



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

Q.823

(07/96)

SERIE Q: CONMUTACIÓN Y SEÑALIZACIÓN

Especificaciones del sistema de señalización N.º 7 –
Interfaz Q3

**Especificaciones funcionales de las etapas 2 y 3
para la gestión del tráfico**

Recomendación UIT-T Q.823

(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 transferencia de mensajes simplificada	Q.710
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
Generalidades	Q.850–Q.919
Capa de enlace de datos	Q.920–Q.929
Capa de red	Q.930–Q.939
Gestión usuario-red	Q.940–Q.949
Descripción de la etapa 3 para los servicios suplementarios que utilizan el sistema de señalización digital de abonado DSS 1	Q.950–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.823

ESPECIFICACIONES FUNCIONALES DE LAS ETAPAS 2 Y 3 PARA LA GESTIÓN DEL TRÁFICO

Resumen

La finalidad de esta Recomendación es ofrecer un modelo de información que comprenda los aspectos de gestión de las funciones de servicio de gestión de tráfico en una central. El alcance se limita únicamente a redes con conmutación de circuitos que utilicen encaminamiento jerárquico.

Esta Recomendación se centra en la descripción de las etapas 2 y 3 de la interfaz Q3 entre los elementos de red (NE) y los sistemas de operaciones (OS). La descripción de la etapa 1 figura en las Recomendaciones de la serie E.410 y en la Recomendación E.502.

Orígenes

La Recomendación UIT-T Q.823 ha sido preparada por la Comisión de Estudio 11 (1993-1996) del UIT-T y fue aprobada por el procedimiento de la Resolución N.º 1 de la CMNT el 9 de julio de 1996.

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 ha recibido/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 1997

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 Introducción	1
2.1 Requisitos funcionales	2
2.2 Metodología de modelado.....	2
3 Referencias normativas.....	2
4 Definiciones	3
4.1 Encaminamiento alternativo	3
4.2 Encaminamiento directo	4
5 Símbolos y abreviaturas.....	4
6 Modelo de información.....	4
6.1 Diagrama de relaciones entre entidades.....	4
6.2 Jerarquía de herencia.....	7
6.3 Jerarquía de denominación	8
7 Descripción del modelo de información.....	9
7.1 Descripciones de clases de objetos	10
7.1.1 Fragmento de elemento gestionado	10
7.1.2 Fragmento de supervisión de calidad de funcionamiento.....	12
7.1.3 Fragmento de control.....	23
7.1.4 Fragmento de estado	42
7.1.5 Fragmento administrativo.....	43
7.2 Definición de los atributos comunes.....	43
7.2.1 Nombre distinguido relativo.....	43
7.2.2 Atributos de estado	43
7.3 Descripción de las acciones	44
7.4 Descripción de las notificaciones.....	44
8 Unidades funcionales	44
8.1 Unidades funcionales de otras Recomendaciones	45
8.2 Negociación de unidades funcionales.....	45
9 Relación con otras Recomendaciones.....	46
10 Conformidad	46
10.1 Conformidad estática	46
10.2 Conformidad dinámica	47

10.3	Requisitos de enunciado de conformidad de implementación de gestión	47
11	Definiciones formales de clases de objeto	48
11.1	Definición de clases de objeto	48
11.1.1	ACC	48
11.1.2	accAffectedTraffic	48
11.1.3	accTrigger	49
11.1.4	adc	49
11.1.5	adcTrigger	50
11.1.6	cancelFrom	51
11.1.7	cancelRerouted	52
11.1.8	cancelTo	52
11.1.9	CircuitEndPointSubgroupCurrentData	53
11.1.10	circuitEndPointSubgroupHistoryData	53
11.1.11	congestionLevelIndication	54
11.1.12	destinationCodeControl	54
11.1.13	dccGroup	55
11.1.14	exchangeCurrentData	56
11.1.15	exchangeHistoryData	56
11.1.16	htrDestination	57
11.1.17	observedDestination	58
11.1.18	observedDestinationCurrentData	58
11.1.19	observedDestinationHistoryData	59
11.1.20	scr	59
11.1.21	scrAffectedTraffic	60
11.1.22	skip	60
11.1.23	StateIndicator	61
11.1.24	tarFrom	61
11.1.25	tarTo	62
11.1.26	tmCircuitEndPointSubgroup	63
11.1.27	tmCircuitEndPointSubgroupCurrentData	63
11.1.28	tmExchangeCurrentData	64
11.1.29	tmObservedDestinationCurrentData	64
11.1.30	tmTrafficControlCurrentData	65
11.1.31	trafficControl	65
11.1.32	trafficControlCurrentData	66
11.1.33	trafficControlHistoryData	66
11.2	Definición de vinculación de nombres	67
11.3	Definición de lotes	70

11.4	Definición de comportamiento común.....	74
11.5	Definición de atributos.....	74
11.6	Definición de acciones.....	84
11.7	Definición de modificaciones	85
11.8	Módulo tipos definidos de ASN.1	86
Apéndice I – Servicio de gestión de la RGT de gestión de tráfico		91
Apéndice II – Correlación entre los controles de tráfico de la Recomendación E.412 y las clases de objeto tratadas en esta Recomendación.....		92
Apéndice III – Diagrama de transición de estados para controles automáticos y manuales		93
Apéndice IV – Diferenciación del tráfico con encaminamiento directo y con encaminamiento alternativo para los controles cancelación hacia y cancelación desde		94
Apéndice V – Utilización de los datos históricos de la Recomendación Q.822 para mediciones de tráfico		95
V.1	Introducción	95
V.2	Visión general del modelo de información de gestión de calidad de funcionamiento.....	95
V.3	Configuración inicial	96
	V.3.1 Datos actuales	96
	V.3.2 Datos históricos	96
	V.3.3 Escáner simple.....	96
V.4	Informe autónomo.....	97
V.5	Informe a petición.....	98
V.6	Cambios al informe.....	98
V.7	Conclusiones.....	98
Apéndice VI – Definición de objeto subhaz de punto extremo de circuitos.....		100
Apéndice VII – Correspondencia de funciones NM con unidades funcionales.....		102
Apéndice VIII – Soporte de la funcionalidad de auditoría utilizando la Recomendación Q.823.....		103
VIII.1	Introducción	103
VIII.2	Requerimientos funcionales.....	103
VIII.3	Tipos de auditoría	104
	VIII.3.1 Auditoría total/parcial.....	104
	VIII.3.2 Auditoría de cambios.....	104
VIII.4	Mecanismos	104

VIII.4.1 Utilización del servicio PT-OBTENCIÓN (PT-GET).....	104
VIII.4.2 Utilización de un escáner simple.....	106
VIII.4.3 Utilización del objeto descubrimiento.....	106
VIII.5 Conclusiones.....	106
Apéndice IX – Ejemplos de errores de direccionamiento en la Recomendación Q.823.....	107
IX.1 Introducción.....	107
IX.2 Errores generados por la máquina de protocolo.....	107
IX.3 Utilización de errores CMISE.....	107
IX.4 Utilización de errores específicos.....	109
IX.5 Conclusión.....	110
Apéndice X – Jerarquía de control.....	110

Recomendación Q.823

ESPECIFICACIONES FUNCIONALES DE LAS ETAPAS 2 Y 3 PARA LA GESTIÓN DEL TRÁFICO

(Ginebra, 1996)

1 Alcance

La finalidad de esta Recomendación es ofrecer un modelo de información que comprenda los aspectos de gestión de las funciones de servicio de gestión de tráfico en una central. El alcance se limita únicamente a redes con conmutación de circuitos que utilicen encaminamiento jerárquico.

Esta Recomendación se centra en la descripción de las etapas 2 y 3 de la interfaz Q3 entre los elementos de red (NE) y los sistemas de operaciones (OS). La descripción de la etapa 1 figura en las Recomendaciones de la serie E.410 y en la Recomendación E.502.

Han de tenerse en cuenta las siguientes restricciones del alcance:

- Para la vigilancia y la gestión del tráfico, esta Recomendación hace uso de las mediciones de tráfico definidas en las Recomendaciones de la serie E.
- El tratamiento y el procesamiento de la información relativa a la gestión de tráfico en el nivel OS y el envío de estos datos en el nivel OS caen fuera del alcance de esta Recomendación.
- Se necesitan datos de calidad de funcionamiento de la red como información para los controles de gestión de tráfico. Algunas veces, los mismos datos pueden utilizarse para mediciones de tráfico, pero esta coincidencia es ignorada en este contexto. Los datos de calidad de funcionamiento de la red se identifican y modelan en la medida en que son pertinentes y cuando no se dispone de los modelos de otras Recomendaciones.
- Las funciones que pueden ser soportadas por modelos de información definidos en otras Recomendaciones no se redefinen en la presente Recomendación. Estos modelos de información son referenciados bien sea directamente mediante la incorporación de los objetos o indirectamente por subclasificación a partir de otra información.
- En la medida de lo posible y de lo razonable, se consideran las funciones definidas en las funciones de gestión de sistemas OSI.

2 Introducción

El objetivo de la gestión de tráfico es hacer posible que se efectúen con éxito tantas llamadas como sea posible. Este objetivo se cumple maximizando la utilización de los recursos disponibles en cualquier situación. Se considera también una parte de esta función supervisar la calidad de funcionamiento de una red y poder, si es necesario, ejercer acción para controlar el flujo de tráfico a fin de optimizar la utilización de la capacidad de red.

El modelo de información presentado en esta Recomendación presenta una visión común de la recuperación de datos de calidad de funcionamiento desde el NE, y de la administración de controles e instrucciones desde el OS al NE. Los datos de calidad de funcionamiento proporcionan información para la activación de los controles de gestión de tráfico y para la validación de acciones de gestión de tráfico. Sirven también como datos de entrada para otras acciones de gestión de tráfico.

2.1 Requisitos funcionales

Las funciones de control de gestión de tráfico consideradas en el modelo se basan en controles especificados en la Recomendación E.412.

Las funciones de supervisión de calidad de funcionamiento en este modelo se basan en elementos de medición seleccionados de la Recomendación E.502.

El apéndice I identifica las distintas funciones de los servicios de gestión de tráfico incorporadas en la Recomendación M.3200 que son tratadas por esta Recomendación y las que caen fuera del alcance de esta Recomendación.

El apéndice II presenta una correspondencia entre los controles de la Recomendación E.412 y los objetos de control tratados en esta Recomendación.

2.2 Metodología de modelado

Las técnicas de modelado descritas en la Recomendación M.3020 (Metodología de especificación de la interfaz de la red de gestión de las telecomunicaciones) se utilizan en esta Recomendación.

Las definiciones de las clases de objeto gestionado soporte indicadas en las Recomendaciones de OSI (Recomendaciones de la serie X.700) han sido adoptadas en la labor de modelado.

3 Referencias normativas

Las siguientes Recomendaciones UIT-T y otras publicaciones 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.

- Recomendación E.410 del CCITT (1992), *Gestión de la red internacional – Información general*.
- Recomendación E.411 del CCITT (1992), *Gestión de la red internacional – Directrices de explotación*.
- Recomendación E.412 del CCITT (1992), *Controles de gestión de la red*.
- Recomendación E.502 del CCITT (1992), *Requisitos de las medidas de tráfico para las centrales digitales de telecomunicación*.
- Recomendación UIT-T M.3010 (1996), *Principios para una red de gestión de las telecomunicaciones*.
- Recomendación UIT-T M.3100 (1995), *Modelo genérico de información de red*.
- Recomendación UIT-T Q.542 (1993), *Objetivos de diseño de las centrales digitales – Operación y mantenimiento*.
- Recomendación UIT-T Q.763 (1993), *Formatos y códigos de la parte usuario de la red digital de servicios integrados del sistema de señalización N.º 7*.
- Recomendación UIT-T Q.822 (1994), *Descripción de la etapa 1, de la etapa 2 y de la etapa 3 para la interfaz Q3 – Gestión de la calidad de funcionamiento*.

- Recomendación X.720 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Modelo de información de gestión.*
- Recomendación X.721 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Definición de la información de gestión.*
- Recomendación X.722 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Directrices para la definición de objetos gestionados.*
- Recomendación X.730 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de objetos.*
- Recomendación X.731 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de estados.*
- Recomendación X.734 del CCITT (1992), *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de informes de eventos.*
- Recomendación UIT-T X.738 (1993), *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de sumario.*

4 Definiciones

En esta Recomendación se definen los términos siguientes.

4.1 Encaminamiento alternativo

4.1.1 tráfico con encaminamiento alternativo: Conjunto de llamadas ofrecidas para las que un subhaz de circuitos no es la primera elección, es decir, el caso de desbordamiento.

4.1.2 señal de respuesta: Información enviada hacia atrás indicando que se responde a la llamada. Este término se define en el anexo A/E.410.

4.1.3 intento de toma: Intento de obtener un circuito de un subhaz o hacia un destino. Un intento de toma puede ser fructuoso o infructuoso en cuanto a la toma de un circuito de dicho haz de circuitos, o hacia ese destino. Este término se define en el anexo A/E.410.

4.1.4 circuito: Medio de transmisión que permite la comunicación entre dos centrales. Este término se define en el anexo A/E.410.

4.1.5 haz de circuitos: Conjunto de todos los circuitos conmutados que interconectan directamente dos centrales entre sí. Este término se define en el anexo A/E.410. En la gestión de tráfico, un haz de circuitos no es visible; es decir, no es modelado en el nivel gestión de NE.

4.1.6 subhaz de circuitos: Conjunto de circuitos, pertenecientes a un haz de circuitos, identificables unívocamente por razones técnicas o de explotación. Un haz de circuitos puede constar de uno o más subhaces de circuitos. Este término se define en el anexo A/E.410.

4.1.7 destino: País, área, central u otro lugar, o servicio especial, donde se halla el abonado llamado, y que puede especificarse dentro del país. Un destino se identifica por uno o más indicativos de destino. Este término se define en el anexo A/E.410.

4.1.8 indicativo de destino: Un indicativo de destino se identifica por las cifras utilizadas para analizar y procesar la llamada. Puede definirse con el grado de precisión que se necesite. Este término se define en el anexo A/E.410.

4.2 Encaminamiento directo

4.2.1 tráfico con encaminamiento directo: Conjunto de llamadas ofrecidas para las que un subhaz de circuitos es la primera elección.

4.2.2 toma: Intento de toma de circuito de un haz que culmina en la obtención de un circuito de dicho haz. Este término se define en el anexo A/E.410.

4.2.3 nodo de conmutación: Central representada por un ejemplar de un elemento gestionado o una subclase del elemento gestionado.

5 Símbolos y abreviaturas

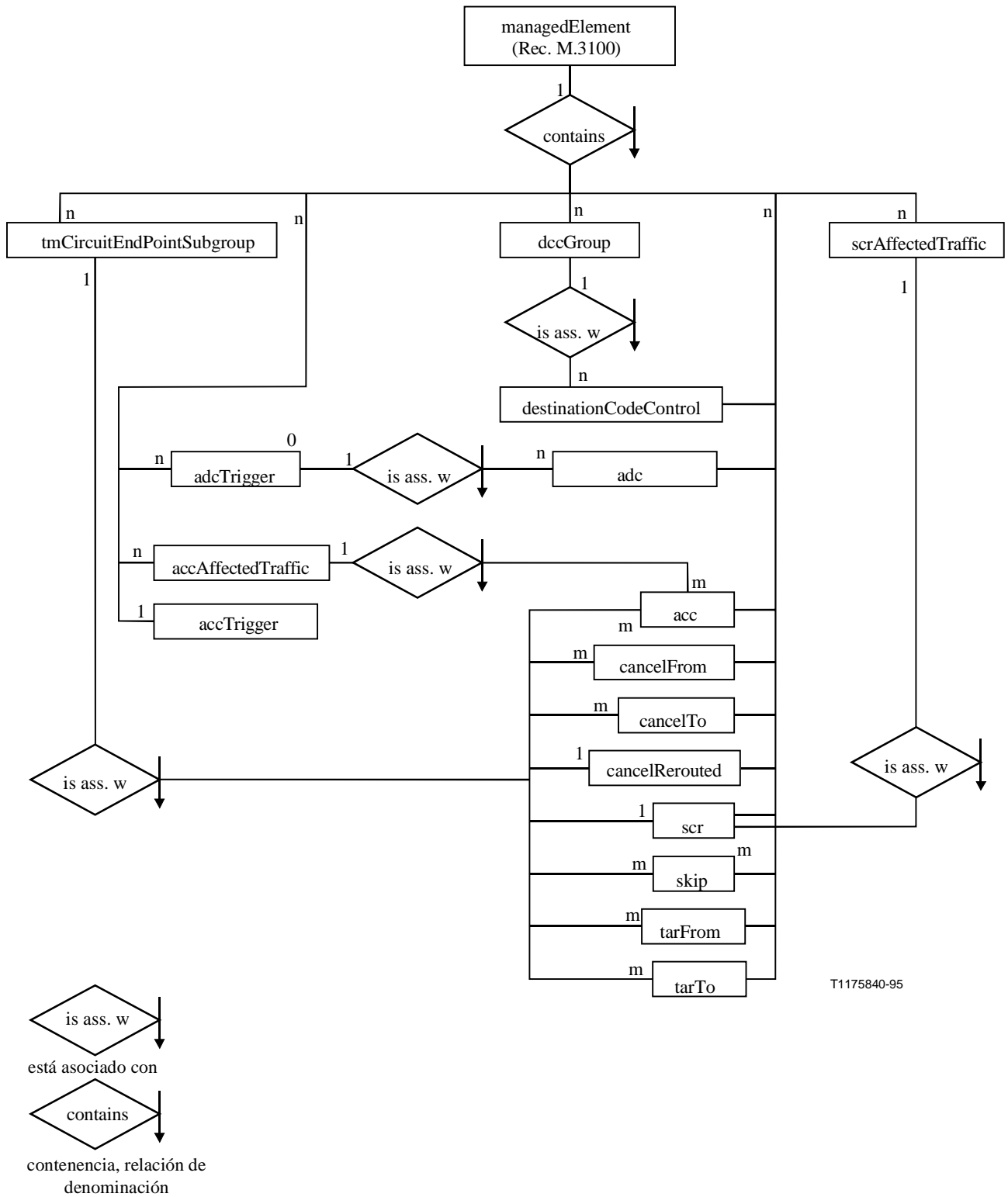
La siguiente es una lista de símbolos y abreviaturas utilizados en esta Recomendación, que no incluye la clase de objeto gestionado ni los nombres de atributo que aparecen en las definiciones formales de esta Recomendación.

ABR	Tasa de intentos de toma con respuesta (<i>answer bid ratio</i>)
ACC	Control automático de congestión (<i>automatic congestion control</i>)
ADC	Control automático de destino (<i>automatic destination control</i>)
ASR	Tasa de tomas con respuesta (<i>answer seizure ratio</i>)
HTR	Difícil de alcanzar (<i>hard-to-reach</i>)
Id	Identificador (<i>identifier</i>)
M/C/O	Obligatorio/condicional/opcional (<i>mandatory/conditional/optional</i>)
M/S	Multivalor/univalor (<i>multivalued/single-valued</i>)
MIB	Base de información de gestión (<i>management information base</i>)
NA	Aspectos de red (<i>network aspects</i>)
NE	Elemento de red (<i>network element</i>)
OA&M	Operación, administración y mantenimiento (<i>operation, administration and maintenance</i>)
OS	Sistema de operaciones (<i>operations system</i>)
OSI	Interconexión de sistemas abiertos (<i>open systems interconnection</i>)
RDSI	Red digital de servicios integrados
RTPC	Red telefónica pública con conmutación
RDN	Nombre distinguido relativo (<i>relative distinguished name</i>)

6 Modelo de información

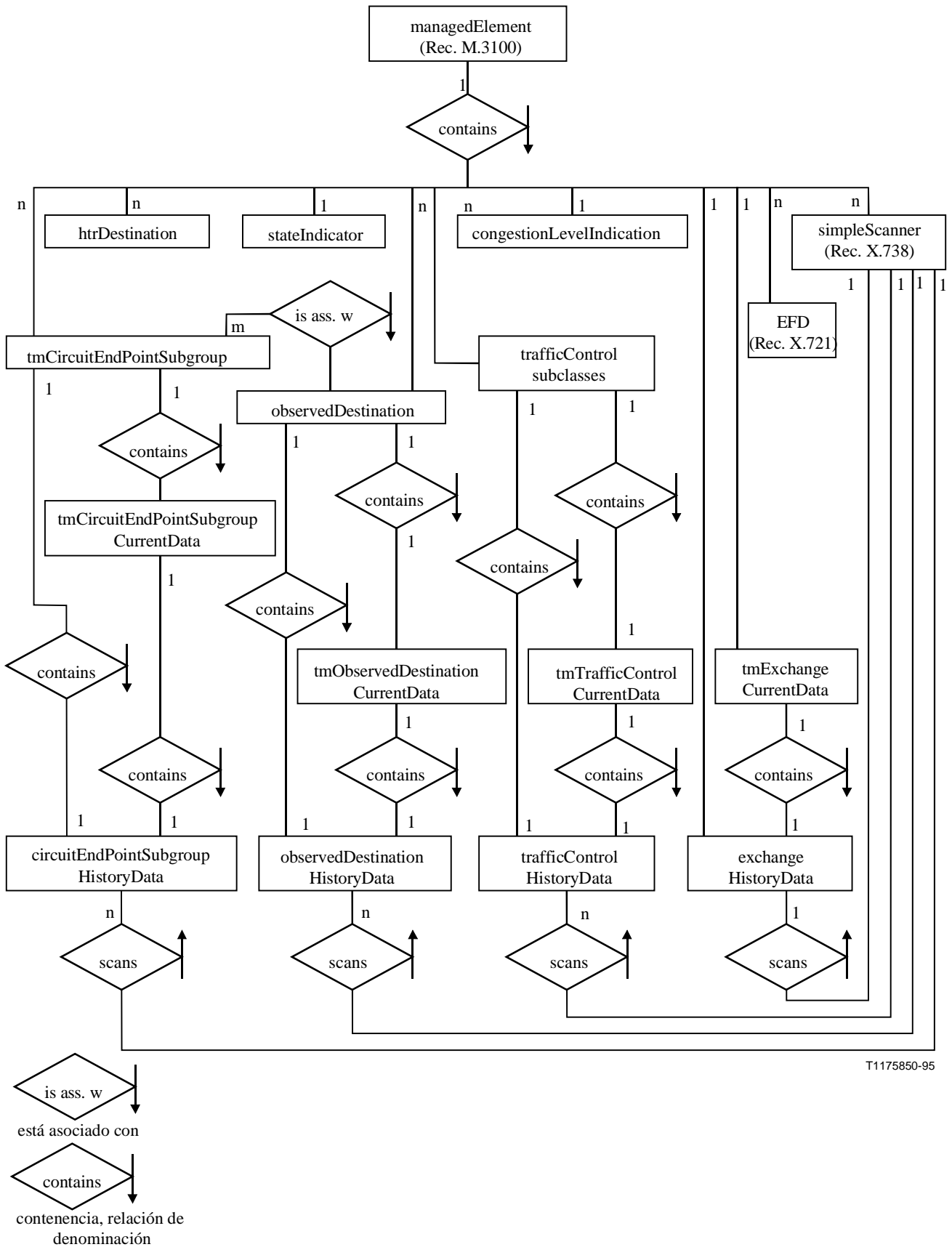
6.1 Diagrama de relaciones entre entidades

Véanse las figuras 6-1 y 6-2.



