nonCUGCall (1),
    outgoingAccessAllowedCUGCall (2),
    outgoingAccessNotAllowedCUGCall (3)}
```

3.4.3.3.3 CallingPartyNumber es el número de red (por ejemplo, E.164) de la parte llamante. Se expresa del mismo modo que el número de la parte llamante PUSI del § 3.7 de la Recomendación Q.763. El código de este parámetro es «10000011».

CallingPartyNumber		Código = 10000011
Contenido	Significado	
-- codificado de acuerdo con § 3.7/Q.763		

```
callingPartyNumber ::= [3] IMPLICIT OCTET STRING
-- el contenido se codifica de acuerdo con el § 3.7 de la Recomendación Q.763.
```

3.4.3.3.4 CalledPartyNumber es el número de red (por ejemplo, E.164) de la parte llamada. Se expresa del mismo modo que el número de la parte llamada PUSI del § 3.6 de la Recomendación Q.763. El código de este parámetro es «10000100».

CalledPartyNumber		Código = 10000100
Contenido	Significado	
-- codificado de acuerdo con § 3.6/Q.763		

```
calledPartyNumber ::= [4] IMPLICIT OCTET STRING
-- el contenido se codifica de acuerdo con el § 3.6 de la Recomendación Q.763.
```

3.4.3.3.5 CUGInterlockCode es el código único de identificación de un GCU dentro de la red. Se expresa del mismo modo que el código de enclavamiento GCU PUSI del § 3.13 de la Recomendación Q.763. El código de este parámetro es «10000101».

CUGInterlockCode		Código = 10000101
Contenido	Significado	
-- codificado de acuerdo con § 3.13/Q.763		

CUGInterlockCode ::= [5] IMPLICIT OCTET STRING

-- el contenido se codifica de acuerdo con el § 3.13 de la Recomendación Q.763.

3.4.3.3.6 CalledUserIndex es el índice local en el usuario llamado para identificar un GCU determinado al que éste pertenece. Véase el § 3.4.3.3.1. El código de este parámetro es «10000110».

CalledUserIndex		Código = 10000110
Contenido	Significado	
Cadena de caracteres AI5	Un carácter AI5 representa una cifra del valor del índice CGU	

CalledUserIndex ::= [6] IMPLICIT LOCAL INDEX

#### 3.4.3.3.7 Errores

UnsuccessfulCheck		Código = 00000001
Parámetros		
Causa	3.4.3.3.8	

```

unsuccessfulCheck    ERROR
    PARAMETER{ Cause }
    ::= 1
    
```

3.4.3.3.8 Causa indica el motivo del fallo de la comprobación GCU.

Causa		Código = 10000111
Contenido binario decimal	Significado	
00110010 (50)	Facilidad pedida no suscrita	
00110101 (53)	Llamadas salientes prohibidas dentro del GCU	
00110111 (55)	Llamadas entrantes prohibidas dentro del GCU	
00111110 (62)	Incoherencia en la información del acceso de salida designado y la clase de abonado	
01010110 (90)	GCU inexistente	
01010111 (87)	El usuario llamado no es miembro del GCU	
01011000 (88)	Destino incompatible	
10000000 (110)	Incoherencia de los datos	

cause ::= [7] IMPLICIT CauseCode