T1175840-95

Figura 6-1/Q.823 – Diagrama de relaciones entre entidades para el fragmento de control

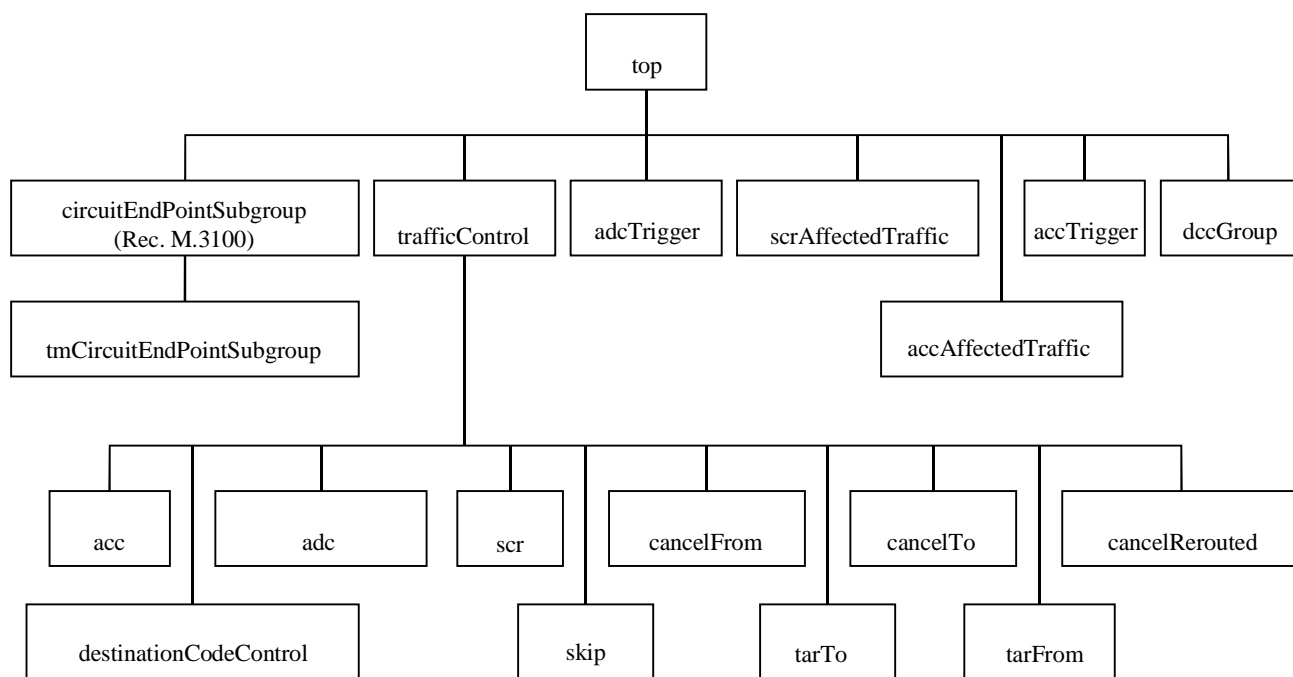


T1175850-95

Figura 6-2/Q.823 – Diagrama de relaciones entre entidades para los fragmentos de elemento gestionado de calidad de funcionamiento, de estado y administrativo

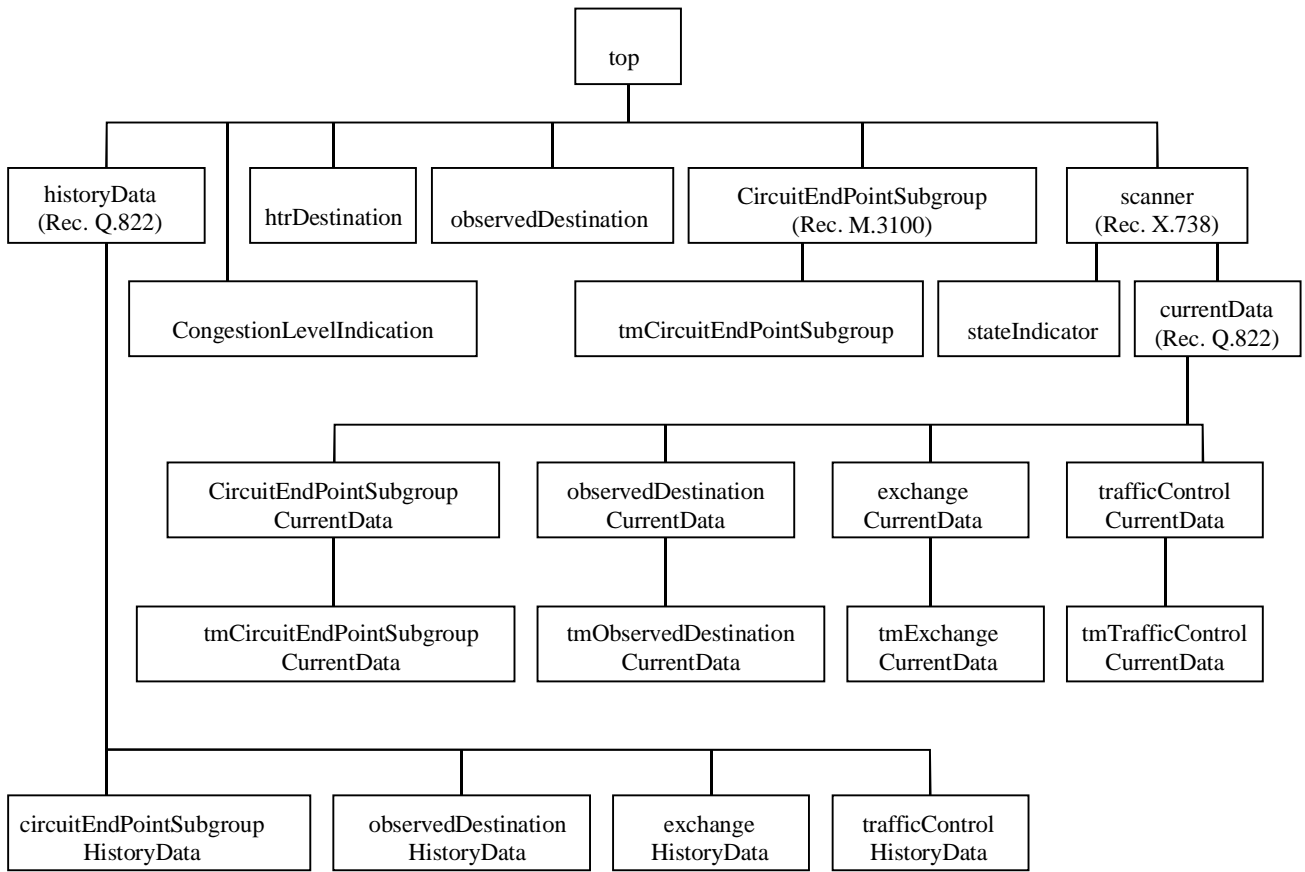
6.2 Jerarquía de herencia

Véanse las figuras 6-3 y 6-4.



T1175860-95

Figura 6-3/Q.823 – Jerarquía de herencia para el fragmento de control

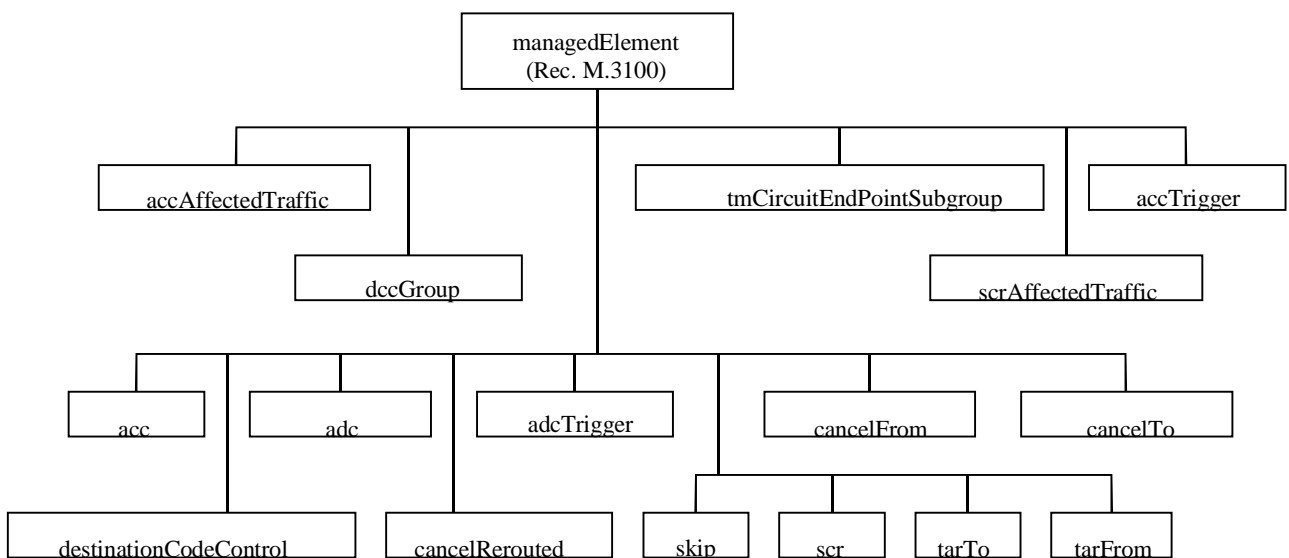


T1175870-95

Figura 6-4/Q.823 – Jerarquía de herencia para los fragmentos de elemento gestionado, de calidad de funcionamiento, de estado y administrativo

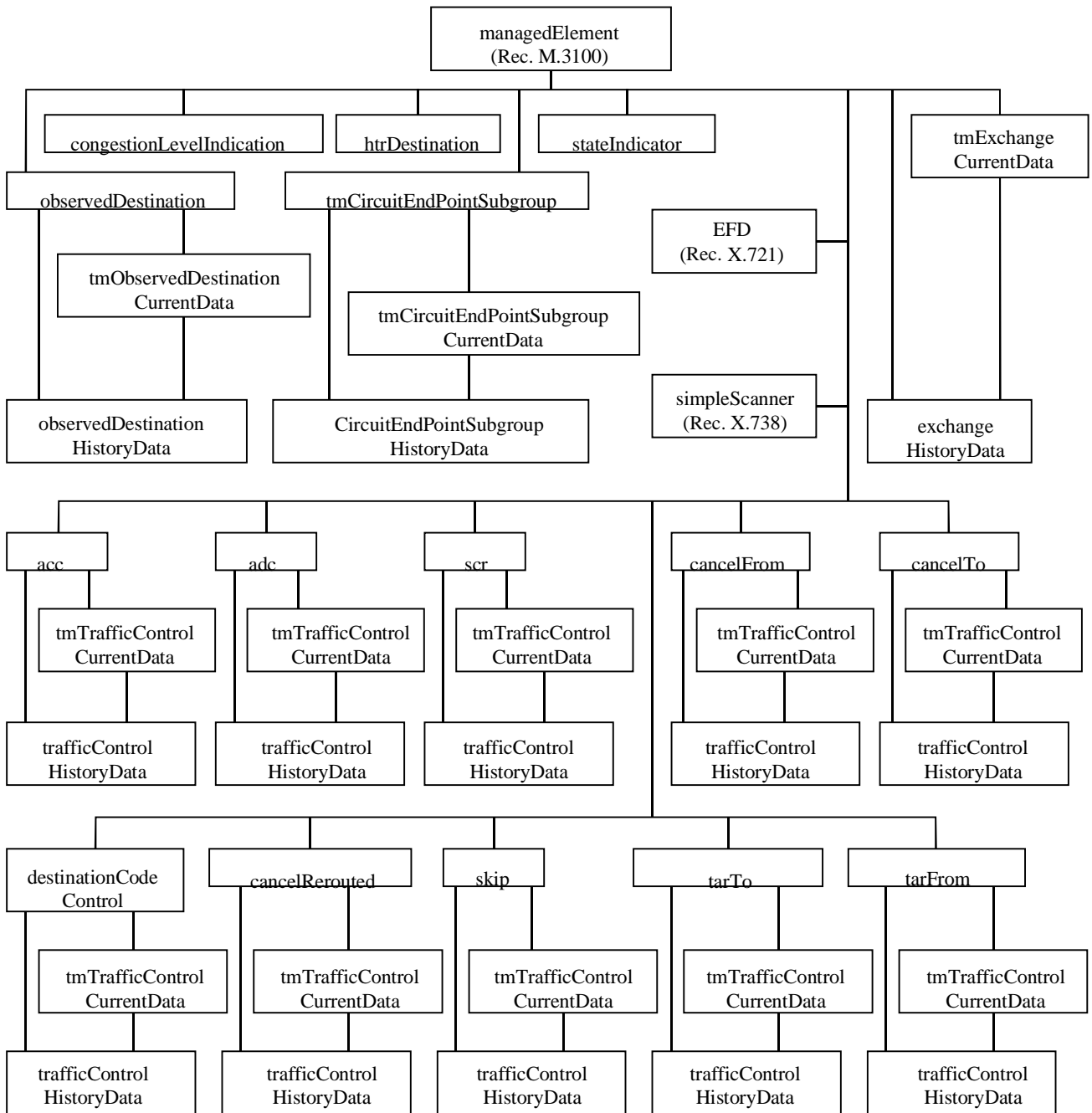
6.3 Jerarquía de denominación

Véanse las figuras 6-5 y 6-6.



T1175880-95

Figura 6-5/Q.823 – Vinculación de nombres para el fragmento de control



T1175890-95

Figura 6-6/Q.823 – Vinculación de nombres para los fragmentos de elemento gestionado, de calidad de funcionamiento, de estado y administrativo

7 Descripción del modelo de información

Esta cláusula presenta una descripción informal de alto nivel del modelo de información de gestión de tráfico.

La subcláusula 7.1 contiene una breve descripción de cada clase de objeto utilizado en el modelo de información y proporciona la información siguiente:

- la finalidad de la clase de objeto,

- los atributos definidos para la clase de objeto, y
- la relación de la clase de objeto con otras clases de objetos.

Los atributos que son comunes a varias clases de objeto se describen en 7.2.

En la subcláusula 7.3 se describen acciones que influyen varias clases de objetos en el modelo de información.

En la subcláusula 7.4 se describen los aspectos comunes de las notificaciones usadas en el modelo de información.

En los cuadros que enumeran los atributos de las clases de objetos, no se mencionan explícitamente los atributos heredados de la clase top (tope) de la "Recomendación X.721", aunque están presentes en estos objetos por herencia.

7.1 Descripción de clases de objetos

Esta subcláusula se subdivide en otras en las que se describen las clases de objetos del modelo de información, en la medida en que no están definidas en otras Recomendaciones. Para clases de objetos definidas en otras Recomendaciones, se especifica la referencia adecuada.

7.1.1 Fragmento de elemento gestionado

Este fragmento proporciona la definición de los recursos asociados con la central.

7.1.1.1 Indicación de nivel de congestión (congestionLevelIndication)

La indicación de nivel de congestión da una indicación del nivel de congestión actual de la instancia de objeto elemento gestionado en la que está contenida. No se permite a esta instancia de objeto ser instanciada por un OS. Véase el cuadro 1.

Cuadro 1/Q.823 – Entidad congestionLevelIndication

Atributos	M/S	O/M/C
congestionLevelIndicationId	S	M
congestionLevel	S	M

Los atributos siguientes definen la entidad congestionLevelIndication:

- congestionlevelindicationid – Este atributo se utiliza como el RDN.
- congestionlevel – Indica la situación de congestión actual en una central. Es expresado por los siguientes niveles de congestión de máquina (MCL, *machine congestion levels*):
 - MCL0: No hay congestión de central
 - MCL1: Congestión de central moderada, la central sigue funcionando. Algunas llamadas pueden ser rechazadas.
 - MCL2: Nivel de congestión grave; la central ha dejado de ser capaz de tratar todo el tráfico ofrecido. Se rechaza un número de llamadas mayor que en MCL1.
 - MCL3: Incapaz de procesar llamadas. Aunque es deseable que lo hagan, algunos NE pueden no ser capaces de proporcionar un indicador de nivel 3 durante los fallos catastróficos.

7.1.1.2 Destino difícil de alcanzar (htrDestination)

Una instancia del objeto htrDestination representa un destino identificado como difícil de alcanzar. La decisión de si un destino es difícil o fácil de alcanzar es tomada sobre la base de información exterior (por ejemplo, un terremoto) o de la tasa de intentos de toma con respuesta o la tasa de tomas con respuesta por el OS o por la gestión de recursos de la central. El atributo administrativeState ofrece la oportunidad de bloquear la situación difícil de alcanzar para que pueda temporalmente no ser tomada en cuenta. La situación HTR de un destino puede también ser correlacionada con subhaces de circuitos. Si este atributo es un conjunto vacío, el destino se considera HTR vía todos los posibles subhaces de circuitos.

Una instancia de htrDestination puede ser explícitamente creada por un gestor (en el caso de destino htr manual) o ser automáticamente creada por el NE (en caso de determinación automática por el agente).

En este modelo, el mecanismo para el reconocimiento de la situación difícil de alcanzar de un destino por la gestión de recursos de la central es un asunto local, y por tanto no es modelado.

Un destino para el cual no hay instanciado ningún htrDestination o que es inhibido [administrativeState = bloqueado (locked)] no se considera difícil de alcanzar.

Todas las instancias de la clase de objeto htrDestination forman la lista HTR. Véase el cuadro 2.

Cuadro 2/Q.823 – Entidad htrDestination

Atributos	M/S	O/M/C
htrDestinationId	S	M
creatorIdentity	S	M
destinationType	S	C
destinationCode	S	M
administrativeState	S	M
tmCircuitEndPointSubgroupList	M	C

Los siguientes atributos describen la entidad htrDestination:

- htrDestinationId (id de destino htr) – Este atributo se utiliza como el RDN.
- creatorIdentity (id del creador) – Identifica quién creó la instancia de objeto.
- destinationType (tipo de destino) – Identifica el tipo de destino. Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Es un indicativo de país, o/y un indicativo de área, o/y una central, u/y otro número de lugar.
- administrativeState (estado administrativo) – Indica si el destinationCode ha sido inhibido de la situación HTR. Si el valor es 'bloqueado (locked)', es inhibido; si el valor es 'desbloqueado (unlocked)', el indicativo no es inhibido. El valor por defecto es 'desbloqueado'.
- tmCircuitEndPointSubgroupList (lista de subgrupos de punto extremo de circuitos de gestión de tráfico) – Identifica una lista de subhaces de circuitos de los cuales el destinationCode es HTR.

7.1.1.3 Elemento gestionado

La clase de objeto gestionado elemento gestionado se define en la Recomendación M.3100.

7.1.1.4 Subhaz de circuitos de gestión de tráfico (tmCircuitEndPointSubgroup)

El tmCircuitEndPointSubgroup es una subclase del circuitEndPointSubgroup de la M.3100. Se utiliza para supervisión de calidad de funcionamiento y controles con fines de gestión de tráfico. Véase el cuadro 3.

Cuadro 3/Q.823 – Entidad tmCircuitEndPointSubgroup

Atributos	M/S	O/M/C
tmSurveillance	S	M

Los siguientes atributos describen la entidad tmCircuitEndPointSubgroup:

- tmSurveillance (vigilancia de gestión de tráfico) – Indica si el tmCircuitEndPointSubgroup está siendo vigilado con fines de gestión de tráfico. Cuando el valor es VERDADERO (TRUE), el NE instanciará automáticamente la correspondiente tmCircuitEndPointSubgroupCurrentData. Cuando el valor es FALSO (FALSE), el NE suprimirá automáticamente la correspondiente tmCircuitEndPointSubgroupCurrentData.

7.1.2 Fragmento de supervisión de calidad de funcionamiento

A fin de realizar las funciones de gestión de tráfico, se recogerán periódicamente datos de calidad de funcionamiento acerca de los siguientes recursos gestionados: subhaz de circuitos, destino, central y las instancias de control de tráfico. Los datos de calidad de funcionamiento se recogerán aplicando el modelo Q.822 (tal como currentData e historyData) y el simpleScanner de la Recomendación X.738. El apéndice V proporciona información práctica sobre el modo de reunir datos de calidad de funcionamiento mediante el mecanismo petición o el autónomo.

7.1.2.1 Datos actuales

La clase de objeto gestionado datos actuales se define en la Recomendación Q.822. Esta Recomendación expone las subclases que existen para las diferentes entidades supervisadas cuando se definen los atributos a supervisar. En esta Recomendación, se define un conjunto mínimo de atributos. Para implementación específica o individual de la tecnología que exija más que el conjunto mínimo, se recomienda otra subclasificación más de la clase currentData.

7.1.2.2 Datos actuales de subhaz de punto extremo de circuitos (CircuitEndPointSubgroupCurrentData)

Datos actuales de subhaz de circuitos es una subclase de la clase de objeto currentData. Se utiliza para supervisar datos de calidad de funcionamiento relativos al subhaz de circuitos definidos en la Recomendación E.502.

Los atributos de supervisión de calidad de funcionamiento para el subhaz de circuitos se basan en la característica de direccionalidad del subhaz de circuitos, que puede ser unidireccional de salida, unidireccional de entrada o bidireccional.

Para fines de gestión de tráfico, esta clase de objeto no será instanciada, pero sí su subclase definida en esta Recomendación o cualquier clase derivada de ella que pueda ser instanciada. Véase el cuadro 4.

Cuadro 4/Q.823 – Entidad circuitEndPointSubgroupCurrentData

Atributos	M/S	O/M/C
incomingSeizures	S	M
outgoingBids	S	M
outgoingSeizures	S	M
answeredOutgoingSeizures	S	M
answeredIncomingSeizures	S	O
overflow	S	M
incomingTrafficUsage	S	M
outgoingTrafficUsage	S	M
numberOfAvailCircuits	S	M

Los siguientes atributos definen la entidad circuitEndPointSubgroupCurrentData:

- incomingSeizures (tomas entrantes) – Identifica el número de tomas entrantes en el subhaz de circuitos [4.2/E.502, tipo 21 b)]. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales.
- outgoingBids (intentos de toma salientes) – Identifica el número de intentos fructuosos o infructuosos de tomar un circuito del subhaz de circuitos [4.2/E.502, tipo 21 a)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. Incluye intentos de toma previos a la aplicación de cualesquiera controles NM.
- outgoingSeizures (tomas salientes) – Identifica el número de intentos de toma salientes que consiguieron obtener un circuito dentro del subhaz de circuitos [4.2.6/E.502, tipo 21 b)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales.
- answeredOutgoingSeizures (tomas salientes con respuesta) – Identifica el número de tomas salientes en las que se recibió una señal de respuesta [4.2/E.502, tipo 21 c)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales.
- answeredIncomingSeizures (tomas entrantes con respuesta) – Identifica el número de tomas entrantes en las que se transmitió una señal de respuesta hacia atrás a la señal precedente. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales.
- overflow (desbordamiento) – Identifica el número de intentos de toma salientes que desbordan de este subhaz de circuitos [4.2.6/E.502, tipo 21 d)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. No incluirá llamadas afectadas por cancelación del desbordamiento reencaminado, encaminamiento alternativo temporal desde, o cancelación desde.
- incomingTrafficUsage (utilización de tráfico entrante) – Identifica el tráfico entrante cursado, en erlangs.segundo [4.2.6/E.502, tipo 21 e)]. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales. Suele proporcionarse mediante el método de muestreo.
- outgoingTrafficUsage (utilización de tráfico saliente) – Identifica el tráfico saliente cursado en erlang.segundos. [4.2.6/E.502, tipo 21 e)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. Suele proporcionarse mediante el método de muestreo.
- numberOfAvailCircuits (número de circuitos disponibles) – Identifica el número de circuitos que pueden cursar tráfico incluyendo los que lo hacen en ese momento [4.2.6/E.502, tipo 21 f)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes,

unidireccionales entrantes y bidireccionales. Se deja a criterio de la implementación si este valor es proporcionado como una instantánea o como valor medio debido a la elevada frecuencia de cambios que se registra normalmente en los circuitos. Ambos métodos son equivalentes.

7.1.2.3 Datos actuales de la central (exchangeCurrentData)

Datos actuales de la central es una categoría de la clase de objeto currentData. Se utiliza para supervisar los datos de calidad de funcionamiento relativos a la central definidos en la Recomendación E.502.

Para fines de gestión de tráfico, esta clase de objeto no será instanciada, pero sí su subclase definida en esta Recomendación o cualquier subclase derivada de ella.

Los atributos de supervisión de calidad de funcionamiento para el intercambio se basan en las principales características de flujo de tráfico definidas en la figura 4/E.502. Véase el cuadro 5.

Cuadro 5/Q.823 – Entidad exchangeCurrentData

Atributos	M/S	O/M/C
incomingTraffic	S	M
outgoingTraffic	S	M
transitTraffic	S	M
terminatingTraffic	S	M
originatingTraffic	S	M
internalTraffic	S	M
callsBlockedByLoadShedding	S	M

Los siguientes atributos definen la entidad exchangeCurrentData:

- incomingTraffic (tráfico entrante) – Identifica el número de llamadas entrantes en la central de las cuales se ha acusado al menos la recepción de un dígito [figura 4/E.502, flujo de tráfico B, 4.2/E.502, tipo 20 a), tipo 4 a) y apéndice I].
- outgoingTraffic (tráfico saliente) – Identifica el número de llamadas salientes (tomas) de la central que han conseguido la toma de un circuito [figura 4/E.502, flujo de tráfico R, 4.2/E.502, tipo 10 a), apéndice I].
- transitTraffic (tráfico de tránsito) – Identifica el número de llamadas de tránsito (tomas) en la central (figura 4/E.502, flujo de tráfico L, 4.2/E.502, tipo 6 a)].
- terminatingTraffic (tráfico de destino) – Identifica el número de llamadas de destino (tomas) (a líneas) en la central [figura 4/E.502, flujo de tráfico Q, 4.2/E.502, tipo 12 d)].
- originatingTraffic (tráfico de origen) – Identifica el número de llamadas de origen (tomas) en la central [figura 4/E.502, flujo de tráfico A, 4.2/E.502, tipo 1 a)].
- internalTraffic (tráfico interno) – Identifica el número de llamadas internas (tomas) en la central [figura 4/E.502, flujo de tráfico F, 4.2/E.502, tipo 2 a)].
- callsBlockedByLoadShedding (llamada bloqueadas por desbordamiento de carga) – Identifica el número de llamadas que no pueden ser tratadas debido a la aplicación de un mecanismo interno de protección de carga de la central [4.2/E.502, tipo 20 g)].

7.1.2.4 Datos actuales del destino observado (observedDestinationCurrentData)

Datos actuales del destino observado es una subclase de la clase de objeto currentData. Se utiliza para supervisar datos de calidad de funcionamiento relativos al destino definidos en la Recomendación E.502.

Para fines de gestión de tráfico, esta clase de objeto no será instanciada, pero sí su subclase definida en esta Recomendación o cualquier subclase derivada de ella. Véase el cuadro 6.

Cuadro 6/Q.823 – Entidad observedDestinationCurrentData

Atributos	M/S	O/M/C
bids	S	M
outgoingSeizures	S	M
answeredOutgoingSeizures	S	M
noCircuitsAvailable	S	M
callsAffectedByDcc	S	O

Los siguientes atributos definen la entidad observedDestinationCurrentData:

- bids (intentos de toma) – Identifica al número de intentos de toma al destino observado [4.2/E.502, tipo 22 a)].
- outgoingSeizures (tomas salientes) – Identifica el número de intentos de toma fructuosos al destino observado [4.2/E.502, tipo 22 b)].
- answeredOutgoingSeizures (tomas salientes con respuesta) – Identifica el número de tomas salientes al destino observado en las que se recibió una señal de respuesta [4.2/E.502, tipo 22 c)].
- noCircuitsAvailable (no hay circuitos disponibles) – Identifica el número de intentos de toma que producen llamadas infructuosas porque no había disponibles circuitos libres al destino observado; es decir, desbordamiento resultante en el subhaz de circuitos final a ese destino. En general, $noCircuitsAvailable \leq (bids - outgoingSeizures)$ [4.2.6/E.502, tipo 22 d)].
- callsAffectedByDcc (llamadas afectadas por control de indicativo de destino) – Identifica el número de llamadas al destino observado que han sido afectadas por la función control de indicativo de destino [4.2/E.502, tipo 22 e)].

7.1.2.5 Datos actuales de control de tráfico (trafficControlCurrentData)

La clase de objeto datos actuales de control de tráfico es una subclase de la clase de objeto currentData definida en la Recomendación Q.822.

Se utiliza para supervisar la eficacia de un control de tráfico, definida en la Recomendación E.502.

La trafficControlCurrentData ha de proporcionarse para cada instancia de control. Si existe necesidad de proporcionar trafficControlCurrentData por la clase de objeto de control, o de indicar cómo un control repercute en una entidad gestionada, puede ser agregada en el OS desde la instancia de control correspondiente de trafficControlCurrentData.

Para fines de gestión de tráfico, esta clase de objeto no será instanciada, pero sí su subclase en esta Recomendación o cualquier subclase derivada de ella. Véase el cuadro 7.

Cuadro 7/Q.823 – Entidad trafficControlCurrentData

Atributos	M/S	O/M/C
callsOfferedToTrafficControl	S	O
callsAffectedByTrafficControl	S	M

Los siguientes atributos definen la entidad trafficControlCurrentData:

- callsOfferedToTrafficControl (llamadas ofrecidas al control de tráfico) – Identifica el número de llamadas que se ofrecieron a la instancia de control de tráfico.
- callsAffectedByTrafficControl (llamadas afectadas por el control de tráfico) – Identifica el número de llamadas que son realmente afectadas por la instancia de control de tráfico.

7.1.2.6 Datos actuales de subhaz de circuitos de gestión de tráfico (tmCircuitEndPointSubgroupCurrentData)

La clase de objeto datos actuales de subhaz de circuitos de gestión de tráfico es una subclase de la clase de objeto circuitEndPointSubgroupCurrentData.

Se utiliza para supervisar los datos de calidad de funcionamiento relativos al subhaz de circuitos definidos en la Recomendación E.502 en el contexto de la gestión de tráfico. Esta clase de objeto o sus subclases son instanciadas para fines de gestión de tráfico.

Una instancia de objeto tmCircuitEndPointSubgroupCurrentData generará sólo una instancia de tmCircuitEndPointSubgroupHistoryData.

Todas las instancias de objeto tmCircuitEndPointSubgroupCurrentData dentro del mismo elemento gestionado tendrán el mismo periodo de granularidad.

A fin de sincronizar el granularityPeriod al siguiente periodo de tiempo integral después de que se instancien las subclases currentData, se recomienda que el valor del periodSynchronizationTime (atributo de periodSynchronizationPackage) se ponga a las doce horas de medianoche (de acuerdo con la hora de la central). Véase el cuadro 8.

Cuadro 8/Q.823 – Entidad tmCircuitEndPointSubgroupCurrentData

Atributos	M/S	O/M/C
historyRetention	S	M

Los siguientes atributos definen la entidad tmCircuitEndPointSubgroupCurrentData:

- historyRetention (retención de datos históricos) – Definido en la Recomendación Q.822. Para el fin de la gestión de tráfico, su valor permitido es "1".

7.1.2.7 Datos actuales de central de gestión de tráfico (tmExchangeCurrentData)

La clase de objeto datos actuales de calidad de funcionamiento de central de gestión de tráfico es una subclase de la clase de objeto exchangeCurrentData.

Se utiliza para supervisar los datos de calidad de funcionamiento relativos a la central definidos en la Recomendación E.502 en el contexto de la gestión de tráfico. Esta clase de objeto o sus subclases son instanciados para fines de gestión de tráfico.

Una instancia de objeto tmExchangeCurrentData generará sólo una instancia de tmExchangePerformanceHistoryData.

A fin de sincronizar el granularityPeriod al siguiente periodo de tiempo integral después de que se instancien las subclases currentData, se recomienda que el valor del periodSynchronizationTime (atributo del periodSynchronizationPackage) se ponga a las doce de medianoche (de acuerdo con la hora de la central). Véase el cuadro 9.

Cuadro 9/Q.823 – Entidad tmExchangeCurrentData

Atributos	M/S	O/M/C
historyRetention	S	M

Los siguientes atributos definen la entidad tmExchangeCurrentData:

- historyRetention (retención de datos históricos) – Definido en la Recomendación Q.822. Para el fin de la gestión de tráfico, su valor permitido es "1".

7.1.2.8 Datos actuales del destino observado de gestión de tráfico (tmObservedDestinationCurrentData)

La clase de objeto datos actuales del destino de gestión de tráfico es una subclase de la clase de objeto observedDestinationCurrentData.

Se utiliza para supervisar los datos de calidad de funcionamiento relativos al destino observado definidos en la Recomendación E.502 en el contexto de la gestión de tráfico. Esta clase de objeto o sus subclases son instanciados para fines de gestión de tráfico.

Una instancia de objeto tmObservedDestinationCurrentData generará sólo una instancia de tmObservedDestinationHistoryData.

Todas las instancias de objeto tmDestinationCurrentData dentro del mismo elemento gestionado tendrán el mismo periodo de granularidad.

A fin de sincronizar el granularityPeriod al siguiente periodo de tiempo integral después de que se instancien las subclases currentData, se recomienda que el valor del periodSynchronizationTime (atributo de periodSynchronizationPackage) se ponga a las doce horas de medianoche (de acuerdo con la hora de la central). Véase el cuadro 10.

Cuadro 10/Q.823 – Entidad tmObservedDestinationCurrentData

Atributos	M/S	O/M/C
historyRetention	S	M

Los siguientes atributos definen la entidad tmObservedDestinationCurrentData:

- historyRetention (retención de datos históricos) – Definido en la Recomendación Q.822. Para el fin de la gestión de tráfico, su valor permitido es "1".

7.1.2.9 Datos actuales de control de tráfico de gestión de tráfico (tmTrafficControlCurrentData)

La clase de objeto datos actuales de control de tráfico de gestión de clase es una subclase de la clase de objeto trafficControlCurrentData.

Se utiliza para supervisar la eficacia de un control de tráfico definida en la Recomendación E.502 en el contexto de gestión de tráfico. Esta clase de objeto o sus subclases son instanciadas para fines de gestión de tráfico.

Una instancia de objeto `tmTrafficControlCurrentData` generará sólo una instancia de `tmTrafficControlHistoryData`. Todas las instancias de objeto `tmTrafficControlCurrentData` dentro del mismo elemento gestionado tendrán el mismo periodo de granularidad.

A fin de sincronizar el `granularityPeriod` al siguiente periodo de tiempo integral después de que se instancien las subclases `currentData`, se recomienda que el valor del `periodSynchronizationTime` (atributo del `periodSynchronizationPackage`) se ponga a las doce horas de medianoche (de acuerdo con la hora de la central). Véase el cuadro 11.

Cuadro 11/Q.823 – Entidad `tmTrafficControlCurrentData`

Atributos	M/S	O/M/C
<code>historyRetention</code>	S	M

Los siguientes atributos definen la entidad `tmTrafficControlCurrentData`:

- `historyRetention` (retención de datos históricos) – Definido en la Recomendación Q.822. Para el fin de la gestión de tráfico, su valor permitido es "1".

7.1.2.10 Datos históricos (`historyData`)

La clase de objeto gestionado datos históricos se define en la Recomendación Q.822.

Para cada una de las entidades supervisadas, las instancias `historyData` contienen una copia de los atributos de calidad de funcionamiento que están presentes en las correspondientes instancias de objeto de subclases `currentData` al final del periodo de granularidad.

En línea con la Recomendación Q.822, se soportan dos tipos de vinculación de nombres para datos históricos en esta Recomendación: `historyData` a `currentData` e `historyData` a entidad supervisada. Cualquiera de estas dos vinculaciones de nombres pueden ser utilizadas para fines de gestión de tráfico, pero la vinculación de nombres debe utilizarse de manera consecuente para todas las instancias de `historyData` dentro de un sistema gestionado; es decir, todas serán denominadas desde la `currentData` y sus subclases o todas serán denominadas desde la entidad supervisada y sus subclases.

7.1.2.11 Datos históricos de subhaz de punto extremo de circuitos (`circuitEndPointSubgroupHistoryData`)

Datos históricos `circuitEndPointSubgroup` de gestión de tráfico es una subclase de la clase de objeto `historyData`. Se utiliza para supervisar los datos de calidad de funcionamiento relativos al `circuitSubgroup` definidos en la Recomendación E.502.

Los atributos de supervisión de calidad de funcionamiento son los mismos que los de la correspondiente `currentData`. Véase el cuadro 12.

Cuadro 12/Q.823 – Entidad circuitEndPointSubgroupHistoryData

Atributos	M/S	O/M/C
incomingSeizures	S	M
outgoingBids	S	M
outgoingSeizures	S	M
answeredOutgoingSeizures	S	M
answeredIncomingSeizures	S	C
overflow	S	M
incomingTrafficUsage	S	M
outgoingTrafficUsage	S	M
numberOfAvailCircuits	S	M

Los siguientes atributos definen la entidad circuitEndPointSubgroupHistoryData:

- incomingSeizures (tomas entrantes) – Identifica el número de tomas entrantes en el subhaz de circuitos [4.2/E.502, tipo 21 b)]. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales.
- outgoingBids (intentos de toma salientes) – Identifica el número de intentos fructuosos o infructuosos de tomar un circuito del subhaz de circuitos [4.2/E.502, tipo 21 a)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. Incluye intentos de toma previos a la aplicación de cualesquiera controles NM.
- outgoingSeizures (tomas salientes) – Identifica el número de intentos de toma salientes que consiguieron obtener un circuito dentro del subhaz de circuitos [4.2.6/E.502, tipo 21 b)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales.
- answeredOutgoingSeizures (tomas salientes con respuesta) – Identifica el número de tomas salientes en las que se recibió una señal de respuesta [4.2/E.502, tipo 21 c)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales.
- answeredIncomingSeizures (tomas entrantes con respuesta) – Identifica el número de tomas entrantes en las que se transmitió una señal de respuesta hacia atrás a la señal precedente. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales.
- overflow (desbordamiento) – Identifica el número de intentos de toma salientes que desbordan de este subhaz de circuitos [4.2.6/E.502, tipo 21 d)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. No incluirá llamadas afectadas por cancelación del desbordamiento reencaminado, encaminamiento alternativo temporal desde, o cancelación desde.
- incomingTrafficUsage (utilización de tráfico entrante) – Identifica el tráfico entrante cursado, en erlangs.segundo [4.2.6/E.502, tipo 21 e)]. Esta cuenta es válida para subhaces de circuitos unidireccionales entrantes o bidireccionales. Suele proporcionarse mediante el método de muestreo.
- outgoingTrafficUsage (utilización de tráfico saliente) – Identifica el tráfico saliente cursado en erlang.segundos [4.2.6/E.502, tipo 21 e)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes o bidireccionales. Suele proporcionarse mediante el método de muestreo.
- numberOfAvailCircuits (número de circuitos disponibles) – Identifica el número de circuitos que pueden cursar tráfico incluidos los que lo hacen en ese momento [4.2.6/E.502, tipo 21 f)]. Esta cuenta es válida para subhaces de circuitos unidireccionales salientes,

unidireccionales entrantes y bidireccionales. Se deja a criterio de la implementación si este valor es proporcionado como una instantánea o como valor medio debido a la elevada frecuencia de cambios que se registra normalmente en los circuitos. Ambos métodos son equivalentes.

7.1.2.12 Datos históricos de la central (exchangeHistoryData)

Datos históricos de la central es una subclase de la clase de objeto historyData. Se utiliza para supervisar los datos de calidad de funcionamiento relativos a la central definidos en la Recomendación E.502.

Los atributos de supervisión de calidad de funcionamiento son los mismos que los de la correspondiente currentData.

Los atributos de supervisión de calidad de funcionamiento para el intercambio se basan en las principales características de flujo de tráfico definidas en la figura 4/E.502. Véase el cuadro 13.

Cuadro 13/Q.823 – Entidad exchangeHistoryData

Atributos	M/S	O/M/C
incomingTraffic	S	M
outgoingTraffic	S	M
transitTraffic	S	M
terminatingTraffic	S	M
originatingTraffic	S	M
internalTraffic	S	M
callsBlockedByLoadShedding	S	M

Los siguientes atributos definen la entidad exchangeHistoryData:

- incomingTraffic (tráfico entrante) – Identifica el número de llamadas entrantes en la central de las cuales se ha acusado al menos la recepción de un dígito [figura 4/E.502, flujo de tráfico B, 4.2/E.502, tipo 20 a), tipo 4 a) y apéndice I).
- outgoingTraffic (tráfico saliente) – Identifica el número de llamadas salientes (tomas) de la central que han conseguido la toma de un circuito [figura 4/E.502, flujo de tráfico R, 4.2/E.502, tipo 10 a), apéndice I).
- transitTraffic (tráfico de tránsito) – Identifica el número de llamadas de tránsito (tomas) en la central [figura 4/E.502, flujo de tráfico L, 4.2/E.502, tipo 6 a)].
- terminatingTraffic (tráfico de destino) – Identifica el número de llamadas de destino (tomas a líneas) en la central [figura 4/E.502, flujo de tráfico Q, 4.2/E.502, tipo 12 d)].
- originatingTraffic (tráfico de origen) – Identifica el número de llamadas de origen (tomas) en la central (figura 4/E.502, flujo de tráfico A, 4.2/E.502, tipo 1 a)].
- internalTraffic (tráfico interno) – Identifica el número de llamadas internas (tomas) en la central (figura 4/E.502, flujo de tráfico F, 4.2/E.502, tipo 2 a)].
- callsBlockedByLoadShedding (llamadas bloqueadas por desbordamiento de carga) – Identifica el número de llamadas que no pueden ser tratadas debido a la aplicación de un mecanismo interno de protección de carga de la central [4.2/E.502, tipo 20 g)].

7.1.2.13 Destino observado (observedDestination)

Una instancia de observedDestination es instanciada cuando ha de ser supervisada para gestión de calidad de funcionamiento, como cuando se determina si es difícil de alcanzar. Un destino es un país, un área, una central u otro lugar en el que está situado el abonado, y que puede especificarse dentro del país. Se define en el anexo A/E.410.

Un destino puede también ser selectivamente observado en un conjunto de subhaces de circuitos. Si la tmCircuitEndPointSubgroupList no está presente, el destino se observará para todos los subhaces de circuitos. Véase el cuadro 14.

Cuadro 14/Q.823 – Entidad observedDestination

Atributos	M/S	O/M/C
observedDestinationId	S	M
creatorIdentity	S	O
destinationType	S	C
destinationCode	S	M
tmCircuitEndPointSubgroupList	M	C
tmSurveillance	S	M

Los siguientes atributos describen la entidad observedDestination:

- observedDestinationId (Id de destino observado) – Este atributo se utiliza como RDN.
- creatorIdentity (identidad del creador) – Indica quién creó la instancia de observedDestination.
- destinationType (tipo de destino) – Identifica el tipo de destino. Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Es un indicativo de país, o/y un indicativo de área, o/y una central, u/y otro número de lugar.
- tmCircuitEndPointSubgroupList (lista de subhaces de punto extremo de circuitos de gestión de tráfico) – Identifica una lista de subhaces de circuitos en los que se observará el destino.
- tmSurveillance (vigilancia de gestión de tráfico) – Indica si está siendo vigilada la instancia observedDestination. Si el valor de atributo es VERDADERO (TRUE), la tmObservedDestinationCurrentData es automáticamente instanciada por el NE. Si el valor de atributo es FALSO (FALSE), la tmObservedDestinationCurrentData es automáticamente suprimida por el NE.

7.1.2.14 Datos históricos del destino observado (observedDestinationHistoryData)

Datos históricos del destino observado son una subclase de la clase de objeto historyData. Se utiliza para supervisar los datos de calidad de funcionamiento relativos al destino definidos en la Recomendación E.502.

Los atributos de supervisión de calidad de funcionamiento son los mismos que los de la correspondiente currentData. Véase el cuadro 15.

Cuadro 15/Q.823 – Entidad observedDestinationHistoryData

Atributos	M/S	O/M/C
bids	S	M
outgoingSeizures	S	M
answeredOutgoingSeizures	S	M
noCircuitsAvailable	S	M
callsAffectedByDcc	S	C

Los siguientes atributos definen la entidad observedDestinationHistoryData:

- bids (intentos de toma)[– Identifica al número de intentos de toma al destino observado [4.2.6/E.502, tipo 22 a)].
- outg[ingSeizures (tomas salientes) – Identifica el número de intentos de toma fructuosos al destino observado [4.2.6/E.502, tipo 22 b)].
- answeredOutgoingSeizures (tomas salientes con respuesta) – Identifica el número de tomas salientes al destino observado en las que se recibió una señal de respuesta [4.2.6/E.502, tipo 22 c)].
- noCircuitsAvailable (no hay circuitos disponibles) – Identifica el número de intentos de toma que producen llamadas infructuosas porque no había disponibles circuitos libres al destino observado; es decir, desbordamiento resultante en el subhaz de circuitos final a ese destino. En general, noCircuitsAvailable <= (bids – outgoingSeizures) [4.2.6/E.502, tipo 22 d)].
- callsAffectedByDcc (llamadas afectadas por control de indicativo de destino) – Identifica el número de llamadas al destino observado que han sido afectadas por la función control de indicativo de destino [4.2/E.502, tipo 22 e)].

7.1.2.15 Datos históricos de control de tráfico (trafficControlHistoryData)

La clase de objeto datos históricos de control de tráfico es una subclase de la clase de objeto historyData definida en la Recomendación Q.822.

Los atributos de supervisión de calidad de funcionamiento son los mismos que en la correspondiente currentData. Véase el cuadro 16.

Cuadro 16/Q.823 – Entidad trafficControlHistoryData

Atributos	M/S	O/M/C
callsOfferedToTrafficControl	S	C
callsAffectedByTrafficControl	S	M

Los siguientes atributos definen la entidad trafficControlHistoryData:

- callsOfferedToTrafficControl (llamadas ofrecidas al control de tráfico) – Identifica el número de llamadas que se ofrecieron a la instancia de control de tráfico.
- callsAffectedByTrafficControl (llamadas afectadas por el control de tráfico) – Identifica el número de llamadas que son realmente afectadas por la instancia de control de tráfico.

7.1.3 Fragmento de control

Se utilizan controles de gestión de tráfico para asegurar una utilización eficaz de la capacidad de red y mantener una calidad de funcionamiento satisfactoria que haga frente a las fluctuantes demandas de tráfico y a las condiciones de emergencia. Pueden ejercerse controles mediante la introducción de información específica procedente de un OS o automáticamente en respuesta a un estímulo interno o externo.

Tanto para controles automáticos como manuales, los datos de calidad de funcionamiento de tráfico son cruciales para permitir determinar cuándo han de aplicarse o suprimirse los controles. Análogamente, a fin de evaluar el efecto que tiene un control sobre el tráfico que transporta capacidad y calidad de funcionamiento de la red, deben obtenerse datos que proporcionen esta información. Así, para cada entidad que pueda ser controlada, por ejemplo, subhaz de circuitos, etc., deben definirse datos de calidad de funcionamiento de tráfico y debe existir para cada control activo una capacidad de medición que mida el efecto del control. Estas capacidades se modelan mediante objetos gestionados de supervisión de calidad de funcionamiento.

Los controles de tráfico manuales y automáticos se modelan en esta Recomendación.

Controles manuales

Los controles manuales son activados y suprimidos mediante acciones de gestión específicas de un sistema de operaciones. En el modelo, la activación y la supresión de los controles manuales se modelan mediante la creación y supresión del objeto gestionado de control. Los objetos gestionados de control no pueden fijarse a un estado desactivado; sin embargo, es posible fijar los parámetros del objeto de control a valores que no repercutan en el volumen de tráfico normal ni en las características de encaminamiento, por ejemplo, el porcentaje de llamadas a bloquear podría fijarse a 0.

Controles automáticos

Los controles automáticos se modelan como controles compuestos por un mecanismo de disparo y un mecanismo de control. El mecanismo de disparo define la condición bajo la cual ha de aplicarse un control y emite automáticamente indicaciones cuando ha de activarse el control. El mecanismo de control, que está preasignado, especifica la naturaleza y la intensidad del objeto a aplicar en respuesta a la indicación de disparo. Estos dos mecanismos se modelan como objetos gestionados. El mecanismo de control es preestablecido en el NE por el gestor de red. Podrá o no permitirse al objeto de disparo ser creado por el gestor de red. El mecanismo de disparo siempre reside en el NE. El mecanismo de disparo y el mecanismo de control pueden estar situados dentro del mismo elemento de red (por ejemplo, ADC descentralizado) o en dos elementos de red diferentes (por ejemplo, ADC centralizado).

Los controles automáticos y los disparadores pueden ser activados o desactivados. Los controles automáticos son activados y desactivados por indicaciones de disparo recibidas. Los controles automáticos y sus mecanismos de disparo pueden también ser inhibidos o desinhibidos en respuesta a acciones de gestión por el OS poniendo el `administrativeState` (estado administrativo) a "bloqueado" (locked) o "desbloqueado" (unlocked), respectivamente.

Cuando un disparador está bloqueado, no emitirá ninguna indicación cuando la condición de disparo resulta verdadera. Esto producirá la activación de cualesquiera nuevos controles automáticos (en este o en cualquier otro elemento de red).

Cuando un control automático está "bloqueado" el control no resultará activado ni siquiera si se recibe una indicación de disparo.

Para permitir que el sistema de operaciones gestor tenga pleno control de los controles automáticos se han introducido dos atributos de situación en todos los objetos de control que pueden ser activados

automáticamente. El atributo `autoActivated` (activado automáticamente) indica si existe actualmente o no un disparador para activar el control. El atributo `administrativeState` (estado administrativo) indica si el control ha sido o no bloqueado por el gestor. Un control automático sólo estará activo si está al mismo tiempo `autoActivated` y desbloqueado. Si un control está bloqueado cuando recibe una indicación de disparo, entrará en la situación `autoActivated`, pero no resultará activo a menos y hasta que el gestor cambie el `administrativeState` a desbloqueado.

El apéndice III representa los diagramas de transición de estados en los cuales los controles manuales y automáticos están activos y no activos.

7.1.3.1 Control automático de congestión (ACC)

El control automático de congestión es un control preasignado que es automáticamente activado cuando se recibe una señal de disparo de congestión de la central adyacente. Se define en 4.1/E.412. En las centrales de recepción, todos los subhaces de circuitos que terminan en la central congestionada deben ser sometidos a controles de tráfico que limitarán el volumen de tráfico enviado a la central congestionada. La disposición de las llamadas que puede utilizarse para frenar el tráfico suelen ser la cancelación o el salto de ruta.

El control se basa en un temporizador automático fijado internamente por el NE. Cuando se recibe el indicador de congestión, se arranca un temporizador en la central recibiente. Posteriormente, los indicadores de congestión recibidos rearrancan el temporizador. Si no se recibe ningún indicador de congestión posterior dentro del temporizador, el control es automáticamente desactivado. La selección del valor del temporizador es específica del vendedor y no es gestionable por un OS. Los aspectos de temporizador del ACC se describen en 5.5.2 b)/Q.542.

Si se desea una `dispositionOfCalls` (disposición de las llamadas) diferente para cada nivel de congestión, debería instanciarse un objeto de control `acc`. Además, las instancias de `acc` que tienen el mismo nivel de congestión y `controlTmCircuitEndPointSubgroup` deben utilizar la misma `dispositionOfCall`.

Una instancia de `accAffectedTraffic` (tráfico afectado por el `acc`) apuntada por `assocAccAffectedTraffic` debe existir antes de que se instancie una clase de objeto `acc`.

Éste es un control de protección. Se trata de una subclase de la clase de objeto `trafficControl`. Véase el cuadro 17.

Cuadro 17/Q.823 – Entidad `acc`

Atributos	M/S	O/M/C
<code>controlTmCircuitEndPointSubgroup</code>	S	M
<code>dispositionOfCalls</code>	S	M
<code>administrativeState</code>	S	M
<code>autoActivated</code>	S	M
<code>assocAccAffectedTraffic</code>	S	M

Los siguientes atributos describen la entidad `acc`:

- `controlTmCircuitEndPointSubgroup` (subhaz de punto extremo de circuito Tm de control) – Identifica el subhaz de circuito al que se aplica este control.
- `dispositionOfCalls` (disposición de llamadas) – Identifica si las llamadas se pasarán al nuevo subhaz de circuitos disponible (si el valor es NULO) o serán canceladas (si el valor es tratamiento).

- **administrativeState** (estado administrativo) – Identifica si el control ha sido bloqueado por el gestor de red. Cuando el valor es "bloqueado", el control no será activado. Si el valor es "desbloqueado", el control será activado. Cuando se crea el control, es creado con el valor de desbloqueado. Los únicos valores admitidos del estado administrativo son "bloqueado" y "desbloqueado".
- **autoActivated** (activado automáticamente) – Indica si está pendiente un disparo de este control. Cuando este atributo es 'VERDADERO' y **administrativeState** está desbloqueado, el control automático está activo. El control automático se desactiva para todas las demás combinaciones de los valores de los atributos **autoActivated** y **administrativeState**. Un cambio de valor de este atributo no generará una notificación automática de cambio de valor de atributo.
- **assocAccAffectedTraffic** (tráfico afectado por control automático de congestión asociado) – Apunta al ejemplar de objeto **accAffectedTraffic** para el nivel de congestión 1 y 2. Debe ser no nulo.

7.1.3.2 Tráfico afectado por el control automático de congestión ACC (**accAffectedTraffic**)

El objeto **accAffectedTraffic** identifica el rigor del control para diversos niveles de congestión para una combinación de aspecto de encaminamiento, aspecto de destino, aspecto de origen y/o criterios adicionales, especificados.

El aspecto de origen, aspecto de destino, aspecto de encaminamiento y criterios adicionales son todos ellos claves en ese orden.

Si se suprime una instancia de **accAffectedTraffic**, todas las instancias de objeto **acc** referentes a esta instancia deben ser suprimidas. Véase el cuadro 18.

Cuadro 18/Q.823 – Entidad **accAffectedTraffic**

Atributos	M/S	O/M/C
accAffectedTrafficId	S	M
cl1ResponseCategories	S	M
cl2ResponseCategories	S	M

Los siguientes atributos describen la entidad **accAffectedTraffic**:

- **accAffectedTrafficId** (Id de tráfico afectado por **acc**) – Este atributo se utiliza como el RDN.
- **cl1ResponseCategories** (categorías de respuesta **cl1**) – Identifica en una secuencia el rigor porcentual de una combinación de aspecto de encaminamiento, aspecto de origen, aspecto de destino y/u otros criterios para nivel 1 de congestión. El rigor del control es aplicado a la combinación de uno o más de estos criterios. Se cree que sólo se soportarán una de las opciones porcentuales.
- **cl2ResponseCategories** (categorías de respuesta **cl2**) – Identifica en una secuencia el rigor porcentual de una combinación de aspecto encaminamiento, aspecto de origen, aspecto de destino y/u otros criterios para el nivel 2 de congestión. El rigor de control se aplica a la combinación de uno o más de estos criterios. Se cree que sólo se soportará una de las opciones porcentuales.

7.1.3.3 Disparador de ACC (accTrigger)

El disparador de ACC reside en la central congestionada. Cuando está bloqueado, emitirá indicaciones de congestión a las centrales adyacentes para que ellas activen automáticamente el ACC preasignado de manera que se frene el tráfico hacia la central congestionada. Cuando está bloqueado, la central congestionada no emitirá la indicación de congestión a las centrales adyacentes.

Obsérvese que congestionLevel se modela en la clase de objeto congestionLevelIndication. Véase el cuadro 19.

Cuadro 19/Q.823 – Entidad accTrigger

Atributos	M/S	O/M/C
accTriggerId	S	M
administrativeState	S	M

Los siguientes atributos describen la entidad accTrigger:

- accTriggerId (Id de disparador de acc) – Este atributo se utiliza como el RDN.
- estado administrativo (administrativeState) – Identifica si la indicación de congestión será emitida desde la central congestionada hasta las centrales adyacentes. Si el valor es "desbloqueado", se emitirá la indicación; si el valor es "bloqueado", no se emitirá la indicación de congestión. Sólo los valores "bloqueado" y "desbloqueado" son válidos para este atributo.

7.1.3.4 Control automático de destino (ADC)

ADC es de uno de dos tipos: centralizado y descentralizado.

En el ADC centralizado, la detección de la congestión hacia un indicativo de destino se realiza en el nodo de destino cuando la tasa de llegadas supera el umbral fijado para ese indicativo de destino. Se describe en 4.3/E.412. El disparador adc identifica cuándo se rebasa el umbral, lo que da lugar al envío de una indicación a cada nodo de origen mediante un mensaje de señalización que indica que ha de reducirse el tráfico ofrecido a ese destino. El mensaje de señalización especifica los detalles del control así como la hora de expiración del mismo. Al recibir el mensaje, el nodo recibiente instanciará el objeto ADC.

En el ADC descentralizado, se detecta la congestión de destino en el NE fuente observando los intentos de toma y las respuestas para un indicativo de destino seleccionado. Se describe en 4.3/E.412. El objeto disparador de ADC (adcTrigger) identifica el destino a supervisar, el valor umbral y el rigor del control. Cuando se cruza el valor umbral en el NE fuente, instanciará automáticamente el objeto ADC. Si se bloquea el control ADC, no se pondrá activo ni siquiera si el disparador está activo. El control ADC estará activo sólo cuando el administrativeState se ponga al estado desbloqueado.

El tipo de ADC se identifica por el atributo adcType que puede tomar el valor de "centralizado" o "descentralizado".

Este objeto de control es instanciado sólo por el NE y no por el OS cuando recibe un mensaje del nodo de destino (en el caso de centralizado) o cuando se cruza el valor umbral en el NE fuente (en el caso de descentralizado).

El tipo de destino y el indicativo de destino son claves para el control en ese orden.

Exactamente uno de los atributos de rigor (adcPercentage o adcContinuousTimer o adcAsynchronousTimer o adcLeakyBucket), debe estar presente cuando se instancia el objeto de control. La instancia de objeto adc se suprime al final de la expiración del tiempo. Si se recibe una señal de disparo posterior antes de la expiración de tiempo, la instancia adc es sustituida por los nuevos valores indicados en el último disparo.

Este objeto es una subclase de la clase de objeto trafficControl. Véase el cuadro 20.

Cuadro 20/Q.823 – Entidad adc

Atributos	M/S	O/M/C
adcType	S	M
destinationType	S	C
destinationCode	S	M
adcPercentage	S	C
adcContinuousTimer	S	C
adcAsynchronousTimer	S	C
adcLeakyBucket	S	C
treatment	S	M
timeExpiration	S	M
administrativeState	S	M
autoActivated	S	M

Los siguientes atributos describen la entidad adc:

- adcType (tipo de adc) – Identifica el tipo de ADC. Puede tomar el valor de "centralizado" o "descentralizado".
- destinationType (tipo de destino) – Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Identifica el indicativo de destino del destino congestionado.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultas de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).
- continuousTimer (temporizador continuo) – Identifica el temporizador continuo que incluye el número de llamadas y un tiempo. Una vez alcanzado el número de intentos de llamada tratados en un ciclo de temporizador, no se permiten nuevos intentos hasta que expire el temporizador (por ejemplo, 5 llamadas en 60 segundos).
- asynchronousTimer (temporizador asíncrono) – Identifica el temporizador asíncrono que especifica un tiempo. El temporizador se fija cuando se permite el intento y no se permiten nuevos intentos de llamada hasta que expire el temporizador.
- leakyBucket (contador dinámico) – Identifica el contador dinámico que incluye el tamaño del contador (máximo valor permitido del contador) y el decremento por unidad de tiempo. Si el contador supera el máximo tamaño definido, se cancela el intento de llamada. Si el tamaño del contador es menor que el máximo, se permite el intento de llamada y se

incrementa el contador. El contador se decrementa a intervalos definidos, haciendo posible que se acepten nuevas llamadas.

- `treatment` (tratamiento) – Indica el tratamiento de la llamada cuando ésta se cancela de resultados del control (por ejemplo, un anuncio). El tratamiento es determinado por el NE cuando se instancia el objeto de control.
- `timeExpiration` (expiración del tiempo) – Identifica el intervalo de tiempo tras el cual se suprimirá automáticamente el control. Actúa como una histéresis para evitar la oscilación frecuente entre la instanciación y la desinstanciación del control.
- `administrativeState` (estado administrativo) – Identifica si el control ha sido inhibido por el gestor de red. Cuando el valor es "bloqueado", el control no será activado. Si el valor es "desbloqueado", el control será activado. El valor bloqueo y desbloqueo sólo son válidos en esta definición de clase de objeto.
- `autoActivated` (activado automáticamente) – Identifica si está pendiente el disparo de este control. Como el control es instanciado sólo cuando el disparo está pendiente, el valor de `autoActivated` es siempre VERDADERO. El valor FALSO no es válido para este atributo.

7.1.3.5 Disparador de control automático de destino (ADC) (`adcTrigger`)

El objeto disparador de ADC es establecido por el OS con los atributos necesarios para que el objeto ADC pueda ser automáticamente instanciado (en el mismo NE en el que reside el `adcTrigger` o en un NE adyacente) cuando se cruza el valor umbral. OS puede especificar el destino que ha de ser supervisado por el NE para ver si hay congestión, el valor umbral de disparo, el rigor del control y la expiración del tiempo.

El `adcTrigger` puede ser del tipo "centralizado" o "descentralizado", o "ambos", como se identifica en el atributo `adcType`.

Si el `adcType` es "centralizado", el `adcTrigger` y el control `adc` están en dos NE diferentes. Se envía un mensaje de señalización desde la NE fuente (donde reside el objeto `adcTrigger`) a todos los NE adyacentes para crear el control `adc`.

Si el `adcType` está "descentralizado", el `adcTrigger` y el control `adc` están en el mismo NE. El NE fuente contiene el `adcTrigger` y también el control `adc`.

Si el `adcType` es "ambos", el NE fuente (donde reside el `adcTrigger`) envía un mensaje de señalización a todos los NE adyacentes (es decir `adc` centralizado) para crear un control `adc`. También instanciará el control `adc` dentro de él mismo (`adc` descentralizado).

Un OS puede instanciar el objeto disparador de ADC, pero no el objeto de control ADC. El disparador ADC puede también ponerse al estado bloqueado. Cuando está en el estado bloqueado, no enviará automáticamente un mensaje a la central adyacente ni instanciará el control ADC ni siquiera si se cruza el umbral de disparo.

Exactamente un atributo de rigor (porcentaje o `continuousTimer` o `asynchronousTimer` o `leakyBucket`) debe estar presente cuando se instancia este objeto.

El tipo de destino y el indicativo de destino son claves para el control en ese orden. Véase el cuadro 21.

Cuadro 21/Q.823 – Entidad adcTrigger

Atributos	M/S	O/M/C
adcTriggerId	S	M
adcTriggerType	S	M
destinationType	S	C
destinationCode	S	C
triggerThreshold	S	C
percentage	S	C
continuousTimer	S	C
asynchronousTimer	S	C
leakyBucket	S	C
administrativeState	S	M
timeExpiration	S	M

Los siguientes atributos definen la entidad adcTrigger:

- adcTriggerId (Id de disparador adc) – Este atributo se utiliza como el RDN.
- adcTriggerType (tipo de disparador adc) – El tipo de adcTrigger.
- destinationType (tipo de destino) – Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Identifica el destino a supervisar para ver si hay congestión.
- triggerThreshold (umbral de disparo) – Identifica al valor umbral de disparo. Si se cruza, se instanciará el control adc.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultados de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).
- continuousTimer (temporizador continuo) – Identifica el temporizador continuo que incluye el número de llamadas y un tiempo. Una vez alcanzado el número de intentos de llamada tratados en un ciclo de temporizador, no se permiten nuevos intentos hasta que expire el temporizador (por ejemplo, 5 llamadas en 60 segundos).
- asynchronousTimer (temporizador asíncrono) – Identifica el temporizador asíncrono que especifica un tiempo. El temporizador se fija cuando se permite el intento y no se permiten nuevos intentos de llamada hasta que expire el temporizador.
- leakyBucket (contador dinámico) – Identifica el contador dinámico que incluye el tamaño del contador (máximo valor permitido del contador) y el decremento por unidad de tiempo. Si el contador supera el máximo tamaño definido, se cancela el intento de llamada. Si el tamaño del contador es menor que el máximo, se permite el intento de llamada y se incrementa el contador. El contador se decrementa a intervalos definidos, haciendo posible que se acepten nuevas llamadas.
- administrativeState (estado administrativo) – Identifica si el disparador ha sido bloqueado por el gestor de red. Cuando el valor es "bloqueado", no se instanciará el control adc. Si el valor es "desbloqueado" se instanciará el control adc.

- timeExpiration (expiración del tiempo) – Identifica el intervalo de tiempo tras el cual se suprimirá automáticamente el control.

7.1.3.6 Cancelación desde (cancelFrom)

Este control es manualmente activado en un subhaz de circuitos salientes y prohíbe que el tráfico desborde a los siguientes subhaces de circuitos de la cadena. Se define en 3.2.1/E.412.

Este es un control de protección de subhaces de circuitos. Se trata de una subclase de la clase de objeto trafficControl.

El control cancelación desde puede utilizarse para prohibir el desbordamiento del tráfico con encaminamiento directo y alternativo. El apéndice IV ofrece una explicación de la forma de prohibir el desbordamiento del tráfico con encaminamiento directo y alternativo utilizando el control cancelFrom. La distinción entre estos dos tipos de tráfico se hace mediante la asignación de tipos de tráfico apropiados al atributo routingAspect.

El atributo controlTmCircuitEndPointSubgroup es la clave con la máxima prioridad, seguido por cada componente del originationAspect, destinationAspect y routingAspect, en ese orden. El valor más específico tiene la prioridad. Véase el cuadro 22.

Cuadro 22/Q.823 – Entidad cancelFrom

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
routingAspect	S	M
destinationAspect	S	M
originationAspect	S	M
percentage	S	M
treatment	S	M

Los siguientes atributos describen la entidad cancelFrom:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica esta instancia de objeto de control.
- routingAspect (aspecto de encaminamiento) – Identifica el aspecto de encaminamiento (tráfico con encaminamiento directo o tráfico con encaminamiento alternativo) para el que es válido este control. Si el valor de atributo es NULO, el control de tráfico se aplica a todos los aspectos de encaminamiento.
- destinationAspect (aspecto de destino) – Identifica el aspecto de destino (por ejemplo, HTR) para el cual es válido este control. Si este atributo es NULO, el control de tráfico es válido para todos los aspectos de destino.
- originationAspect (aspecto de origen) – Identifica el origen y la categoría de la parte llamante (de acuerdo con la Recomendación Q.763) para los que es válido este atributo. Si este atributo es la secuencia vacía, el control de tráfico es válido para todos los aspectos de origen.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultados de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).

- treatment (tratamiento) – Indica el tratamiento de la llamada cuando ésta se cancela de resultas del control (por ejemplo, un anuncio).

7.1.3.7 Cancelación de desbordamiento reencaminado (cancelRerouted)

Este control impide el encaminamiento adicional o el encaminamiento alternativo de llamadas que ya han sido reencaminadas. No se permite que las llamadas reencaminadas desborden al subhaz de circuitos al cual se aplica el control de cancelación de desbordamiento reencaminado, mientras el tráfico de desbordamiento normal no sea afectado. Se define en 3.2.4/E.412.

Es de señalar que a fin de proporcionar este control, cada llamada ha de ser marcada mediante un mensaje de señalización siempre que se aplique a la misma un encaminamiento alternativo temporal (TAR, *temporary alternative routing*).

Este es un control manual de protección. Se trata de una subclase de la clase de objeto trafficControl. Véase el cuadro 23.

Cuadro 23/Q.823 – Entidad cancelRerouted

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
treatment	S	M

Los siguientes atributos describen la entidad cancelRerouted:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica esta instancia de objeto de control.
- treatment (tratamiento) – Identifica cómo se tratará el flujo de tráfico afectado por la cancelación (por ejemplo, un anuncio).

7.1.3.8 Cancelación hacia (cancelTo)

Este control comprende la cancelación del encaminamiento directo y la cancelación del encaminamiento alternativo a un determinado controlTmCircuitEndPointSubgroup.

La cancelación del encaminamiento directo bloquea el volumen de tráfico con encaminamiento directo que accede a un subhaz de circuitos. Se define en 3.1.2/E.412.

La cancelación del encaminamiento alternativo al control es activada en un subhaz de circuitos salientes y prohíbe al tráfico de desbordamiento acceder al subhaz de circuitos controlado. Se define en 3.2.1/E.412.

La distinción entre estos dos tipos se hace mediante la asignación del atributo routingAspect apropiado.

El apéndice IV explica el modo de prohibir que el tráfico con encaminamiento directo o alternativo acceda a un subhaz de circuitos utilizando el control cancelación hacia.

El atributo controlTmCircuitEndPointSubgroup es la clave con la máxima prioridad, seguida por cada componente del originationAspect, destinationAspect y routingAspect, en ese orden. El valor más específico tiene la prioridad.

Éste es un control manual de protección. Se trata de una subclase de la clase de objeto trafficControl. Véase el cuadro 24.

Cuadro 24/Q.823 – Entidad cancelTo

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
routingAspect	S	M
destinationAspect	S	M
originationAspect	S	M
percentage	S	M
treatment	S	M

Los siguientes atributos describen la entidad cancelTo:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica esta instancia de objeto de control.
- routingAspect (aspecto de encaminamiento) – Identifica el aspecto de encaminamiento (tráfico con encaminamiento directo o tráfico con encaminamiento alternativo) para el que es válido este control. Si el valor de atributo es NULO, el control de tráfico se aplica a todos los aspectos de encaminamiento.
- destinationAspect (aspecto de destino) – Identifica el aspecto de destino (por ejemplo, HTR) para el cual es válido este control. Si este atributo es NULO, el control de tráfico es válido para todos los aspectos de destino.
- originationAspect (aspecto de origen) – Identifica el origen y la categoría de la parte llamante (de acuerdo con la Recomendación Q.763) para los que es válido este atributo. Si este atributo es la secuencia vacía, el control de tráfico es válido para todos los aspectos de origen.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultados de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).
- treatment (tratamiento) – Indica el tratamiento de la llamada cuando ésta se cancela de resultados del control (por ejemplo, un anuncio).

7.1.3.9 Control de indicativo de destino (destinationCodeControl)

Este control manual prohíbe el encaminamiento a un destino concreto en porcentaje (denominado en la Recomendación E.412 bloqueo de código) o en tasa de llamadas (denominado en la Recomendación E.412 espaciado de llamadas).

El control de indicativo de destino puede aplicarse a un indicativo de país, o/y un indicativo de área, o/y un indicativo de central u/y otro número de lugar. Se define en 3.1.1.1/E.412.

Cuando este control está activo en el modo bloqueo de código, se bloquea (cancela) el porcentaje de llamadas especificado.

Cuando el código está activo en el modo con espaciado de llamadas, se controlará la frecuencia a la que se entregan las llamadas al destino. Puede especificarse en uno de estos tres métodos: temporizador continuo, temporizador asíncrono o contador dinámico.

El rigor (bloqueo de código o espaciado de llamadas) del control de indicativo de destino puede estar dentro de la misma clase de objeto o puede definirse en una clase de objeto separada. El primer caso es aplicable cuando el rigor del control se aplica únicamente a un destino. En este caso, existen

cuatro posibilidades (percentage, continuousTimer, asynchronousTimer o leakyBucket) para especificar el rigor de la misma clase de objeto, de las cuales debe especificarse exactamente una. En estas circunstancias, la instancia de objeto única actúa como una función de control de indicativo.

El último caso citado de definir el rigor en una clase de objeto separada es aplicable únicamente cuando se desea aplicar el mismo rigor a un grupo de destino (el contador dinámico es un ejemplo en el que puede ser útil dicho comportamiento). En este caso, se requiere el atributo assocOwnerDccGroup, que apunta a la clase de objeto que contiene el rigor del control. En estas circunstancias, dos clases de objeto juntas representan una función de control de indicativo.

Las siguientes reglas son válidas independientemente de si se utilizan una o dos clases de objeto para representar una función de control de indicativo de destino:

- No es posible crear dos o más instancias de objeto destinationCodeControl con una combinación de valores idéntica de los atributos claves destinationType, destinationCode y originationAspect.
- El atributo destinationCode es la clave con la máxima prioridad, seguido por el destinationType (si está presente) y los componentes del atributo originationAspect, en ese orden.
- Cuando existe más de un control con dígitos superpuestos, se aplica el control más específico del mismo tipo en lugar del control menos específico, los cuales podrían ambos afectar a la misma llamada.

Si se especifica el atributo assocOwnerDccGroup, debe existir el dccGroup asociado. Las instancias de esta clase de objeto pertenecen al mismo grupo sólo si tienen el mismo valor del assocOwnerDccGroup.

Este control es una subclase de la clase de objeto trafficControl. Véase el cuadro 25.

Cuadro 25/Q.823 – Entidad destinationCodeControl

Atributos	M/S	O/M/C
destinationType	S	C
destinationCode	S	M
originationAspect	S	M
percentage	S	C
continuousTimer	S	C
asynchronousTimer	S	C
leakyBucket	S	C
assocOwnerDccGroup	S	C
treatment	S	M

Los siguientes atributos describen la entidad destinationCodeControl:

- destinationType (tipo de destino) – Identifica el tipo de destino. Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Es un indicativo de país o/y un indicativo de área, o/y una central, u/y otro número de lugar.

- `originationAspect` (aspecto de origen) – Identifica el origen y la categoría de la parte llamante (de acuerdo con la Recomendación G.763) para los que es válido este control. Si este atributo es la secuencia vacía, el control es válido para todos los aspectos de origen.
- `percentage` (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultados de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado). Este atributo se utiliza solamente si el rigor del control está dentro de la misma clase de objeto que el propio control.
- `continuousTimer` (temporizador continuo) – Identifica el temporizador continuo que incluye el número de llamadas y un tiempo. Una vez alcanzado el número de intentos de llamada tratados en un ciclo de temporizador, no se permiten nuevos intentos hasta que expire el temporizador (por ejemplo, 5 llamadas en 60 segundos). Este atributo se utiliza solamente si el rigor del control está dentro de la misma clase que el propio control.
- `asynchronousTimer` (temporizador asíncrono) – Identifica el temporizador asíncrono que especifica un tiempo. Este temporizador se fija cuando se permite el intento de llamada y no se permiten nuevos intentos de llamadas hasta que expire el temporizador. Este atributo se utiliza solamente si el rigor del control está dentro de la misma clase de objeto que el propio control.
- `leakyBucket` (contador dinámico) – Identifica el contador dinámico que incluye el tamaño del contador (máximo valor permitido del contador) y el decremento por unidad de tiempo. Si el contador supera el máximo tamaño definido, se cancela el intento de llamada. Si el tamaño del contador es menor que el máximo, se permite el intento de llamada y se incrementa el contador. El contador se decrementa a intervalos definidos haciendo posible que se acepten nuevas llamadas. Este atributo se utiliza solamente si el rigor del control está dentro de la misma clase de objeto que el propio control.
- `assocOwnerDccGroup` (grupo de controles asociados de indicativo de destino de un propietario) – Este atributo está presente si ha de aplicarse el mismo rigor a un grupo de destinos; es decir, esta instancia `destinationCodeControl` es un elemento de un grupo de controles de destino a los que ha de aplicarse el mismo rigor. Apunta a la instancia de objeto `dccGroup` en la que se define el rigor del control.
- `treatment` (tratamiento) – Indica el tratamiento de la llamada cuando ésta se cancela de resultados del control (por ejemplo, un anuncio).

7.1.3.10 Grupo de control de indicativo de destino (`dccGroup`)

Esta clase de objeto define el rigor del control de indicativo de destino cuando se debe aplicar el mismo rigor a un grupo de destinos. Las instancias de esta clase de objeto están asociadas con instancias de la clase de objeto `destinationCodeControl` mediante una relación (1:n). Esta clase de objeto no proporciona en sí la función de control de indicativo de destino; cuando está asociada con la(s) instancia(s) `destinationCodeControl` representa la función de control de indicativo de destino.

Exactamente uno de los atributos de rigor (`percentage`, `continuousTimer`, `asynchronousTimer` o `leakyBucket`) debe estar presente cuando se instancia el objeto de control. Véase el cuadro 26.

Cuadro 26/Q.823 – Entidad dccGroup

Atributos	M/S	O/M/C
dccGroupId	S	M
percentage	S	C
continuousTimer	S	C
asynchronousTimer	S	C
leakyBucket	S	C

Los siguientes atributos describen la entidad dccgroup:

- dccGroupId (Id de grupo dcc) – Este atributo es el identificador de objeto. Se utiliza como el RDN.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que serán canceladas de resultas de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).
- continuousTimer (temporizador continuo) – Identifica el temporizador continuo que incluye el número de llamadas y un tiempo. Una vez alcanzado el número de intentos de llamada tratados en un ciclo de temporizador, no se permiten nuevos intentos hasta que expire el temporizador (por ejemplo, 5 llamadas en 60 segundos).
- asynchronousTimer (temporizador asíncrono) – Identifica el temporizador asíncrono que especifica un tiempo.
- leakyBucket (contador dinámico) – Identifica el contador dinámico que incluye el tamaño del contador (máximo valor permitido del contador) y el decremento por unidad de tiempo. Si el contador supera el máximo tamaño definido, se cancela el intento de llamada. Si el tamaño del contador es menor que el máximo, se permite el intento de llamada y se incrementa el contador. El contador se decrementa a intervalos definidos haciendo posible que se acepten nuevas llamadas.

7.1.3.11 Control de reserva selectiva de circuitos (scr, selective circuit reservation control)

El control de reserva selectiva de circuitos permite a una central dar preferencia automáticamente a determinados atributos de tráfico sobre otros (por ejemplo, a las llamadas con encaminamiento directo sobre las llamadas con encaminamiento alternativo) en el momento en que la congestión de circuitos está presente o es inminente. Puede disponerse con uno o dos umbrales, con posterior provisión de una mayor selectividad. El control es preasignado con umbrales que indican cuántos circuitos deben mantenerse en reposo durante los diversos tipos de tráfico en un subhaz de circuitos. Cuando el número de circuitos en reposo o la capacidad en reposo en el subhaz de circuitos es menor o igual que el umbral de reserva, la llamada es cancelada o salta al siguiente subhaz de circuitos de la cadena de rutas. Este control se define en 4.2/E.412.

El control SCR tiene las siguientes variables operativas:

- umbrales de reserva
- respuesta de control
- opción de acción de control

Los umbrales de reserva y la correspondiente respuesta de control son determinados por el atributo activationThresholds y por la instancia de objeto scrAffectedTraffic asociada. La opción de acción de control para el procesamiento de llamadas a las que se ha negado el acceso al subhaz de circuitos se indican en el atributo dispositionOfCalls.

Cuando el número de circuitos o la capacidad en reposo en el subhaz de circuitos es menor o igual que el umbral de reserva, la central comprobará la instancia de objeto `scrAffectedTraffic` indicada para determinar si se controlará la llamada. Las disposiciones que pueden utilizarse para frenar el tráfico son cancelación o salto de ruta.

Los niveles definidos en el atributo `activationThresholds` y la referencia indicada en los correspondientes atributos de tráfico afectados por la `scr` corresponderán a un control uniumbral o multiumbral.

La instancia de objeto `scrAffectedTraffic` debe estar presente antes de que pueda instanciarse la clase de objeto `scr`.

Éste es un control automático de protección. Se trata de una subclase de la clase de objeto `trafficControl`. Véase el cuadro 27.

Cuadro 27/Q.823 – Entidad `scr`

Atributos	M/S	O/M/C
<code>controlTmCircuitEndPointSubgroup</code>	S	M
<code>dispositionOfCalls</code>	S	M
<code>administrativeState</code>	S	M
<code>autoActivated</code>	S	M
<code>activationThresholds</code>	S	M
<code>assocScrAffectedTraffic</code>	S	M

Los siguientes atributos definen la entidad `scr`:

- `controlTmCircuitEndPointSubgroup` (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al que se aplica este control.
- `dispositionOfCalls` (disposición de las llamadas) – Identifica si las llamadas han de saltar al siguiente subhaz de circuitos disponible (si el valor es NULO) o serán canceladas (si el valor es tratamiento).
- `administrativeState` (estado administrativo) – Identifica si el control ha sido bloqueado por el gestor de red. Cuando el valor es "bloqueado", el control no será activado. Si el valor es "desbloqueado" el control será activado.
- `autoActivated` (con activación automática) – Indica si está pendiente un disparo de este control. Cuando este atributo es "VERDADERO" y `administrativeState` está 'desbloqueado', el control automático está activo. El control automático se desactiva para todas las demás combinaciones de los valores de los atributos `autoActivated` y `administrativeState`. Un cambio de valor de este atributo no generará una notificación automática de cambio de valor de atributo.
- `activationThresholds` (umbrales de activación) – Identifica los umbrales de activación para disparar el control.
- `assocScrAffectedTraffic` (tráfico afectado por la reserva selectiva de circuitos asociada) – Apunta al ejemplar de objeto del `scrAffectedTraffic` para el nivel de reserva 1 y 2. Debe ser no nulo.

7.1.3.12 Tráfico afectado por la reserva selectiva de circuitos (scrAffectedTraffic)

La clase de objeto tráfico afectado por la reserva selectiva de circuitos representa la categoría de respuesta de control para el control SCR.

Determina para cada aspecto de encaminamiento, aspecto de destino, aspecto de origen y/o criterios adicionales, la intensidad del tráfico a controlar.

Si se especifica level2ResponseCategories, deben especificarse los umbrales de activación nivel 1 y nivel 2 utilizando la misma unidad (número o porcentaje).

El aspecto de origen, el aspecto de destino, el aspecto de encaminamiento y los criterios adicionales son claves en ese orden. Véase el cuadro 28.

Cuadro 28/Q.823 – Entidad scrAffectedTraffic

Atributos	M/S	O/M/C
scrAffectedTrafficId	S	M
level1ResponseCategories	S	M
level2ResponseCategories	S	C

Los siguientes atributos describen la entidad scrAffectedTraffic:

- scrAffectedTrafficId (Id de tráfico afectado por la scr) – Este atributo se utiliza como el RDN.
- level1ResponseCategories (categorías de respuesta de nivel 1) – Identifica en una secuencia el rigor para una combinación de aspecto de encaminamiento y/o destinationAspect, originationAspect y cualquier otro criterio adicional para el nivel 1 de reserva.
- level2ResponseCategories (categorías de respuesta de nivel 2) – Identifica en una secuencia el rigor para una combinación de aspecto de encaminamiento y/o destinationAspect, OriginationAspect y cualquier otro criterio adicional para el nivel 2 de reserva.

7.1.3.13 Salto de ruta (skip)

Este control es activado en un haz de circuitos saliente si se utiliza para forzar el tráfico al siguiente subhaz de circuitos de la cadena en la tabla de encaminamientos. El control del salto de ruta puede afectar al tráfico de encaminamiento directo y al alternativo. Se define en 3.2.2/E.412.

El atributo controlTmCircuitEndPointSubgroup es la clave con la máxima prioridad, seguido por cada componente del originationAspect, destinationAspect y routingAspect, en ese orden. El valor más específico tiene la prioridad.

Éste es un control manual de protección. Se trata de una subclase de la clase de objeto trafficControl. Véase el cuadro 29.

Cuadro 29/Q.823 – Entidad skip

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
routingAspect	S	M
destinationAspect	S	M
originationAspect	S	M
percentage	S	M

Los siguientes atributos describen la entidad skip:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica esta instancia de objeto de control.
- routingAspect (aspecto de encaminamiento) – Identifica el aspecto de encaminamiento (tráfico con encaminamiento directo o tráfico con encaminamiento alternativo) para el que es válido este control. Si el valor de atributo es NULO, el control de tráfico se aplica a todos los aspectos de encaminamiento.
- destinationAspect (aspecto de destino) – Identifica el aspecto de destino (por ejemplo, HTR) para el cual es válido este control. Si este atributo es NULO, el control de tráfico es válido para todos los aspectos de destino.
- originationAspect (aspecto de origen) – Identifica el origen y la categoría de la parte llamante (de acuerdo con la Recomendación Q.763) para los que es válido este atributo. Si este atributo es NULO, el control de tráfico es válido para todos los aspectos de origen.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que saltarán de ruta de resultados de la activación del control. Cuando este control está activo, el porcentaje especificado de llamadas es bloqueado (cancelado).

7.1.3.14 Encaminamiento alternativo temporal desde (tarFrom)

El encaminamiento alternativo temporal (TAR) desde es un control expansivo que aumenta temporalmente el número de posibilidades de encaminamiento para completar llamadas mediante:

- adición de nuevos subhaces de circuitos al final de la tabla de encaminamientos, o
- inserción de nuevos subhaces de circuitos en la tabla de encaminamientos entre subhaces de circuitos existentes, para proporcionar trayectos adicionales de desbordamiento que no estén normalmente disponibles en el plan de encaminamiento normal.

Estos dos tipos de TAR (adición al final o inserción entre haces de circuitos) se describen en 3.2.3/E.412. tarFrom sólo se aplica a nivel de subhaces de circuitos y repercute en todas las tablas de encaminamiento en las que aparece controlTmCircuitEndPointSubgroup. Los haces de circuitos de encaminamiento alternativo temporal deben terminar en una central que tenga la capacidad de alcanzar el destino final.

El efecto de este control puede limitarse a destinos que estén en la lista HTR (destinos difíciles de alcanzar) fijando el atributo destinationAspect, o especificando explícitamente el destino en el atributo destinationCode. Si el indicativo de destino es una cadena vacía, el control se aplica a los destinos servidos por ese subhaz de circuitos.

El efecto de este control puede también limitarse al tráfico con encaminamiento directo o alternativo.

Los controlTmCircuitEndPointSubgroup, destinationCode, destinationType (si está presente), los componentes de originationAspect, los componentes de destinationAspect y los componentes de routingAspect son todos claves para el control, en ese orden.

Éste es un control manual expansivo. Se trata de una subclase de la clase de objeto trafficControl. Véase el cuadro 30.

Cuadro 30/Q.823 – Entidad tarFrom

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
newTmCircuitEndPointSubgroups	S	M
routingAspect	S	M
destinationAspect	S	M
originationAspect	S	M
percentage	S	M
destinationType	S	C
destinationCode	S	M
returnAction	S	M

Los siguientes atributos describen la entidad en tarFrom:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica el control tarFrom.
- newTmCircuitEndPointSubgroups (nuevos subhaces de punto extremo de circuitos de gestión de tráfico) – Secuencia de nuevos haces de circuitos. Debe estar presente al menos un elemento en la secuencia.
- routingAspect (aspecto de encaminamiento) – Identifica el aspecto de encaminamiento (tráfico con encaminamiento directo o tráfico con encaminamiento alternativo) para el que es válido este control. Si el valor de atributo es NULO, el control de tráfico se aplica a todos los aspectos de encaminamiento.
- destinationAspect (aspecto de destino) – Identifica el aspecto de destino (por ejemplo, HTR) para el cual es válido este control. Si este atributo tiene el valor NULO, el control de tráfico es válido para todos los aspectos de destino.
- originationAspect (aspecto de origen) – Identifica el origen y la categoría de parte llamante (de acuerdo con la Recomendación Q.763) para los que es válido este control. Si este atributo es la secuencia vacía, el control de tráfico es válido para todos los aspectos de origen.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que se reencaminarán de resultados de la activación del control.
- destinationType (tipo de destino) – Identifica el tipo de destino. Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Es un indicativo de país, o/y un indicativo de área, o/y una central, u/y otro número de lugar para el cual se aplica este TAR. Si el valor de atributo es una cadena vacía, el control es válido para todos los destinos en ese controlTmCircuitEndPointSubgroup.
- returnAction (acción de retorno) – Identifica la disposición del tráfico de desbordamiento desde los newTmCircuitEndPointSubgroups. Existen dos posibilidades:
 - poniendo returnAction a un valor NULO se insertan newTmCircuitEndPointSubgroups después de TmCircuitEndPointSubgroup;
 - poniendo returnAction en tratamiento se cancelará el tráfico que se ha desbordado del newTmCircuitEndPointSubgroup.

7.1.3.15 Encaminamiento alternativo temporal hacia (tarTo)

El encaminamiento alternativo temporal (TAR) hacia es un control expansivo que aumenta temporalmente el número de posibilidades de encaminamiento para completar llamadas añadiendo nuevos subhaces de circuitos al comienzo de la tabla de encaminamientos de manera que el tráfico se ofrezca primero al nuevo subhaz de circuitos, o sustituyendo el subhaz de circuitos existente por un conjunto de nuevos subhaces de circuitos. Se ponen disponibles uno o varios haces de circuitos, que no están normalmente disponibles en el plan de encaminamiento normal, pero que tienen capacidad en reposo.

Estos dos tipos de TAR (añadir al comienzo o sustituir un subhaz de circuitos existente) se describen en la 3.2.3/E.412. tarTo se aplica solamente al nivel de subhaz de circuitos y repercute en todas las tablas de encaminamientos en las que está presente tmCircuitEndPointSubgroup. Los haces de circuitos de encaminamiento alternativos temporales deben terminar en una central que tenga la capacidad de alcanzar el destino final.

El efecto de este control puede limitarse a destinos que están en la lista HTR fijando el atributo aspecto de encaminamiento, o especificando explícitamente el destino en el atributo destinationCode. Si el valor del atributo destinationCode es una cadena vacía, el control es válido para todos los destinos en ese tmCircuitEndPointSubgroup.

El efecto de este control puede también limitarse al tráfico con encaminamiento directo o alternativo.

Los controlTmCircuitEndPointSubgroup, destinationCode, destinationType (si está presente), los componentes de originationAspect, los componentes de destinationAspect y los componentes de routingAspect son todos claves para el control, en ese orden.

Éste es un control manual expansivo. Se trata de una subclase de la clase de objeto trafficControl. Véase el cuadro 31.

Cuadro 31/Q.823 – Entidad tarTo

Atributos	M/S	O/M/C
controlTmCircuitEndPointSubgroup	S	M
newTmCircuitEndPointSubgroups	S	M
routingAspect	S	M
destinationAspect	S	M
originationAspect	S	M
percentage	S	M
destinationType	S	C
destinationCode	S	M
returnAction	S	M

Los siguientes atributos describen la entidad tarTo:

- controlTmCircuitEndPointSubgroup (subhaz de punto extremo de circuitos de gestión de tráfico de control) – Identifica el subhaz de circuitos al cual se aplica el control tarTo.
- newTmCircuitEndPointSubgroups (nuevos subhaces de punto extremo de circuitos de gestión de tráfico) – Secuencia de nuevos haces de circuitos. Debe estar presente al menos un elemento en la secuencia.

- routingAspect (aspecto de encaminamiento) – Identifica el aspecto de encaminamiento (tráfico con encaminamiento directo o tráfico con encaminamiento alternativo) para el que es válido este control. Si el valor de atributo es NULO, el control de tráfico se aplica a todos los aspectos de encaminamiento.
- destinationAspect (aspecto de destino) – Identifica el aspecto de destino (por ejemplo, HTR) para el cual es válido este control. Si este atributo tiene el valor NULO, el control de tráfico es válido para todos los aspectos de destino.
- originationAspect (aspecto de origen) – Identifica el origen y la categoría de parte llamante (de acuerdo con la Recomendación Q.763) para los que es válido este control. Si este atributo es la secuencia vacía, el control de tráfico es válido para todos los aspectos de origen.
- percentage (porcentaje) – Identifica el porcentaje de llamadas que se reencaminarán de resultas de la activación del control.
- destinationType (tipo de destino) – Identifica el tipo de destino. Es la naturaleza de la dirección en una cadena de siete bits de acuerdo con la Recomendación Q.763, o el tipo de destino como una lista enumerada.
- destinationCode (indicativo de destino) – Es un indicativo de país, o/y un indicativo de área, o/y una central, u/y otro número de lugar para el cual se aplica este TAR. Si el valor de atributo es una cadena vacía, el control es válido para todos los destinos en ese controlTmCircuitEndPointSubgroup.
- returnAction (acción de retorno) – Identifica la disposición del tráfico de desbordamiento desde los newTmCircuitEndPointSubgroups.

Existen tres posibilidades:

- poniendo returnAction al valor NULO se insertan los newTmCircuitEndPointSubgroups antes del controlTmCircuitEndpointSubgroup;
- poniendo returnAction en salto de ruta se sustituye el controlTmCircuitEndPointSubgroup por los TmCircuitEndPointSubgroups;
- poniendo returnAction en tratamiento se sustituirá el controlTmCircuitEndPointSubgroup y los subhaces de circuitos restantes en el plan de encaminamiento por los newTmCircuitEndPointSubgroups.

7.1.3.16 Control de tráfico (trafficControl)

La clase de objeto control de tráfico es una superclase para todas las clases de objeto que representan controles de tráfico, y que se definen en la Recomendación E.412. Esta superclase no es instanciada.

En caso de que se instancien varios controles del mismo tipo en el mismo recurso gestionado, se describe un mecanismo (denominado clave) en cada subclase de control de tráfico para seleccionar una instancia del control. Tiene precedencia el control que tiene los criterios más específicos.

Cuando están activos varios controles, deberá aplicarse la jerarquía de controles especificada en la Recomendación E.412. Hasta que se incorpore a dicha Recomendación la citada jerarquía, se utilizará el apéndice X como directriz. Véase el cuadro 32.

Cuadro 32/Q.823 – Entidad trafficControl

Atributos	M/S	O/M/C
trafficControlId	S	M
creatorIdentity	S	O
tmSurveillance	S	M

Los siguientes atributos describen la entidad trafficControl:

- trafficControlId (Id de control de tráfico) – Este atributo es el identificador de objeto. Se utiliza como el RDN.
- creatorIdentity (identidad del creador) – Identifica quién creó la instancia de objeto.
- tmSurveillance (vigilancia de tm) – Indica si está siendo vigilada la instancia de control de tráfico. Si el valor de atributo es VERDADERO, la tmTrafficControlCurrentData es automáticamente instanciada por el NE. Si el valor de atributo es FALSO, la tmTrafficControlCurrentData es automáticamente suprimida por el NE.

7.1.4 Fragmento de estado

La finalidad del fragmento de estado es proporcionar un mecanismo para recibir un conjunto finito de información de situación de diversos eventos/actividades/anomalías de NE que sean de interés para un gestor de red en una forma condensada a intervalos periódicos, pero bastantes breves.

7.1.4.1 Indicador de estado (stateIndicator)

Esta clase de objeto define una acción y notificación que contiene un bit para cada condición. Si en cualquier instante durante el periodo la condición ha estado presente, el valor del bit es 1; en otro caso, el valor del bit es 0. Se definen los siguientes bits:

- exchangeCongestionLevel1 (nivel de congestión de central 1): Este bit es igual a 1 si durante el granularityPeriod el NE experimenta el nivel de congestión 1.
- exchangeCongestionLevel2 (nivel de congestión de central 2): Este bit es igual a 1 si durante el granularityPeriod el NE experimenta el nivel de congestión 2.
- congestionLevel1Received (recibido nivel de congestión 1): Este bit es igual a 1 si durante el granularityPeriod el NE recibe el indicador CL1 de cualesquiera centrales adyacentes.
- congestionLevel2Received (recibido nivel de congestión 2): Este bit es igual a 1 si durante el granularityPeriod el NE recibe el indicador CL2 de cualesquiera centrales adyacentes.
- scrTriggered (disparada scr): Este bit es igual a 1 si durante el granularityPeriod el administrativeState está desbloqueado y autoActivated es VERDADERO para al menos una instancia de control SCR.
- accTriggered (disparado acc): Este bit es igual a 1 si durante el granularityPeriod el administrativeState está desbloqueado y autoActivated es VERDADERO para al menos una instancia de control acc.
- protectiveControlActive (control de protección activo): Este bit es igual a 1 si durante el granularityPeriod están presentes en el NE una o más instancias de los controles de protección manuales (es decir, cancelTo, cancelFrom, skip, cancelRerouted).
- expansiveControlActive (control de expansión activo): Este bit es igual a 1 si durante el granularityPeriod están presentes en el NE una o más instancias de controles de expansión manuales (es decir, tarTo, tarFrom).

- i) `destinationControlActive` (control de destino activo): Este bit es igual a 1 si durante el `granularityPeriod` están presentes en el NE una o más instancias de controles de destino manuales (es decir, funciones `destinationCodeControl`, `adc`).
- j) `htrDestinationActive` (destino htr activo): Este bit es igual a 1 durante el `granularityPeriod` si están presentes en el NE una o más instancias de `htrDestination`.
- k) `circuitEndPointSubgroupAddedOrDeleted` (subhaz de punto de extremo de circuitos añadido o suprimido): Este bit es igual a 1 durante el `granularityPeriod` cuando se añade o suprime en el NE un subhaz de circuitos.
- l) `accTransmissionInhibited`: Este bit es igual a 1 si durante el `granularityPeriod` se pone en estado bloqueado el `administrativeState` de `accTrigger`.
- m) `adcTriggered`: Este bit es igual a 1 si durante `granularityPeriod` se desbloquea `administrativeState` y `autoActivated` es VERDADERO durante al menos una instancia de control `adc`.

El `granularityPeriod` de esta clase de objeto será menor o igual que el utilizado para la `currentData` definido en esta Recomendación, y se especificará en segundos o minutos.

7.1.5 Fragmento administrativo

7.1.5.1 Discriminador de retransmisión de eventos

La clase de objeto gestionado discriminador de retransmisión de eventos (`event forwarding discriminator`) se define en la Recomendación X.721.

7.1.5.2 Escáner simple

La clase de objeto gestionado escáner simple (`simple scanner`) se define en la Recomendación X.738.

7.2 Definición de los atributos comunes

Esta subcláusula contiene la descripción de todos los atributos genéricos utilizados dentro de este modelo de información. Se han identificado los siguientes atributos genéricos, cuya definición puede verse en las normas apropiadas referenciadas en esta Recomendación:

- nombre distinguido relativo;
- atributos de estado.

Los atributos específicos de este modelo de información se definen como parte de las definiciones de clase de objeto.

7.2.1 Nombre distinguido relativo

La semántica del tipo de atributo nombre distinguido relativo se especifica en la Recomendación X.720. Este tipo de atributo se utiliza para identificar una instancia de un objeto gestionado unívocamente dentro del contexto de su objeto gestionado superior. Las constricciones en la sintaxis y semántica del atributo se especifican en la Recomendación X.720.

7.2.2 Atributos de estado

Los atributos relacionados con el estado de los objetos gestionados en este modelo de información están compuestos por los atributos de modelo de estado genérico definidos en la Recomendación X.731, y cada atributo de estado específico que es especificado para cualquiera de las clases de objeto específicas definidas en esta Recomendación.

7.2.2.1 Estado administrativo

La semántica del atributo `administrativeState` se especifica en la Recomendación X.731. La sintaxis del atributo `administrativeState` se especifica en la Recomendación X.721.

7.3 Descripción de las acciones

Todas las acciones descritas a continuación se efectúan en las clases de objeto identificadas en el cuadro.

Acción	Clase de objeto afectado	Observaciones
<code>stateIndicatorAction</code>	<code>stateIndicator</code>	Ninguna

7.4 Descripción de las notificaciones

Las siguientes notificaciones genéricas utilizan el modelo de información definido en esta Recomendación:

Notificación	Recomendaciones con la definición
Notificación de creación de objeto	X.721 y X.730
Notificación de supresión de objeto	X.721 y X.730
Notificación de cambio de valor de atributo	X.721 y X.730
Notificación de cambio de estado	X.721 y X.731
Notificación de informe de exploración	X.738

La siguiente notificación específica es utilizada por el modelo de información definido en esta Recomendación:

Notificación	Clase de objeto afectado	Observaciones
<code>stateIndicatorNotification</code>	<code>stateIndicator</code>	Ninguna

8 Unidades funcionales

Se definen en esta Recomendación las siguientes unidades funcionales:

- A) Unidad funcional control de tráfico (traffic control functional unit): Esta unidad funcional permite a un gestor habilitar/inhabilitar los controles de tráfico y modificar los atributos y disparadores de estos controles (crear/suprimir/fijar subclases de `trafficControl`, tablas de parámetros y disparadores).
- B) Unidad funcional mediciones de control de tráfico: Esta unidad funcional permite a un gestor recoger las mediciones para controles de tráfico.
- C) Unidad funcional mediciones de central: Esta unidad funcional permite a un gestor recoger el conjunto de mediciones de tráfico de central.
- D) Unidad funcional mediciones de subhaz de punto extremo de circuitos: Esta unidad funcional permite a un gestor recoger mediciones para subhaces de punto extremo de circuitos.

- E) Unidad funcional mediciones de destino: Esta unidad funcional permite a un gestor recoger mediciones para destinos.
- F) Unidad funcional acceso a datos históricos: Esta unidad funcional permite a un gestor efectuar tomas en subclases de datos históricos.
- G) Unidad funcional difícil de alcanzar: Esta unidad funcional permite a un gestor determinar que algunos destinos son difíciles de alcanzar (crear/suprimir/fijar instancias de `htrDestination`).
- H) Unidad funcional acceso a datos de referencia: Esta unidad funcional permite a un gestor acceder a instancias de clases de objeto distintas de `historyData` o `currentData` utilizando la operación GET (OBTENCIÓN).
- I) Unidad funcional notificación de indicador de estado autónomo: Esta unidad funcional requiere el soporte de la `stateIndicatorNotification` definida en esta Recomendación.
- J) Unidad funcional notificación de indicador de estado pedido: Esta unidad funcional requiere el soporte de la `stateIndicatorAction` definida en esta Recomendación.
- K) Habilitación/inhabilitación de mediciones: Esta unidad funcional permite a un gestor especificar las distintas entidades de un tipo para el que han de recogerse mediciones (fijar atributo `tmSurveillance`) y/o la creación de una o más subclases `currentData`.

A fin de asegurar la interoperabilidad, el gestor y el agente tienen que compartir al menos una unidad funcional común y un conjunto común de clases de objetos.

El apéndice VI proporciona una correspondencia de las funciones de gestión de tráfico (enumeradas en el apéndice I) con unidades funcionales.

8.1 Unidades funcionales de otras Recomendaciones

Además de las unidades funcionales citadas, esta Recomendación soporta las siguientes unidades funcionales procedentes de otras Recomendaciones del CCITT:

- i) Unidad funcional estimulación de exploración (Recomendación X.738)
- ii) Unidad funcional informe de eventos de sumario (Recomendación X.738)
- iii) Unidad funcional gestión de informe de evento (Recomendación X.734)

8.2 Negociación de unidades funcionales

Esta Recomendación asigna el siguiente valor de identificador de objeto.

```
{joint-iso-ccitt ms(9) function Recommendation(0) q(17) q823(823) functionalUnitPackage(1)}
```

como un valor del `FunctionalUnitPackageId` de tipo ASN.1 definido en la Recomendación X.701 para negociar la disponibilidad de una de las siguientes unidades funcionales:

- 0 Unidad funcional control de tráfico
- 1 Unidad funcional mediciones de control de tráfico
- 2 Unidad funcional mediciones de central
- 3 Unidad funcional mediciones de subhaz de punto extremo de circuitos
- 4 Unidad funcional mediciones de destino
- 5 Unidad funcional acceso a datos históricos
- 6 Unidad funcional difícil de alcanzar
- 7 Unidad funcional acceso a datos de referencia
- 8 Unidad funcional notificación de indicador de estado autónomo

9 Unidad funcional notificación de indicador de estado pedido

10 Habilitación/inhabilitación de mediciones

donde el número indica la posición de bit en la CADENA DE BITS asignada a las unidades funcionales, y los nombres referencian las unidades funcionales.

Dentro del contexto de aplicación de gestión de sistemas, el mecanismo para negociar las unidades funcionales se describen en la Recomendación X.701.

9 Relación con otras Recomendaciones

- Esta Recomendación utiliza definiciones de la Recomendación X.721, Definición de la información de gestión.
- Esta Recomendación utiliza servicios definidos en la Recomendación X.730, Función de gestión de objetos, para la creación y supresión de objetos gestionados, recuperación y actualización de atributos, las notificaciones sobre creación de objetos, supresión de objetos y cambios de valores de atributos.
- Esta Recomendación utiliza servicios definidos en la Recomendación X.731, Función de gestión de estados, para la notificación de cambios de estado.
- Esta Recomendación utiliza definiciones y servicios de la Recomendación X.734, Función de gestión de informes de evento.
- Esta Recomendación utiliza definiciones y servicios de la Recomendación X.738, Función de sumario.
- Esta Recomendación utiliza definiciones y servicios de la Recomendación Q.822, Gestión de la calidad de funcionamiento.

10 Conformidad

Las implementaciones que se pretenden conformes con esta Recomendación cumplirán los requisitos de conformidad definidos en las subcláusulas siguientes.

10.1 Conformidad estática

La implementación cumplirá los requisitos de esta Recomendación en el cometido de gestor, el cometido de agente o en ambos cometidos. El enunciado de conformidad de la implementación pretenderá conformidad con al menos un cometido cuando el ICS está disponible.

Para pretender conformidad en un cometido de gestor o en un cometido de agente, una implementación soportará al menos una de las siguientes capacidades:

- supervisión de calidad de funcionamiento, o
- control.

La supervisión de calidad de funcionamiento requiere al menos una de las unidades funcionales i), ii) o F), y al menos una unidad funcional B), C), D) o E). La capacidad administrativa y de auditoría puede incluirse opcionalmente en una implementación.

NOTA – Si se negocia la capacidad de auditoría, puede considerarse negociar la unidad funcional selección de objeto múltiple y la unidad funcional filtro de la Recomendación X.710.

Capacidades	Unidades funcionales	Clases de objetos
Supervisión de calidad de funcionamiento	i), ii), F), B), C), D), E)	Objetos subclasados a partir de historyData para vigilancia (por ejemplo, tmObservedDestinationHistoryData, tmExchangeHistoryData, tmTrafficControlHistoryData)
Aplicación de control	A)	Objetos subclasados a partir de trafficControl (por ejemplo, acc, adc, cancelFrom), disparadores (por ejemplo accTrigger), y tablas de parámetros (por ejemplo, accAffectedTraffic)
Auditoría	i), H)	Objetos que modelan recursos y controles (por ejemplo, circuitEndPointSubgroups, simpleScanner, cancelFrom)
	D), J)	stateIndicator
Administración		Objetos gestionados para fines NTM:
	iii)	edf
	(nota)	simpleScanner
	G)	htrDestination
NOTA – La actual Recomendación X.738 no incluye la gestión de subclases de escáner a la granularidad deseada para la gestión de tráfico. Por tanto, la gestión de subclases de escáner necesita ser negociada en el momento de la implementación.		

Los objetos gestionados soportados por el sistema abierto para la gestión de tráfico cumplirán la sintaxis y la semántica del modelo de información especificada en la cláusula 9.

La implementación soportará la sintaxis de transferencia derivada de las reglas de codificación especificadas en la Recomendación X.209 denominada {joint-iso-ccitt asn1(1) basicEncoding(1)} para los tipos de datos abstractos referenciados por las definiciones para las que se pretende soporte.

10.2 Conformidad dinámica

El sistema, en el cometido o cometidos, en los que se pretende conformidad, soportará los elementos de procedimiento definidos en:

- Recomendación X.730 para los servicios PT-OBTENCIÓN (PT-GET), PT-CREACIÓN (PT-CREATE), PT-SUPRESIÓN (PT-DELETE), PT-FIJACIÓN (PT-SET).
- Recomendación X.730 para los servicios informe creación de objeto, informe supresión de objeto, e informe cambio de valor de atributo si se soporta la clase de objeto apropiada.
- Recomendación X.731 para el servicio informe cambio de estado si se soporta la clase de objeto apropiada.

10.3 Requisitos de enunciado de conformidad de implementación de gestión

Cualquier formulario de MCS, formulario de MICS, formulario de MOCS, o formulario de PICS que cumpla esta Recomendación será técnicamente idéntico a los formularios especificados en futuros anexos de esta Recomendación. Deben mantener la numeración de los cuadros y la numeración de los puntos, y diferir sólo en la paginación y en los encabezamientos de página.

El suministrador de una implementación que se pretende conforme con esta Recomendación completará un ejemplar del sumario de conformidad de gestión (MCS, *management conformance summary*) (a suministrar) como parte de los requisitos de conformidad, junto con cualesquiera otros formularios ICS referenciados, aplicables desde ese MCS. Un ICS que cumple la Recomendación:

- describirá una implementación que cumpla esta Recomendación,

- habrá sido rellenado de acuerdo con las instrucciones para hacerlo, indicadas en la Recomendación X.724, e
- incluirá la información necesaria para identificar de forma única al suministrador y la implementación.

Las pretensiones de conformidad con la información de gestión definida en esta Recomendación en clases de objetos gestionados definidas en otro lugar incluirán los requisitos del formulario de MIDS, que se especificarán en el futuro MOCS para la clase de objeto gestionado.

11 Definiciones formales de clases de objeto

11.1 Definición de clases de objeto

11.1.1 ACC

acc MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

"ITU-T Rec. M.3100":stateChangeNotificationPackage,
accPackage PACKAGE

BEHAVIOUR

accBehaviour BEHAVIOUR

DEFINED AS

"The automatic congestion control is a preassigned control which is automatically activated when a congestion trigger is received from the adjacent exchange. It is defined in 4.1/E.412. At the receiving exchanges, all circuit subgroups that terminate to the congested exchange should be subjected to traffic controls which will limit the amount of traffic sent to the congested exchange. Disposition of calls that may be used to throttle traffic are typically cancel or skip. This control relies on an automatic timer set internally by the NE. When the congestion indicator is received, a timer is started in the receivingexchange. Subsequently received congestion indicators restart the timer. If no subsequent congestion indicator is received within the timer, the control is automatically deactivated. The selection of the timer value is vendor specific and is not manageable by an OS. The timer aspects for ACC are described in 5.5.2 b)/Q.542. If a different dispositionOfCalls for each congestion level is desired, a separate acc control object should be instantiated. Furthermore, instances of acc having the same congestion level and tmCircuitEndPointSubgroup must use the same dispositionOfCall.

The origination aspect, destination aspect, routing aspect and additional criteria, are all keys in that order. An instance of accAffectedTraffic pointed to by assocAccAffectedTraffic must exist before an acc object class is instantiated.

This is a protective control. It is a subclass of trafficControl object class.";;

ATTRIBUTES

controlTmCircuitEndPointSubgroup GET,
dispositionOfCalls GET-REPLACE,

"CCITT Rec. X.721:1992":administrativeState

DEFAULT

VALUE

Q823-TM-

ASN1Module.defaultAdministrativeState

PERMITTED VALUES Q823-TM-ASN1Module.PermittedState

GET-REPLACE,

autoActivated

GET,

assocAccAffectedTraffic

GET-REPLACE;

;;

REGISTERED AS {q823ObjectClass 1};

11.1.2 accAffectedTraffic

accAffectedTraffic MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":objectManagementNotificationsPackage,
accAffectedTrafficPackage PACKAGE

BEHAVIOUR

accAffectedTrafficBehaviour BEHAVIOUR

DEFINED AS

"The accAffectedTraffic object identifies the strength of the control for various congestion levels for a combination of routing aspect, destination aspect, origination aspect and/or additional criteria specified. If an instance of accAffectedTraffic is deleted, all acc object instances referring to this instance must be deleted.";;

ATTRIBUTES

accAffectedTrafficId GET,
cl1ResponseCategories GET-REPLACE,
cl2ResponseCategories GET-REPLACE;
;;

REGISTERED AS {q823ObjectClass 2};

11.1.3 accTrigger

accTrigger MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":stateChangeNotificationPackage,
accTriggerPackage PACKAGE

BEHAVIOUR

accTriggerBehaviour BEHAVIOUR

DEFINED AS

"ACC trigger is resident in the congested exchange. When it is unlocked, it will emit congestionLevel indications to the adjacent exchanges for them to automatically activate the preassigned ACC so that traffic towards the congested exchange is throttled. When it is locked, the congested exchange will not emit the congestionLevel indication to the adjacent exchanges. Note that the congestionLevel is modelled in congestionLevelIndication object class.";;

ATTRIBUTES

accTriggerId GET,
"CCITT Rec. X.721:1992":administrativeState
DEFAULT VALUE Q823-TM-ASN1Module.defaultAdministrativeState
PERMITTED VALUES Q823-TM-ASN1Module.PermittedState
GET-REPLACE;
;;

REGISTERED AS {q823ObjectClass 3};

11.1.4 adc

adc MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

"ITU-T Rec. M.3100":stateChangeNotificationPackage,
adcPackage PACKAGE

BEHAVIOUR

adcBehaviour BEHAVIOUR

DEFINED AS

"ADC is of two types: centralized and decentralized.

In ADC centralized, detection of congestion to a destination code is done at the destination node when the call arrival rate exceeds the threshold set for that destination code. It is described in 4.3/E.412. The adc trigger identifies when the threshold is exceeded, which results in an indication to be sent to each originating node via signalling message indicating that the offered traffic to that destination is to be reduced.

The signalling message specifies the details of the control along with the expiration time of the control. Upon receiving the message, the receiving node will instantiate the ADC object.

In the ADC decentralized, destination congestion is detected at the source NE by observing the bids and answers for a selected destination code. It is described in 4.3/E.412. The ADC Trigger (adcTrigger) object identifies the destination to be monitored, the threshold value and the strength of the control. When the threshold value is crossed at the source NE, it will automatically instantiate the ADC object. If the ADC

control is locked, it will not become active even if the Trigger is active. The ADC control will be active only when the administrativeState is set to unlocked state.

The type of ADC is identified by the adcType attribute which can take the value of "centralized" or "decentralized".

This control object is instantiated only by the NE and not by the OS when either it receives a message from the destination node (in the case of centralized) or when the threshold value is crossed at the source NE (in the case of decentralized).

Destination type and destination code are keys to the control in that order.

Exactly one of the strength attributes (adcPercentage or adcContinuousTimer or adcAsynchronousTimer or adcLeakyBucket), must be present when the control object is instantiated. The adc object instance is deleted at the end of time expiration. If a subsequent trigger is received prior to time expiration, the adc instance is replaced with the new values identified in the latest trigger. The value of autoActivated is always TRUE. This object is a subclass of the trafficControl object class.";;

ATTRIBUTES

```
adcType          GET,
destinationCode GET,
treatment        GET,
"CCITT Rec. X.721:1992":administrativeState
    DEFAULT VALUE Q823-TM-ASN1Module.defaultAdministrativeState
    PERMITTED VALUES Q823-TM-ASN1Module.PermittedState
    GET-REPLACE,
autoActivated    GET,
timeExpiration   GET;
```

;;

CONDITIONAL PACKAGES

```
destinationTypePackage
PRESENT IF "destination type is required to unambiguously identify the destinationCode",
adcPercentagePackage
PRESENT IF "adcContinuousTimerPackage and adcAsynchronousTimerPackage and
adcLeakyBucketPackage are not present, and NE supports it.",
adcContinuousTimerPackage
PRESENT IF "adcPercentagePackage and adcAsynchronousTimerPackage and
adcLeakyBucketPackage are not present, and NE supports it.",
adcAsynchronousTimerPackage
PRESENT IF "adcPercentagePackage and adcContinuousTimerPackage and
adcLeakyBucketPackage are not present, and NE supports it.",
adcLeakyBucketPackage
PRESENT IF "adcPercentagePackage and adcContinuousTimerPackage and
adcAsynchronousTimerPackage are not present, and NE supports it.";
```

REGISTERED AS {q823ObjectClass 4};

11.1.5 adcTrigger

adcTrigger MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

```
"ITU-T Rec. M.3100":objectManagementNotificationsPackage,
"ITU-T Rec. M.3100":stateChangeNotificationPackage,
adcTriggerPackage PACKAGE
```

BEHAVIOUR

adcTriggerBehaviour BEHAVIOUR

DEFINED AS

"The ADC trigger object is set up by the OS with the necessary attributes so that the ADC object may be automatically instantiated (in the same NE where the adcTrigger resides or adjacent NE) when the trigger threshold value is crossed. OS can specify the destination which is to be monitored by the NE for congestion, the trigger threshold value, the strength for the control and the time expiration.

The adcTrigger can be of type "centralized" or "decentralized" or "both" as identified in adcType attribute.

If the adcType is "centralized", the adcTrigger and the adc control are in two different NEs. A signalling message is sent from the source NE (where the adcTrigger object resides) to all the adjacent NEs to create the adc control.

If the adcType is "decentralized", the adcTrigger and the adc control are in the same NE. The source NE contains both the adcTrigger and the adc control.

If the adcType is "both", the source NE (where the adcTrigger resides) sends a signalling message to all the adjacent NEs (i.e. centralized adc) to create an adc control. It will instantiate the adc control within itself (decentralized adc).

An OS can instantiate the ADC trigger object, but not the ADC control object. The ADC trigger can also be set to the locked state. When it is in the locked state, it will not automatically send a message to the adjacent exchange or will not instantiate the ADC control even if the trigger threshold is crossed.

Exactly one strength attribute (percentage or continuousTimer or asynchronousTimer or leakyBucket) must be present when this object is instantiated.

Both destination type and destination code are keys to the control in that order.";;

ATTRIBUTES

adcTriggerId GET,
adcTriggerType GET-REPLACE,
"CCITT Rec. X.721:1992":administrativeState
DEFAULT VALUE Q823-TM-ASN1Module.defaultAdministrativeState
PERMITTED VALUES Q823-TM-ASN1Module.PermittedState
GET-REPLACE,
timeExpiration GET-REPLACE;
;;

CONDITIONAL PACKAGES

destinationCodePackage
PRESENT IF "if the triggering of the adc control is restricted to a predetermined destination.",
destinationTypePackage
PRESENT IF "destinationCodePackage is present and destination type is required to unambiguously identify the destinationCode.",
triggerThresholdPackage
PRESENT IF "a threshold is to be defined by the manager for adcTrigger to activate",
percentagePackage
PRESENT IF "continuousTimerPackage and asynchronousTimerPackage and leakyBucketPackage are not present, and NE supports it.",
continuousTimerPackage
PRESENT IF "percentagePackage and asynchronousTimerPackage and leakyBucketPackage are not present, and NE supports it.",
asynchronousTimerPackage
PRESENT IF "percentagePackage and continuousTimerPackage and leakyBucketPackage are not present, and NE supports it.",
leakyBucketPackage
PRESENT IF "percentagePackage and continuousTimerPackage and asynchronousTimerPackage Are not present, and NE supports it.";

REGISTERED AS {q823ObjectClass 5};

11.1.6 cancelFrom

cancelFrom MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

cancelFromPackage PACKAGE

BEHAVIOUR

cancelFromBehaviour BEHAVIOUR

DEFINED AS

"This control is manually activated on an outgoing circuit subgroup and prohibits traffic from overflowing to the next in-chain circuit subgroups. It is defined in 3.2.1/E.412.

This is a protective circuit subgroup control. It is a subclass of trafficControl object class.

Cancel From control can be used for prohibiting overflow of both direct routed traffic and alternate traffic. The distinction between these two types of traffic is made via the assignment of appropriate traffic types to the routingAspect attribute.

The controlTmCircuitEndPointSubgroup attribute is the key with the highest priority, followed by each component in the originationAspect, destinationAspect and routingAspect in that order.

The more specific value has the priority.";;

ATTRIBUTES

controlTmCircuitEndPointSubgroup GET,
routingAspect REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultRoutingAspects
GET-REPLACE,
destinationAspect REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultDestinationAspect
GET-REPLACE,
originationAspect REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
GET-REPLACE,
percentage GET-REPLACE,
treatment GET-REPLACE;
;;

REGISTERED AS {q823ObjectClass 6};

11.1.7 cancelRerouted

cancelRerouted MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

cancelReroutedPackage PACKAGE

BEHAVIOUR

cancelReroutedBehaviour BEHAVIOUR

DEFINED AS

"This control prevents additional rerouting or alternate routing of calls which have been already rerouted. Rerouted calls are not allowed to overflow the circuit subgroup to which the Cancel Rerouted Overflow control is applied, while normal overflow traffic is not affected. It is defined in 3.2.4/E.412.

Note that in order to provide this control, each call has to be marked via signalling message whenever a TAR is applied to that call.

This is a protective manual control. It is a subclass of trafficControl object class.";;

ATTRIBUTES

controlTmCircuitEndPointSubgroup GET,
treatment GET-REPLACE
;;

;

REGISTERED AS {q823ObjectClass 7};

11.1.8 cancelTo

cancelTo MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

cancelToPackage PACKAGE

BEHAVIOUR

cancelToBehaviour BEHAVIOUR

DEFINED AS

"This control covers both cancellation of direct routing and cancellation of alternative routing to a given controlTmCircuitEndPointSubgroup.

The cancellation of direct routing blocks the amount of direct routed traffic accessing a circuit subgroup. It is defined in 3.1.2/E.412.

The cancellation of alternative routing to control is activated on an outgoing circuit subgroup and prohibits overflow traffic from accessing the controlled circuit subgroup. It is defined in 3.2.1/E.412.

The distinction between these two types is made via the assignment of appropriate routingAspect attribute. Appendix IV provides an explanation of how to prohibit direct and alternate routed traffic from accessing a circuit subgroup using Cancel To control.

The controlTmCircuitEndPointSubgroup attribute is the key with the highest priority, followed by each component in the originationAspect, destinationAspect and routingAspect in that order.

The more specific value has the priority.

This is a protective manual control. It is a subclass of the trafficControl object class.";;

ATTRIBUTES

controlTmCircuitEndPointSubgroup routingAspect GET,
REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultRoutingAspects
destinationAspect GET-REPLACE,
REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultDestinationAspect
originationAspect GET-REPLACE,
REPLACE-WITH-DEFAULT
DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
percentage GET-REPLACE,
treatment GET-REPLACE,
GET-REPLACE;
;;

REGISTERED AS {q823ObjectClass 8};

11.1.9 CircuitEndPointSubgroupCurrentData

circuitEndPointSubgroupCurrentData MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. Q.822:1994":currentData;

CHARACTERIZED BY

circuitEndPointSubgroupCurrentDataPackage PACKAGE

BEHAVIOUR

circuitEndPointSubgroupCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The circuit subgroup current data is a subclass of the currentData object class. It is used for monitoring circuit subgroup related performance data as defined in Recommendation E.502.

The performance monitoring attributes for the circuit subgroup are based on the circuit subgroup directionality characteristic, which can be one-way-out, one-way-in or two-way.

For traffic management purposes, this object class shall not be instantiated, but its subclass defined in this Recommendation or any subclass derived from that may be instantiated.";;

ATTRIBUTES

incomingSeizures INITIAL VALUE Q823-TM-ASN1Module.initialCount,
outgoingBids INITIAL VALUE Q823-TM-ASN1Module.initialCount,
outgoingSeizures INITIAL VALUE Q823-TM-ASN1Module.initialCount,
answeredOutgoingSeizures INITIAL VALUE Q823-TM-ASN1Module.initialCount,
overflow INITIAL VALUE Q823-TM-ASN1Module.initialCount,
incomingTrafficUsage INITIAL VALUE Q823-TM-ASN1Module.initialInteger,
outgoingTrafficUsage INITIAL VALUE Q823-TM-ASN1Module.initialInteger,
numberOfAvailCircuits INITIAL VALUE Q823-TM-ASN1Module.initialGauge;
;;

CONDITIONAL PACKAGES

answeredIncomingSeizuresPackage

PRESENT IF "this performance measurement is supported by the NE";

REGISTERED AS {q823ObjectClass 9};

11.1.10 circuitEndPointSubgroupHistoryData

circuitEndPointSubgroupHistoryData MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. Q.822:1994":historyData;

CHARACTERIZED BY

circuitEndPointSubgroupHistoryDataPackage PACKAGE

BEHAVIOUR

circuitEndPointSubgroupHistoryDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic management circuitEndPointSubgroup history data is a subclass of the historyData object class. It is used for monitoring circuitSubgroup related performance data as defined in Recommendation E.502.

The performance monitoring attributes are the same as those in the corresponding currentData.";;

ATTRIBUTES

incomingSeizures	GET,
outgoingBids	GET,
outgoingSeizures	GET,
answeredOutgoingSeizures	GET,
overflow	GET,
incomingTrafficUsage	GET,
outgoingTrafficUsage	GET,
numberOfAvailCircuits	GET;

;;

CONDITIONAL PACKAGES

answeredIncomingSeizureHistoryPackage

PRESENT IF "this package is present in the corresponding currentData";

REGISTERED AS {q823ObjectClass 10};

11.1.11 congestionLevelIndication

congestionLevelIndication MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":attributeValueChangeNotificationPackage,
congestionLevelIndicationPackage PACKAGE

BEHAVIOUR

congestionLevelIndicationBehaviour BEHAVIOUR

DEFINED AS

"Congestion Level Indication provides an indication of the current congestion level of the managed element object instance in which it is contained. This object instance is not allowed to be instantiated by an OS.";;

ATTRIBUTES

congestionLevelIndicationId	GET,
congestionLevel	GET;

;;

REGISTERED AS {q823ObjectClass 11};

11.1.12 destinationCodeControl

destinationCodeControl MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

destinationCodeControlPackage PACKAGE

BEHAVIOUR

destinationCodeControlBehaviour BEHAVIOUR

DEFINED AS

"This manual control bars routing for a specific destination either on a percentage basis (referred to in Recommendation E.412 as code blocking) or on a rate basis (referred to in Recommendation E.412 as call gapping).

Destination code control can be done on a country code, or/and an area code, or/and an exchange code or/and another location number. It is defined in 3.1.1.1/E.412.

When this control is active in the code blocking mode, the specified percentage of calls is blocked (cancelled).

When this control is active in the call gapping mode, the rate at which calls are released to the destination will be controlled. It can be specified in one of the three methods: continuous timer or asynchronous timer or leaky bucket.

The strength (code blocking or call gapping) of the destination code control can be within the same object class or can be defined in a separate object class. The former case is applicable when the strength of the

control is applied to only one destination. In this case, four possibilities (percentage or continuousTimer or asynchronousTimer or leakyBucket) exist for specifying the strength in the same object class, exactly one of which must be specified. Under this scenario, the single object instance acts as a code control function.

The latter case of defining the strength in a separate object class is applicable only when it is desired to apply the same strength to a group of destinations (leaky bucket is an example of where such may be useful). In this case, assocOwnerDccGroup attribute is required, which points to the object class containing the strength of the control. Under this scenario, two object classes together represent a code control function.

The following rules are valid independently whether one or two object classes are used to represent a destination code control function:

- It is not possible to create two or more destinationCodeControl object instances with an identical value combination of the key attributes destinationType, destinationCode, and originationAspect.
- The destinationCode attribute is the key with the highest priority, followed by the destinationType (if present) and components of the originationAspect attribute in that order.

When more than one control exists with overlapping digits, the more specific control of the same type applies instead of the less specific control, both of which could affect the same call.

If assocOwnerDccGroup attribute is specified, the associated dccGroup must exist. Instances of this object class belong to the same group if and only if they have the same value of the assocOwnerDccGroup.

This control is a subclass of the trafficControl object class.";;

ATTRIBUTES

```
destinationCode      GET,
originationAspect    REPLACE-WITH-DEFAULT
                     DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
                     GET-REPLACE,
treatment            GET-REPLACE;
;;
```

CONDITIONAL PACKAGES

```
destinationTypePackage
PRESENT IF "destination type is required to unambiguously identify the destinationCode.",
percentagePackage
PRESENT IF "continuousTimerPackage and asynchronousTimerPackage and leakyBucketPackage and
assocOwnerDccGroupPackage are not present, and the NE supports it.",
continuousTimerPackage
PRESENT IF "percentagePackage and asynchronousTimerPackage and leakyBucketPackage and
assocOwnerDccGroupPackage are not present, and the NE supports it.",
asynchronousTimerPackage
PRESENT IF "percentagePackage and continuousTimerPackage and leakyBucketPackage and
assocOwnerDccGroupPackage are not present, and the NE supports it.",
leakyBucketPackage
PRESENT IF "percentagePackage and continuousTimerPackage and asynchronousTimerPackage and
assocOwnerDccGroupPackage are not present, and the NE supports it.",
assocOwnerDccGroupPackage
PRESENT IF "percentagePackage and continuousTimerPackage and asynchronousTimerPackage and
leakyBucketPackage are not present, and the NE supports it.";
```

REGISTERED AS {q823ObjectClass 12};

11.1.13 dccGroup

dccGroup MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

```
"ITU-T Rec. M.3100":objectManagementNotificationsPackage,
dccGroupPackage PACKAGE
```

BEHAVIOUR

dccGroupBehaviour BEHAVIOUR

DEFINED AS

"This object class defines the strength of the destination code control when it is desired to apply the same strength to a group of destinations. Instances of this object class are associated with instances of the destinationCodeControl object class by an (1:n) relationship. This object class by itself does not provide the destination code control function; when associated with the destinationCodeControl instance(s), it represents the destination code control function.

Exactly one of the strength attribute (percentage or continuousTimer or asynchronousTimer or leakyBucket) must be present when the control object is instantiated.";;

ATTRIBUTES

dccGroupId **GET;**

;;

CONDITIONAL PACKAGES

percentagePackage

PRESENT IF "continuousTimerPackage and asynchronousTimerPackage and leakyBucketPackage are not present, and the NE supports it.",

continuousTimerPackage

PRESENT IF "percentagePackage and asynchronousTimerPackage and leakyBucketPackage are not present, and the NE supports it.",

asynchronousTimerPackage

PRESENT IF "percentagePackage and continuousTimerPackage and leakyBucketPackage are not present, and the NE supports it.",

leakyBucketPackage

PRESENT IF "percentagePackage and continuousTimerPackage and asynchronousTimerPackage are not present, and the NE supports it.";

REGISTERED AS {q823ObjectClass 13};

11.1.14 exchangeCurrentData

exchangeCurrentData **MANAGED OBJECT CLASS**

DERIVED FROM "ITU-T Rec. Q.822:1994":currentData;

CHARACTERIZED BY

exchangeCurrentDataPackage **PACKAGE**

BEHAVIOUR

exchangeCurrentDataBehaviour **BEHAVIOUR**

DEFINED AS

"The exchange current data is a subclass of the currentData object class. It is used for monitoring exchange related performance data as defined in Recommendation E.502.

For traffic management purposes, this object class shall not be instantiated, but its subclass defined in this Recommendation or any subclass derived from that shall be instantiated.

The performance monitoring attributes for the exchange is based on the main traffic flow characteristics defined in Figure 4/E.502.";

ATTRIBUTES

incomingTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

outgoingTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

transitTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

terminatingTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

originatingTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

internalTraffic

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

callsBlockedByLoadShedding

INITIAL VALUE Q823-TM-ASN1Module.initialCount;

;;

REGISTERED AS {q823ObjectClass 14};

11.1.15 exchangeHistoryData

exchangeHistoryData **MANAGED OBJECT CLASS**

DERIVED FROM "ITU-T Rec. Q.822:1994":historyData;

CHARACTERIZED BY

exchangeHistoryDataPackage **PACKAGE**

BEHAVIOUR

exchangeHistoryDataBehaviour **BEHAVIOUR**

DEFINED AS

"The exchange history data is a subclass of the historyData object class. It is used for monitoring exchange related performance data as defined in Recommendation E.502.

The performance monitoring attributes are the same as those in the corresponding currentData.

The performance monitoring attributes for the exchange is based on the main traffic flow characteristics defined in Figure 4/E.502.";;

ATTRIBUTES

incomingTraffic	GET,
outgoingTraffic	GET,
transitTraffic	GET,
terminatingTraffic	GET,
originatingTraffic	GET,
internalTraffic	GET,
callsBlockedByLoadShedding	GET

;;

;
REGISTERED AS {q823ObjectClass 15};

11.1.16 htrDestination

htrDestination MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":objectManagementNotificationsPackage,
"ITU-T Rec. M.3100":stateChangeNotificationPackage,
creatorPackage,
htrDestinationPackage PACKAGE

BEHAVIOUR

htrDestinationBehaviour BEHAVIOUR

DEFINED AS

"An instance of the htrDestination object represents a destination identified as hard-to-reach. The decision whether a destination is hard or easy to reach is made either on the basis of external information (e.g. earthquake) or on the answer bid ratio or answer seizure ratio either by the OS or by the resource management of the exchange. The administrativeState attribute provides the opportunity to lock the hard-to-reach status so that it can be temporarily disregarded. The HTR status of a destination can also be correlated with circuit subgroups. If this attribute is an empty set, the destination is considered HTR via all possible circuit subgroups.

An instance of htrDestinaton may be either explicitly created by a manager (in the case of manual htr destination) or automatically created by the NE (in case of automatic determination by the agent). In this model, the mechanism for the recognition of the hard-to-reach status of a destination by the resource management of the exchange is a local matter and therefore is not modelled.

A destination for which no htrDestination is instantiated or which is inhibited (administrativeState = locked) is to be considered as non Hard-to-Reach.

All instances of the htrDestination object class form the HTR List.";;

ATTRIBUTES

htrDestinationId	GET,
destinationCode	GET,
"CCITT Rec. X.721:1992":administrativeState	DEFAULT VALUE Q823-TM-ASN1Module.defaultAdministrativeState PERMITTED VALUES Q823-TM-ASN1Module.PermittedState GET-REPLACE;

;;

CONDITIONAL PACKAGES

destinationTypePackage

PRESENT IF "destination type is required to unambiguously identify the destination code",

tmCircuitEndPointSubgroupListPackage

PRESENT IF "the htrDestination is correlated with tmCircuitEndPointSubgroups";

REGISTERED AS {q823ObjectClass 16};

11.1.17 observedDestination

observedDestination MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":objectManagementNotificationsPackage,
observedDestinationPackage PACKAGE

BEHAVIOUR

observedDestinationBehaviour BEHAVIOUR

DEFINED AS

"An instance of observedDestination is instantiated when it is to be monitored for performance management, such as determining Hard-to-Reach. A destination is a country, an area, an exchange or other location in which the called subscriber is located and that may be specified within the country. It is defined in Annex A/E.410. A destination can be also selectively observed on a set of circuit subgroups. If the tmCircuitEndPointSubgroupList is not present, the destination shall be observed for all circuit subgroups.";

ATTRIBUTES

observedDestinationId GET,
destinationCode GET,
tmSurveillance GET-REPLACE;
;;

CONDITIONAL PACKAGES

destinationTypePackage

PRESENT IF "destination type is required to unambiguously identify the destination code",

creatorPackage

PRESENT IF "if an instance supports it",

tmCircuitEndPointSubgroupListPackage

PRESENT IF "a destination's performance should be monitored in correlation with certain tmCircuitEndPointSubgroups";

REGISTERED AS {q823ObjectClass 17};

11.1.18 observedDestinationCurrentData

observedDestinationCurrentData MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. Q.822:1994":currentData;

CHARACTERIZED BY

observedDestinationCurrentDataPackage PACKAGE

BEHAVIOUR

observedDestinationCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The Observed Destination current data is a subclass of the currentData object class. It is used for monitoring destination related performance data as defined in Recommendation E.502.

For traffic management purposes, this object class shall not be instantiated, but its subclass defined in this Recommendation or a subclass derived from that.";

ATTRIBUTES

bids

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

outgoingSeizures

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

answeredOutgoingSeizures

INITIAL VALUE Q823-TM-ASN1Module.initialCount,

noCircuitsAvailable

INITIAL VALUE Q823-TM-ASN1Module.initialCount;

;;

CONDITIONAL PACKAGES

callsAffectedByDccPackage PRESENT IF "an instance supports it";

REGISTERED AS {q823ObjectClass 18};

11.1.19 observedDestinationHistoryData

observedDestinationHistoryData MANAGED OBJECT CLASS
DERIVED FROM "ITU-T Rec. Q.822:1994":historyData;
CHARACTERIZED BY

observedDestinationHistoryDataPackage PACKAGE

BEHAVIOUR

observedDestinationHistoryDataBehaviour BEHAVIOUR

DEFINED AS

"The observed destination history data is a subclass of the historyData object class. It is used for monitoring destination related performance data as defined in Recommendation E.502.

The performance monitoring attributes are the same as those in the corresponding currentData.";;

ATTRIBUTES

bids GET,
outgoingSeizures GET,
answeredOutgoingSeizures GET,
noCircuitsAvailable GET;

;;

CONDITIONAL PACKAGES

callsAffectedByDccHistoryPackage

PRESENT IF "this package is present if it is present in the corresponding instance of currentData";

REGISTERED AS {q823ObjectClass 19};

11.1.20 scr

scr MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

"ITU-T Rec. M.3100":stateChangeNotificationPackage,

scrPackage PACKAGE

BEHAVIOUR

scrBehaviour BEHAVIOUR

DEFINED AS

"The Selective Circuit Reservation Control enables an exchange to automatically give preference to specific traffic attributes over others (e.g. direct routed calls over alternate routed calls) at the moment later providing greater selectivity. The control is preassigned with thresholds of how many circuits should be kept idle for the various traffic types in a circuit subgroup. When the number of idle circuits or the idle capacity in the circuit subgroup is less than or equal to the reservation threshold, the call is either cancelled or skipped to the next circuit subgroup in the route chain. This control is defined in 4.2/E.412.

SCR control has the following operating variables:

reservation thresholds;

control response;

control action option.

The reservation thresholds and the related control response are determined by the activationThresholds attribute and by the associated scrAffectedTraffic object instance. The control action option for processing of calls denied access to the circuit subgroup is given in the dispositionOfCalls attribute.

When the number of circuits or the idle capacity in the circuit subgroup is less than or equal to the reservation threshold, the exchange shall check the indicated scrAffectedTraffic object instance to determine if call shall be controlled. The dispositions that may be used to throttle traffic are cancel or skip.

The defined levels in the activationThresholds attribute and the reference given in the associated scr affected traffic attributes shall correspond to a single or multi threshold control.

The scrAffectedTraffic object instance must be present before scr object class can be instantiated.

This is a protective automatic control. It is a subclass of trafficControl object class.";;

ATTRIBUTES

controlTmCircuitEndPointSubgroup GET,
dispositionOfCalls GET-REPLACE,
"CCITT Rec. X.721:1992":administrativeState

DEFAULT

VALUE

Q823-TM-ASN1Module

defaultAdministrativeState

```

PERMITTED VALUES Q823-TM-ASN1Module.PermittedState
    GET-REPLACE,
autoActivated          GET,
activationThresholds  GET-REPLACE,
assocScrAffectedTraffic GET-REPLACE;
;;
REGISTERED AS {q823ObjectClass 20};

```

11.1.21 scrAffectedTraffic

```

scrAffectedTraffic MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721:1992":top;
CHARACTERIZED BY
    "ITU-T Rec. M.3100":objectManagementNotificationsPackage,
    scrAffectedTrafficPackage PACKAGE
BEHAVIOUR
    scrAffectedTrafficBehaviour BEHAVIOUR
DEFINED AS
    "The selective circuit reservation affected traffic object class represents the control response category for
    the SCR control.
    It determines per individual routing aspect, destination aspect, origination aspect and/or additional
    criteria, the strength of the traffic to be controlled.
    If level2ResponseCategories is specified, level1 and level2 activationThresholds must be specified using the
    same unit (number or percentage).
    The origination aspect, destination aspect, routing aspect and additional criteria are keys in that order.";;
ATTRIBUTES
    scrAffectedTrafficId          GET,
    level1ResponseCategories GET-REPLACE;
;;
CONDITIONAL PACKAGES
    level2ResponseCategoriesPackage
    PRESENT IF "instance supports multi threshold selective circuit reservation";
REGISTERED AS {q823ObjectClass 21};

```

11.1.22 skip

```

skip MANAGED OBJECT CLASS
DERIVED FROM trafficControl;
CHARACTERIZED BY
    skipPackage PACKAGE
BEHAVIOUR
    skipBehaviour BEHAVIOUR
DEFINED AS
    "This control is activated on an outgoing circuit group and is used to force traffic to the next in-chain
    circuit subgroup in the routing table. The skip control can affect both direct and alternate routed traffic. It
    is defined in 3.2.2/E.412.
    The controlTmCircuitEndPointSubgroup attribute is the key with the highest priority, followed by each
    component in the originationAspect, destinationAspect and routingAspect in that order. The more specific
    value has the priority.
    This is a protective manual control. It is a subclass of trafficControl object class.";;
ATTRIBUTES
    controlTmCircuitEndPointSubgroup GET,
    routingAspect                    REPLACE-WITH-DEFAULT
                                     DEFAULT VALUE Q823-TM-ASN1Module.defaultRoutingAspects
                                     GET-REPLACE,
    destinationAspect                REPLACE-WITH-DEFAULT
                                     DEFAULT VALUE Q823-TM-ASN1Module.defaultDestinationAspect
                                     GET-REPLACE,
    originationAspect                REPLACE-WITH-DEFAULT
                                     DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
                                     GET-REPLACE,

```

percentage

GET-REPLACE

;;

;

REGISTERED AS {q823ObjectClass 22};

11.1.23 StateIndicator

stateIndicator MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. X.739:1993":scanner;

CHARACTERIZED BY

stateIndicatorPackage PACKAGE

BEHAVIOUR

stateIndicatorBehaviour BEHAVIOUR

DEFINED AS

"This object class defines action and notification containing a bit for each condition. If at any time during the period if the condition has been present, the value of the bit is 1; otherwise the value of the bit is 0. The following bits are defined:

a) exchangeCongestionLevel1: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 1.

b) exchangeCongestionLevel2: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 2.

c) congestionLevel1Received: This bit is equal to 1 if during the granularityPeriod the NE receives CL1 indicator from any adjacent exchanges.

d) congestionLevel2Received: This bit is set equal to 1 if during the granularityPeriod when the NE received CL2 indicator from any adjacent exchanges.

e) scrTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of SCR control.

f) accTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of acc control.

g) protectiveControlActive: This bit is equal to 1 if during the granularityPeriod one or more instances of manual protective controls (i.e. cancelTo, cancelFrom, skip, cancelRerouted) are present in the NE.

h) expansiveControlActive: This bit is equal to 1 if during the granularityPeriod one or more instances of manual expansive controls (i.e. tarTo, tarFrom) are present in the NE.

i) destinationControlActive: This bit is equal to 1 if during the granularityPeriod one or more instances of manual destination controls (i.e. destinationCodeControl functions) are present in the NE.

j) htrDestinationActive: This bit is equal to 1 if during the granularityPeriod one or more instances of htrDestination are present in the NE.

k) circuitEndPointSubgroupAddedOrDeleted: This bit is equal to 1 during the granularityPeriod when a circuit subgroup is added or deleted in the NE.

l) accTransmissionInhibited: This bit is equal to 1 if during the granularityPeriod the administrativeState of accTrigger is set to locked.

m) adcTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of adc control.

The granularityPeriod of this object class shall be less than or equal to the one used for the currentData defined in this Recommendation, and it will be specified in seconds or minutes.";;;

ATTRIBUTES

stateIndicatorId

GET;

CONDITIONAL PACKAGES

stateIndicatorActionPackage

PRESENT IF "stateIndicatorNotificationPackage is not present",

stateIndicatorNotificationPackage

PRESENT IF "stateIndicatorActionPackage is not present";

REGISTERED AS {q823ObjectClass 23};

11.1.24 tarFrom

tarFrom MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

tarFromPackage PACKAGE

BEHAVIOUR

tarFromBehaviour BEHAVIOUR

DEFINED AS

"Temporary Alternative Routing (TAR) From is an expansive control which temporarily increases the number of routing possibilities to complete calls by:

- adding new circuit subgroups at the end of routing table, or
- by inserting new circuit subgroups into the routing table between existing circuit subgroups to provide additional overflow paths which are not normally available in the normal routing plan.

These two types of TAR (add at the end or insert between circuit subgroups) are described in 3.2.3/E.412. returnAction of skip is not valid for tarFrom.

TarFrom is only applied at circuit subgroup level and impacts all routing tables where the controlTmCircuitEndPointSubgroup appears. The temporary alternative routing circuit groups must terminate on an exchange that has the capability of reaching the final destination.

The effect of this control can be limited to destinations which are on HTR List by setting the destinationAspect attribute, or by explicitly specifying the destination in the destinationCode. Attribute. If the destination code is an empty string, the control applies to destinations served by that circuit subgroup.

The effect of this control can also be limited to directed or alternated routed traffic.

controlTmCircuitEndPointSubgroup, destinationType (if present), destinationCode, components of originationAspect, components of destinationAspect and components of routingAspect are all keys to the control in that order.

This is expansive manual control. It is a subclass of trafficControl object class.";;

ATTRIBUTES

```
controlTmCircuitEndPointSubgroup  GET,
newTmCircuitEndPointSubgroups    GET-REPLACE,
routingAspect                     REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultRoutingAspects
                                GET-REPLACE,
destinationAspect                 REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultDestinationAspect
                                GET-REPLACE,
originationAspect                REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
                                GET-REPLACE,
percentage                       GET-REPLACE,
destinationCode                  DEFAULT VALUE Q823-TM-ASN1Module.defaultAllDestinationCodes
                                GET,
returnAction PERMITTED VALUES  Q823-TM-ASN1Module.PermittedTarFromReturnAction
                                GET
;;
```

;

CONDITIONAL PACKAGES

destinationTypePackage

PRESENT IF "destination type is required to unambiguously identify the destinationCode";

REGISTERED AS {q823ObjectClass 24};

11.1.25 tarTo

tarTo MANAGED OBJECT CLASS

DERIVED FROM trafficControl;

CHARACTERIZED BY

tarToPackage PACKAGE

BEHAVIOUR

tarToBehaviour BEHAVIOUR

DEFINED AS

"Temporary Alternative Routing (TAR) To is an expansive control which temporarily increases the number of routing possibilities to complete calls by adding new circuit subgroups at the beginning of routing table so that traffic will be first offered to the new circuit subgroup, or by replacing the existing circuit subgroup with a set of new circuit subgroups. One or several circuit groups, which are not normally available in the normal routing plan, but have idle capacity are made available.

These two types of TAR (add at beginning or replace existing circuit subgroup) are described in 3.2.3/E.412. Setting returnAction to NULL inserts the new TmCircuitEndPointSubgroups before the

controlTmCircuitEndPointSubgroup. Setting returnAction to skip replaces the controlTmCircuitEndPointSubgroup with the new TmCircuitEndPointSubgroups. Setting returnAction to Treatment will replace the controlTmCircuitEndPointSubgroup and the remaining circuit subgroups in the routing plan with the newTmCircuitEndPointSubgroups.

TarTo is only applied at the circuit subgroup level and impacts all routing tables where the controlTmCircuitEndPointSubgroup is present. The temporary alternative routing circuit groups must terminate on an exchange that has the capability of reaching the final destination.

The effect of this control can be limited to destinations which are on the HTR List by setting the routing aspect attribute, or by explicitly specifying the destination in the destinationCode Attribute. If the destinationCode attribute value is an empty string, the control is valid for all destinations on that controlTmCircuitEndPointSubgroup.

The effect of this control can also be limited to directed or alternated routed traffic.

controlTmCircuitEndPointSubgroup, destinationType (if present), destinationCode, components of originationAspect, components of destinationAspect and components of routingAspect are all keys to the control in that order.

This is expansive manual control. It is a subclass of trafficControl object class.";;

ATTRIBUTES

```
controlTmCircuitEndPointSubgroup  GET,
newTmCircuitEndPointSubgroups    GET-REPLACE,
routingAspect                     REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultRoutingAspects
                                GET-REPLACE,
destinationAspect                 REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultDestinationAspect
                                GET-REPLACE,
originationAspect                 REPLACE-WITH-DEFAULT
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultOriginAspect
                                GET-REPLACE,
percentage                         GET-REPLACE,
destinationCode                   GET-REPLACE,
                                DEFAULT VALUE Q823-TM-ASN1Module.defaultAllDestinationCodes
                                GET
returnAction                       GET;
```

;;

CONDITIONAL PACKAGES

destinationTypePackage

PRESENT IF "destination type is required to unambiguously identify the destinationCode";

REGISTERED AS {q823ObjectClass 25};

11.1.26 tmCircuitEndPointSubgroup

tmCircuitEndPointSubgroup MANAGED OBJECT CLASS
 DERIVED FROM "ITU-T Rec. M.3100":circuitEndPointSubgroup;
 CHARACTERIZED BY

tmCircuitEndPointSubgroupPackage PACKAGE

BEHAVIOUR

tmCircuitEndPointBehaviour BEHAVIOUR

DEFINED AS

"The tmCircuitEndPointSubgroup is a subclass of M.3100 circuitEndPointSubgroup. It is used for performance monitoring and controls for traffic management purposes.";;

ATTRIBUTES

tmSurveillance GET-REPLACE

;;

;

REGISTERED AS {q823ObjectClass 26};

11.1.27 tmCircuitEndPointSubgroupCurrentData

tmCircuitEndPointSubgroupCurrentData MANAGED OBJECT CLASS
 DERIVED FROM circuitEndPointSubgroupCurrentData;
 CHARACTERIZED BY

**"ITU-T Rec. X.739:1993":periodSynchronizationPackage,
tmCircuitEndPointSubgroupCurrentDataPackage PACKAGE
BEHAVIOUR**

tmCircuitEndPointSubgroupCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic management circuit subgroup current data object class is a subclass of the circuitEndPointSubgroupCurrentData object class.

It is used for monitoring circuit subgroup related performance data as defined in Recommendation E.502 in the traffic management context. This object or its subclasses are instantiated for traffic management purposes.

A tmCircuitEndPointSubgroupCurrentData object instance shall generate only one instance of tmCircuitEndPointSubgroupHistoryData.

All tmCircuitEndPointSubgroupCurrentData object instances within the same managed element shall have the same granularity period.

In order to synchronize the granularityPeriod to the next integral time period after the currentData subclasses are instantiated, it is recommended that the value of periodSynchronizationTime (attribute of periodSynchronizationPackage) shall be set to twelve midnight (according to exchange time).";

ATTRIBUTES

"ITU-T Rec. Q.822:1994":historyRetention

PERMITTED VALUES Q823-TM-ASN1Module.PermittedHistoryRetention

GET

::

;
REGISTERED AS {q823ObjectClass 27};

11.1.28 tmExchangeCurrentData

tmExchangeCurrentData MANAGED OBJECT CLASS

DERIVED FROM exchangeCurrentData;

CHARACTERIZED BY

"ITU-T Rec. X.739:1993":periodSynchronizationPackage,

tmExchangeCurrentDataPackage PACKAGE

BEHAVIOUR

tmExchangeCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic management exchange performance current data object class is a subclass of the exchangeCurrentData object class.

It is used for monitoring exchange related performance data as defined in Recommendation E.502 in the traffic management context. This object class or its subclasses are instantiated for traffic management purposes.

A tmExchangeCurrentData object instance shall generate only one instance of tmExchangePerformanceHistoryData.

In order to synchronize the granularityPeriod to the next integral time period after the currentData subclasses are instantiated, it is recommended that the value of periodSynchronizationTime (attribute of periodSynchronizationPackage) shall be set to twelve midnight (according to exchange time).";

ATTRIBUTES

"ITU-T Rec. Q.822:1994":historyRetention

PERMITTED VALUES Q823-TM-ASN1Module.PermittedHistoryRetention

GET

::

;
REGISTERED AS {q823ObjectClass 28};

11.1.29 tmObservedDestinationCurrentData

tmObservedDestinationCurrentData MANAGED OBJECT CLASS

DERIVED FROM observedDestinationCurrentData;

CHARACTERIZED BY

"ITU-T Rec. X.739:1993":periodSynchronizationPackage,

tmObservedDestinationCurrentDataPackage PACKAGE

BEHAVIOUR

tmObservedDestinationCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic management destination current data object class is a subclass of the observedDestinationCurrentData object class.

It is used for monitoring observed destination related performance data as defined in Recommendation E.502 in the traffic management context. This object or its subclasses are instantiated for traffic management purposes.

A tmObservedDestinationCurrentData object instance shall generate only one instance of tmObservedDestinationHistoryData.

All tmDestinationCurrentData object instances within the same managed element shall have the same granularity period.

In order to synchronize the granularityPeriod to the next integral time period after the currentData subclasses are instantiated, it is recommended that the value of periodSynchronizationTime (attribute of periodSynchronizationPackage) shall be set to twelve midnight (according to exchange time).";

ATTRIBUTES

"ITU-T Rec. Q.822:1994":historyRetention

PERMITTED VALUES Q823-TM-ASN1Module.PermittedHistoryRetention

GET

::

;

REGISTERED AS {q823ObjectClass 29};

11.1.30 tmTrafficControlCurrentData

tmTrafficControlCurrentData MANAGED OBJECT CLASS

DERIVED FROM trafficControlCurrentData;

CHARACTERIZED BY

"ITU-T Rec. X.739:1993":periodSynchronizationPackage,

tmTrafficControlCurrentDataPackage PACKAGE

BEHAVIOUR

tmTrafficControlCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic management traffic control current data is a subclass of the trafficControlCurrentData object. It is used for monitoring the effectiveness of a traffic control as defined in Recommendation E.502 in the traffic management context. This object class or its subclasses are instantiated for traffic management purposes.

A tmTrafficControlCurrentData object instance shall generate only one instance of tmTrafficControlHistoryData.

All tmTrafficControlCurrentData object instances within the same managed element shall have the same granularity period.

In order to synchronize the granularityPeriod to the next integral time period after the currentData subclasses are instantiated, it is recommended that the value of periodSynchronizationTime (attribute of periodSynchronizationPackage) shall be set to twelve midnight (according to exchange time).";

ATTRIBUTES

"ITU-T Rec. Q.822:1994":historyRetention

PERMITTED VALUES Q823-TM-ASN1Module.PermittedHistoryRetention

GET

::

;

REGISTERED AS {q823ObjectClass 30};

11.1.31 trafficControl

trafficControl MANAGED OBJECT CLASS

DERIVED FROM "CCITT Rec. X.721:1992":top;

CHARACTERIZED BY

"ITU-T Rec. M.3100":objectManagementNotificationsPackage,

trafficControlPackage PACKAGE

BEHAVIOUR

trafficControlBehaviour BEHAVIOUR

DEFINED AS

"The Traffic Control object class is a superclass for all object classes representing traffic controls as defined in Recommendation E.412. This superclass is not instantiated.

In case of several controls of the same type are instantiated on the same managed resource, a mechanism (referred to as key) is described in each traffic control subclass to select an instance of the control. The control with the more specific criteria has precedence.";;

ATTRIBUTES

trafficControlId GET,
tmSurveillance GET-REPLACE;
;;

CONDITIONAL PACKAGES

creatorPackage
PRESENT IF "an instance supports it";

REGISTERED AS {q823ObjectClass 31};

11.1.32 trafficControlCurrentData

trafficControlCurrentData MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. Q.822:1994":currentData;

CHARACTERIZED BY

trafficControlCurrentDataPackage PACKAGE

BEHAVIOUR

trafficControlCurrentDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic control current data object class is a subclass of the currentData object class defined in Recommendation Q.822.

It is used for monitoring the effectiveness of a traffic control as defined in Recommendation E.502.

The trafficControlCurrentData is to be provided for each control instance. If there is a need to provide trafficControlCurrentData by control object class, or how a control impacts a managed entity, it can be aggregated at the OS from the corresponding control instance of traffic ControlCurrentData.

For traffic management purposes, this object class shall not be instantiated, but its subclass defined in this Recommendation or any subclass derived from that shall be instantiated.";;

ATTRIBUTES

callsAffectedByTrafficControl
INITIAL VALUE Q823-TM-ASN1Module.initialCount;
;;

CONDITIONAL PACKAGES

callsOfferedPackage
PRESENT IF "an instance supports it";

REGISTERED AS {q823ObjectClass 32};

11.1.33 trafficControlHistoryData

trafficControlHistoryData MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec. Q.822:1994":historyData;

CHARACTERIZED BY

trafficControlHistoryDataPackage PACKAGE

BEHAVIOUR

trafficControlHistoryDataBehaviour BEHAVIOUR

DEFINED AS

"The traffic control history data object class is a subclass of the historyData object class defined in ITU-T Recommendation Q.822.

The performance monitoring attributes are same as those in the corresponding currentData.";;

ATTRIBUTES

callsAffectedByTrafficControl GET;
;;

CONDITIONAL PACKAGES

callsOfferedHistoryPackage
PRESENT IF "this package is present if it is present in the corresponding instance of currentData";

REGISTERED AS {q823ObjectClass 33};

11.2 Definición de vinculación de nombres

La vinculación de nombres historyData-currentData se define en la Recomendación Q.822. Se utiliza para las subclases currentData e historyData definidas en esta Recomendación.

accAffectedTraffic-managedElement NAME BINDING

```
SUBORDINATE OBJECT CLASS accAffectedTraffic;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE accAffectedTrafficId;
BEHAVIOUR
accAffectedTrafficContainmentBehaviour      BEHAVIOUR
DEFINED AS "the delete operation will fail if one or more instance of acc points to this instance.";;
CREATE WITH-REFERENCE-OBJECT,
      WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE;
REGISTERED AS {q823NameBinding 1};
```

accTrigger-managedElement NAME BINDING

```
SUBORDINATE OBJECT CLASS accTrigger AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE accTriggerId;
REGISTERED AS {q823NameBinding 2};
```

adcTrigger-managedElement NAME BINDING

```
SUBORDINATE OBJECT CLASS adcTrigger AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE adcTriggerId;
CREATE WITH-REFERENCE-OBJECT,
      WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE;
REGISTERED AS {q823NameBinding 3};
```

autoHTRDestination-managedElement NAME BINDING

```
SUBORDINATE OBJECT CLASS htrDestination AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100": managedElement AND SUBCLASSES;
WITH ATTRIBUTE htrDestinationId;
BEHAVIOUR
autoHTRDestinationBehaviour      BEHAVIOUR
DEFINED AS "This name binding is used when an instance of htrDestination or its subclasses is
      created by the agent after determining by a local means that a destination is hard-to-reach.";;
REGISTERED AS {q823NameBinding 10};
```

circuitEndPointSubgroupHistoryData-tmCircuitEndPointSubgroup NAME BINDING

```
SUBORDINATE OBJECT CLASS circuitEndPointSubgroupHistoryData AND
SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS tmCircuitEndPointSubgroup AND
SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec. Q.822":historyDataId;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 4};
```

congestionLevelIndication-managedElement NAME BINDING

```
SUBORDINATE OBJECT CLASS congestionLevelIndication AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
```

WITH ATTRIBUTE congestionLevelIndicationId;
BEHAVIOUR
congestionLevelIndicationToMEBehaviour BEHAVIOUR
DEFINED AS "this object class is inherently created by NE";
REGISTERED AS {q823NameBinding 5};

dccGroup-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS dccGroup AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE dccGroupId;
BEHAVIOUR
dccGroupContainmentBehaviour BEHAVIOUR
DEFINED AS
"delete operation will fail if one or more instances of destinationCodeControl points to this object instance.";
CREATE WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE;
REGISTERED AS {q823NameBinding 6};

exchangeHistoryData-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS exchangeHistoryData AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec. Q.822:1994":historyDataId;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 8};

htrDestination-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS htrDestination AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE htrDestinationId;
BEHAVIOUR
htrDestinationToMEBehaviour BEHAVIOUR
DEFINED AS "this name binding is used when the object is explicitly created by the manager";
CREATE WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE;
REGISTERED AS {q823NameBinding 9};

observedDestination-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS observedDestination AND SUBCLASSES ;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE observedDestinationId;
CREATE WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 11};

observedDestinationHistoryData-observedDestination NAME BINDING
SUBORDINATE OBJECT CLASS observedDestinationHistoryData AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS observedDestination AND SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec.Q.822:1994":historyDataId;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 12};

scrAffectedTraffic-managedElement NAME BINDING
 SUBORDINATE OBJECT CLASS scrAffectedTraffic AND SUBCLASSES;
 NAMED BY
 SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
 WITH ATTRIBUTE scrAffectedTrafficId;
 BEHAVIOUR
 scrAffectedContainmentBehaviour BEHAVIOUR
 DEFINED AS "delete operation will fail if one or more instances of scr points to this object instance.";;
 CREATE WITH-REFERENCE-OBJECT,
 WITH-AUTOMATIC-INSTANCE-NAMING;
 DELETE DELETES-CONTAINED-OBJECTS;
 REGISTERED AS {q823NameBinding 13};

simpleScanner-managedElement NAME BINDING
 SUBORDINATE OBJECT CLASS "ITU-T Rec. X.738:1993":simpleScanner AND SUBCLASSES;
 NAMED BY
 SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":
 managedElement AND SUBCLASSES;
 WITH ATTRIBUTE "ITU-T Rec. X.739:1993":scannerId;
 CREATE WITH-REFERENCE-OBJECT,
 WITH-AUTOMATIC-INSTANCE-NAMING;
 DELETE ONLY-IF-NO-CONTAINED-OBJECTS;
 REGISTERED AS {q823NameBinding 14};

stateIndicator-managedElement NAME BINDING
 SUBORDINATE OBJECT CLASS stateIndicator AND SUBCLASSES;
 NAMED BY
 SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
 WITH ATTRIBUTE stateIndicatorId;
 BEHAVIOUR
 stateIndicatorContainmentBehaviour BEHAVIOUR
 DEFINED AS "this object class is inherently created by the NE.";;
 REGISTERED AS {q823NameBinding 15};

tmCircuitEndPointSubgroupCurrentData-tmCircuitEndPointSubgroup NAME BINDING
 SUBORDINATE OBJECT CLASS tmCircuitEndPointSubgroupCurrentData AND
 SUBCLASSES;
 NAMED BY
 SUPERIOR OBJECT CLASS tmCircuitEndPointSubgroup AND SUBCLASSES;
 WITH ATTRIBUTE "ITU-T Rec. X.739:1993":scannerId;
 BEHAVIOUR
 tmCircuitEndPointSubgroupCurrentDataContainmentBehaviour BEHAVIOUR
 DEFINED AS "instance of tmCircuitEndPointSubgroupCurrentData will be automatically instantiated
 when the tmSurveillance of the superior object class, tmCircuitEndPointSubgroup is set to TRUE. Instance
 of tmCircuitEndPointSubgroupCurrentData will be automatically deleted when the tmSurveillance
 attribute of the superior object class, tmCircuitEndPointSubgroup is set to FALSE. Instances of this class
 may also be inherently created by the NE.";;
 REGISTERED AS {q823NameBinding 16};

-- NOTE – the value of currentData attributes are determined by the NE using local means.

tmExchangeCurrentData-managedElement NAME BINDING
 SUBORDINATE OBJECT CLASS tmExchangeCurrentData AND SUBCLASSES;
 NAMED BY
 SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
 WITH ATTRIBUTE "ITU-T Rec. X.739:1993":scannerId;
 BEHAVIOUR
 tmExchangeCurrentDataContainmentBehaviour BEHAVIOUR
 DEFINED AS "only one instance of tmExchangeCurrentData is allowed within managedElement Instance
 of this class is inherently created by the NE";;

CREATE
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 17};

tmObservedDestinationCurrentData-observedDestination NAME BINDING
SUBORDINATE OBJECT CLASS tmObservedDestinationCurrentData AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS observedDestination AND SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec. X.739:1993":scannerId;
BEHAVIOUR
tmObservedDestinationCurrentDataContainmentBehaviour BEHAVIOUR
DEFINED AS "instance of tmObservedDestinationCurrentData will be automatically instantiated when the tmSurveillance of the superior object class, observedDestination is set to TRUE. Instance of tmObservedDestinationCurrentData will be automatically deleted when the tmSurveillance attribute of the superior object class, observedDestination is set to FALSE. Instances of this class may also be inherently created by the NE";;
REGISTERED AS {q823NameBinding 18};

-- NOTE – the value of currentData attributes are determined by the NE using local means.

tmTrafficControlCurrentData-trafficControl NAME BINDING
SUBORDINATE OBJECT CLASS tmTrafficControlCurrentData AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS trafficControl AND SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec. X.739:1993":scannerId;
BEHAVIOUR
tmTrafficControlCurrentDataContainmentBehaviour BEHAVIOUR
DEFINED AS "instance of tmTrafficControlCurrentData will be automatically instantiated when the tmSurveillance of the superior object class, trafficControl is set to TRUE. Instance of tmTrafficControlCurrentData will be automatically deleted when the tmSurveillance attribute of the superior object class, trafficControl is set to FALSE. Instances of this class may also be inherently created by the NE ";;
REGISTERED AS {q823NameBinding 19};

-- NOTE – the value of currentData attributes are determined by the NE using local means.

trafficControl-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS trafficControl AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE trafficControlId;
CREATE WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 20};

trafficControlHistoryData-trafficControl NAME BINDING
SUBORDINATE OBJECT CLASS trafficControlHistoryData AND SUBCLASSES;
NAMED BY
SUPERIOR OBJECT CLASS trafficControl AND SUBCLASSES;
WITH ATTRIBUTE "ITU-T Rec. Q.822:1994":historyDataId;
DELETE DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q823NameBinding 21};

11.3 Definición de lotes

adcAsynchronousTimerPackage PACKAGE
BEHAVIOUR
adcAsynchronousTimerBehaviour BEHAVIOUR

DEFINED AS

"identifies the asynchronous timer specifying a time value. The timer is set when the call attempt is allowed and no further call attempts are allowed until the timer expires.";;

ATTRIBUTES

asynchronousTimer GET;

REGISTERED AS {q823Package 1};

adcContinuousTimerPackage PACKAGE**BEHAVIOUR**

adcContinuousTimerBehaviour BEHAVIOUR

DEFINED AS

"identifies the continuous timer which includes the number of calls and a time. Once the number of call attempts has been handled with a timer cycle no further attempts are allowed until the time expires (e.g. 5 calls in 60 seconds.)";;

ATTRIBUTES

continuousTimer GET;

REGISTERED AS {q823Package 2};

adcLeakyBucketPackage PACKAGE**BEHAVIOUR**

adcLeakyBucketTimerBehaviour BEHAVIOUR

DEFINED AS

"identifies the leaky bucket which includes the bucket size (maximum allowed counter value) and the decrement per time unit. If the counter exceeds the defined maximum size, the call attempt is cancelled. If the count value is less than the maximum size, the call attempt is allowed and the counter is incremented. The counter is decremented at defined intervals making it possible for new calls to be accepted.";;

ATTRIBUTES

leakyBucket GET;

REGISTERED AS {q823Package 3};

adcPercentagePackage PACKAGE**BEHAVIOUR**

adcPercentageBehaviour BEHAVIOUR

DEFINED AS

"identifies the strength of the control in percentages. It is expected that only one of the choices of percentage for the strength will be supported.";;

ATTRIBUTES

percentage GET;

REGISTERED AS {q823Package 4};

answeredIncomingSeizuresPackage PACKAGE**BEHAVIOUR**

answeredIncomingSeizuresPackageBehaviour BEHAVIOUR

DEFINED AS

"Identifies the number of incoming seizures where an answer signal was transmitted back to the preceding exchange.";;

ATTRIBUTES

answeredIncomingSeizures INITIAL VALUE Q823-TM-ASN1Module.initialCount;

REGISTERED AS {q823Package 5};

answeredIncomingSeizureHistoryPackage PACKAGE**BEHAVIOUR**

answeredIncomingSeizureHistoryPackageBehaviour BEHAVIOUR

DEFINED AS

"Identifies the number of incoming seizures where an answer signal was transmitted back to the preceding exchange.";;

ATTRIBUTES

answeredIncomingSeizures GET;

REGISTERED AS {q823Package 6};

assocOwnerDccGroupPackage PACKAGE

BEHAVIOUR
 assocOwnerDccGroupBehaviour **BEHAVIOUR**
DEFINED AS
 "points to dccGroup instance.";;
ATTRIBUTES
 assocOwnerDccGroup **GET-REPLACE**;
REGISTERED AS {q823Package 7};

asynchronousTimerPackage PACKAGE
BEHAVIOUR
 asynchronousTimerPackageBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the asynchronous timer specifying a time value. The timer is set when the call attempt is allowed and no further call attempts are allowed until the timer expires.";;
ATTRIBUTES
 asynchronousTimer **GET-REPLACE**;
REGISTERED AS {q823Package 8};

callsAffectedByDccPackage PACKAGE
BEHAVIOUR
 callsAffectedByDccPackageBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the number of calls to the observed destination which have been affected by the destination control function, by type of control.";;
ATTRIBUTES
 callsAffectedByDcc **INITIAL VALUE** Q823-TM-ASN1Module.initialCount;
REGISTERED AS {q823Package 9};

callsAffectedByDccHistoryPackage PACKAGE
BEHAVIOUR
 callsAffectedByDccHistoryBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the number of calls to the observed destination affected by the destination controls.";;
ATTRIBUTES
 callsAffectedByDcc **GET**;
REGISTERED AS {q823Package 10};

callsOfferedPackage PACKAGE
BEHAVIOUR
 callsOfferedBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the number of calls which were offered to the traffic control instance.";;
ATTRIBUTES
 callsOfferedToTrafficControl **INITIAL VALUE** Q823-TM-ASN1Module.initialCount;
REGISTERED AS {q823Package 11};

callsOfferedHistoryPackage PACKAGE
BEHAVIOUR
 callsOfferedHistoryBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the number of calls which were offered to the traffic control instance";;
ATTRIBUTES
 callsOfferedToTrafficControl **GET**;
REGISTERED AS {q823Package 12};

continuousTimerPackage PACKAGE
BEHAVIOUR
 continuousTimerPackageBehaviour **BEHAVIOUR**

DEFINED AS

"identifies the continuous timer which includes the number of calls and a time. Once the number of call attempts has been handled with a timer cycle no further attempts are allowed until the time expires (e.g. 5 calls in 60 seconds.)";

ATTRIBUTES

continuousTimer GET-REPLACE;

REGISTERED AS {q823Package 13};

creatorPackage PACKAGE

BEHAVIOUR

creatorBehaviour BEHAVIOUR

DEFINED AS

"Identifies the creator of the object instance. The value of this attribute will be set by the agent at creation time. The means for determining the creatorIdentity is a local matter.";

ATTRIBUTES

creatorIdentity GET;

REGISTERED AS {q823Package 14};

destinationCodePackage PACKAGE

BEHAVIOUR

destinationCodePackageBehaviour BEHAVIOUR

DEFINED AS

"identifies the destination.";

ATTRIBUTES

destinationCode GET;

REGISTERED AS {q823Package 15};

destinationTypePackage PACKAGE

BEHAVIOUR

destinationTypePackageBehaviour BEHAVIOUR

DEFINED AS

"identifies the type of destination. It is either the nature of address in a seven bit string according to CCITT Recommendation Q.763, or the type of destination as an enumerated list.";

ATTRIBUTES

destinationType GET;

REGISTERED AS {q823Package 16};

leakyBucketPackage PACKAGE

BEHAVIOUR

leakyBucketTimerBehaviour BEHAVIOUR

DEFINED AS

"identifies the leaky bucket which includes the bucket size (maximum allowed counter value) and the decrement per time unit. If the counter exceeds the defined maximum size, the call attempt is cancelled. If the count value is less than or equal to the maximum size, the call attempt is allowed and the counter is incremented. The counter is decremented at defined intervals making it possible for new calls to be accepted.";

ATTRIBUTES

leakyBucket GET-REPLACE;

REGISTERED AS {q823Package 17};

level2ResponseCategoriesPackage PACKAGE

BEHAVIOUR

level2ResponseCategoriesPackageBehaviour BEHAVIOUR

DEFINED AS

"Identifies in a sequence the strength for a combination of routing aspect and/or origination aspect and/or destination aspect and/or additional criteria for threshold level2.";

ATTRIBUTES

level2ResponseCategories GET-REPLACE;

REGISTERED AS {q823Package 18};

percentagePackage PACKAGE

BEHAVIOUR

percentagePackageBehaviour BEHAVIOUR

DEFINED AS

"identifies the strength of the control in percentages. It is expected that only one of the choices of percentage for the strength will be supported.";;

ATTRIBUTES

percentage GET-REPLACE;

REGISTERED AS {q823Package 19};

stateIndicatorActionPackage PACKAGE

BEHAVIOUR

stateIndicatorActionPackageBehaviour BEHAVIOUR

DEFINED AS "to trigger the emission of the ACTION REPLY containing bit map showing the condition of the NE during the granularityPeriod.";;

ACTIONS

stateIndicatorAction;

REGISTERED AS {q823Package 20};

stateIndicatorNotificationPackage PACKAGE

BEHAVIOUR

stateIndicatorNotificationPackageBehaviour BEHAVIOUR

DEFINED AS "This notification is automatically emitted at the end of granularityPeriod, and provides a bit map showing the condition of the NE during the granularityPeriod.";;

NOTIFICATIONS

stateIndicatorNotification;

REGISTERED AS {q823Package 21};

tmCircuitEndPointSubgroupListPackage PACKAGE

BEHAVIOUR

tmCircuitEndPointSubgroupListPackageBehaviour BEHAVIOUR

DEFINED AS

"Identifies list of circuit subgroups";;

ATTRIBUTES

tmCircuitEndPointSubgroupList

GET-REPLACE

ADD-REMOVE;

REGISTERED AS {q823Package 22};

triggerThresholdPackage PACKAGE

BEHAVIOUR

triggerThresholdPackageBehaviour BEHAVIOUR

DEFINED AS

"identifies the threshold the trigger threshold value, if crossed will instantiate the adc control.";;

ATTRIBUTES

triggerThreshold GET;

REGISTERED AS {q823Package 23};

11.4 Definición de comportamiento común

instancePointerAndNameBehaviour BEHAVIOUR

DEFINED AS "if the string choice for the syntax is used, matching of the substrings is permitted. If the number choice for the syntax is used, then matching on ordering is permitted.";

11.5 Definición de atributos

accAffectedTrafficId ATTRIBUTE

DERIVED FROM trafficManagementObjectRdn;

REGISTERED AS {q823Attribute 1};

accResponseCategories **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.ResponseCategories;
BEHAVIOUR
accResponseValueBehaviour **BEHAVIOUR**
DEFINED AS
"Identifies in a sequence the strength for combination of routing aspect, and/or origination aspect and/or destination aspect and/or additional criteria.";;
REGISTERED AS {q823Attribute 2};

activationThresholds **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.ActivationThresholds;
MATCHES FOR EQUALITY;
BEHAVIOUR
 activationThresholdBehaviour **BEHAVIOUR**
DEFINED AS "identifies the thresholds for activation of the control";
REGISTERED AS {q823Attribute 3};

accTriggerId **ATTRIBUTE**
 DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 4};

adcTriggerId **ATTRIBUTE**
 DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 5};

adcTriggerType **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.AdcTriggerType;
MATCHES FOR EQUALITY;
BEHAVIOUR
adcTriggerTypeBehaviour **BEHAVIOUR**
DEFINED AS
"Identifies whether the adc control object instance to be created is 'centralized' or 'decentralized' or 'both'.";;
REGISTERED AS {q823Attribute 6};

adcType **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.AdcType;
MATCHES FOR EQUALITY;
BEHAVIOUR
adcTypeBehaviour **BEHAVIOUR**
DEFINED AS
"Identifies whether the adc control object instance to be created is 'centralized' or 'decentralized'.";;
REGISTERED AS {q823Attribute 7};

answer **ATTRIBUTE**
 DERIVED FROM "CCITT Rec. X.721:1992":counter;
MATCHES FOR EQUALITY;
BEHAVIOUR
answerBehaviour **BEHAVIOUR**
DEFINED AS
"information sent in the backward direction indication that the call is answered. This term is defined in Annex A/E.410" ;;
REGISTERED AS {q823Attribute 8};

answeredIncomingSeizures **ATTRIBUTE**
 DERIVED FROM answer;
BEHAVIOUR
answeredIncomingSeizuresBehaviour **BEHAVIOUR**

DEFINED AS
 "identifies the number of incoming seizures where an answer signal was transmitted back to the preceding exchange.";;
REGISTERED AS {q823Attribute 9};

answeredOutgoingSeizures **ATTRIBUTE**
DERIVED FROM answer;
BEHAVIOUR
answeredOutgoingSeizuresBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the number of outgoing seizures where an answer signal was received.";;
REGISTERED AS {q823Attribute 10};

assocAccAffectedTraffic **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.AssocAccAffectedTraffic;
MATCHES FOR EQUALITY;
BEHAVIOUR
assocAccAffectedTrafficBehaviour **BEHAVIOUR**
DEFINED AS
 "points to the accAffectedTraffic instance for congestion level 1 and 2";;
REGISTERED AS {q823Attribute 66};

assocOwnerDccGroup **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.AssocOwnerDccGroup;
MATCHES FOR EQUALITY;
BEHAVIOUR
assocOwnerDccGroupPointerBehaviour **BEHAVIOUR**
DEFINED AS
 "points to dccGroup Object instance.";;
REGISTERED AS {q823Attribute 11};

assocScrAffectedTraffic **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.AssocScrAffectedTraffic;
MATCHES FOR EQUALITY;
BEHAVIOUR
assocScrAffectedTrafficBehaviour **BEHAVIOUR**
DEFINED AS
 "points to the scrAffectedTraffic instance for reservation level 1 and 2";;
REGISTERED AS {q823Attribute 53};

asynchronousTimer **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.Timer;
MATCHES FOR EQUALITY;
BEHAVIOUR
asynchronousTimerBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the asynchronous timer specifying a time value. The timer is set when the call attempt is allowed and no further call attempts are allowed until the timer expires. It is expected that only one of the timer choices will be supported.";;
REGISTERED AS {q823Attribute 12};

autoActivated **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.TrueFalse;
MATCHES FOR EQUALITY;
BEHAVIOUR
autoActivatedBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies whether a trigger for an automatic control is outstanding. When the value is 'TRUE' and the administrativeState is 'unlocked', the automatic control is active. The automatic control is deactivated for all other combinations of the values of the autoActivated and administrativeState. The value of this attribute is initialized by and maintained by the NE.";;

REGISTERED AS {q823Attribute 13};

bids **ATTRIBUTE**

DERIVED FROM "CCITT Rec. X.721:1992":counter;

MATCHES FOR EQUALITY;

BEHAVIOUR

bidsBehaviour BEHAVIOUR

DEFINED AS

"An attempt to obtain a circuit in a circuit subgroup or to a destination. A bid may be successful or unsuccessful in seizing a circuit in that circuit subgroup or to that destination. This term is defined in Annex A/E.410.";;

REGISTERED AS {q823Attribute 14};

callsAffectedByDcc **ATTRIBUTE**

DERIVED FROM "CCITT Rec. X.721:1992":counter;

MATCHES FOR EQUALITY;

BEHAVIOUR

callsAffectedByDccBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of calls to the observed destination affected by destination code control.";;

REGISTERED AS {q823Attribute 15};

callsAffectedByTrafficControl **ATTRIBUTE**

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

callsAffectedByTrafficControlBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of calls which are actually affected by traffic control instance.";;

REGISTERED AS {q823Attribute 16};

callsBlockedByLoadShedding **ATTRIBUTE**

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

callsBlockedByLoadSheddingBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of calls which cannot be handled due to application of an exchange internal overload protection mechanism.";;

REGISTERED AS {q823Attribute 17};

callsOfferedToTrafficControl **ATTRIBUTE**

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

callsOfferedToTrafficControlBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of calls which were offered to traffic control instance.";;

REGISTERED AS {q823Attribute 18};

c11ResponseCategories **ATTRIBUTE**

WITH **ATTRIBUTE**

SYNTAX

Q823-TM-

ASN1Module.ResponseCategories;

MATCHES FOR EQUALITY;

BEHAVIOUR

c11ResponseCategoriesBehaviour

BEHAVIOUR

DEFINED AS

"response categories for congestion level 1.";;

REGISTERED AS {q823Attribute 19};

c12ResponseCategories **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module. ResponseCategories;

MATCHES FOR EQUALITY;

BEHAVIOUR

c12ResponseCategoriesBehaviour

BEHAVIOUR

DEFINED AS
 "response categories for congestion level 2.";;
REGISTERED AS {q823Attribute 20};

congestionLevel **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.CongestionLevel;
MATCHES FOR EQUALITY;
BEHAVIOUR
 congestionLevelBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the present congestion situation in an exchange. It is expressed by the following
 Machine Congestion Levels:
 – MCL0: no exchange congestion
 – MCL1: Moderate exchange congestion; the exchange keeps working. Some calls may be
 rejected.
 – MCL2: Serious congestion level; the exchange is no longer able to handle all offered traffic.
 A large number of calls are rejected than MCL1.
 – MCL3: Unable to process call. While it is desirable, some NE may not be able to provide a
 level 3 indicator during catastrophic failures.
 The value of this Attribute will be automatically updated by the NE to reflect the present
 congestion situation. Modification of the value of this Attribute shall generate an
 AttributeValueChange Notification.";;
REGISTERED AS {q823Attribute 21};

congestionLevelIndicationId **ATTRIBUTE**
DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 22};

continuousTimer **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.ContinuousTimer;
MATCHES FOR EQUALITY;
BEHAVIOUR
 continuousTimerBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the continuous timer which