```
CauseCode ::= INTEGER{
    requestedFacilityNotSubscribed(50),
    outgoingCallsBarredWithinCUG(53),
    incomingCallsBarredWithinCUG(55),
    inconsistencyInDesignatedOutgoingAccessInformationAndsubscriberClass(62),
    nonExistentCUG(90),
    calledUserNotMemberOfCUG(87),
    incompatibleDestination(88),
    inconsistencyInData(110) }
```

4.5.3 *Atribución y gestión de los códigos de operación y de error*

El enfoque más simple consiste en proporcionar un módulo que contenga la definición de las operaciones y de los errores que utiliza como un dominio local autosuficiente.

Antes de proceder a definir una nueva operación, el diseñador de la aplicación deberá verificar todos los módulos, a fin de comprobar si existe ya una operación similar. Para evitar definir nuevamente la operación en un cierto número de módulos, se necesita disponer de métodos que permitan que un módulo importe la definición de las operaciones que utiliza de otros módulos. Si la operación no existe, el diseñador deberá especificarla localmente.

*Ejemplo:* El código de operación 00000010 tiene un significado para la ESA 1 y, probablemente un significado totalmente distinto para la ESA 2: afecta a dos dominios.

Obsérvese que en una ESA pueden utilizarse numerosos dominios. Sin embargo, por razones de sencillez, se supone en lo que sigue que una ESA utiliza solamente un dominio.

Además de sus operaciones locales, una ESA puede necesitar utilizar operaciones ya definidas en otro dominio. Existen dos métodos para efectuar esto:

- importar tipos de operaciones y de error desde otros módulos;
- importar valores de operación y de error desde otros módulos.

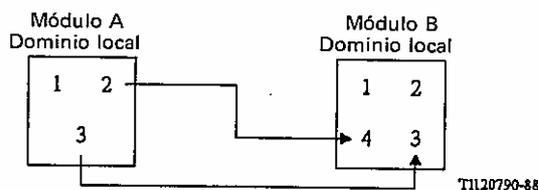
#### 4.5.3.1 Importación de tipos

La definición de un tipo de operación comprende los aspectos de notación (véase la OPERACIÓN MACRO anterior) sin atribución de los valores de código.

Puede ser conveniente importar el tipo de una operación ya existente, sin embargo el módulo importador puede desear atribuir su propio punto de código local a la operación o al error importados. La operación o el error importados, se transforman en un miembro del dominio local de ese módulo.

Si dos módulos distintos importan una operación dada mediante el tipo, su punto de código en cada uno de los dominios locales de importación es, en general, distinto.

La importación mediante tipos permite una descripción común de las operaciones. Una importación de módulos por tipos utiliza un solo dominio (su dominio local), como se representa en la figura 4/Q.775.



Las operaciones 2 y 3 del dominio del módulo A, se transforman, respectivamente en las operaciones 4 y 3 del dominio del módulo B.

**FIGURA 4/Q.775**  
**Importación por tipo**

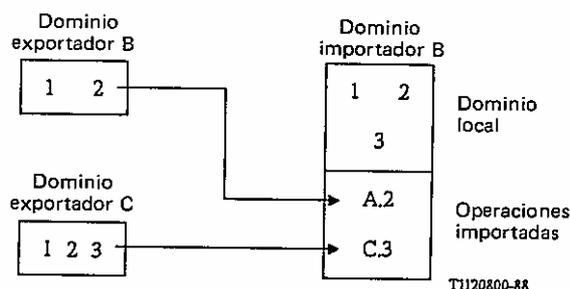
#### 4.5.3.2 Importación de valores

Cuando se importan valores de operación, el tipo y la codificación son los mismos en las ESA exportadora e importadora.

Un módulo que importe operaciones o errores mediante valores, utiliza:

- un dominio local para sus operaciones locales, y
- los dominios exportadores para sus operaciones importadas.

En el segundo caso, se necesita un valor global para evitar la ambigüedad entre los códigos de punto locales y los códigos de punto importados, como se representa en la figura 5/Q.775.



**FIGURA 5/Q.775**  
**Importación por valor global**

#### 4.6 *Aplicación de los conceptos a los protocolos de servicio*

Antes de efectuar la asignación de códigos de operación, deben examinarse en primer término las ESA de servicio (cada una de las cuales es un conjunto integrado de acciones) y asignarlas a las EA. Los extremos son, por una parte que todas las ESA de servicio asignadas a una EA y por otra, que cada EA esté constituida por una ESA de servicio solamente. El caso más probable corresponde a varias agrupaciones de ESA de servicio.

Cada ESA deberá identificarse por un NSE pero no necesariamente un NSE fijo especificado en la Recomendación Q.713. Dentro de una EA se utiliza un esquema de asignación de código de operación de manera que dos operaciones no puedan tener el mismo código de operación.





## 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
<b>Serie Q</b>	<b>Conmutación y señalización</b>
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 Y	Infraestructura mundial de la información y aspectos del protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación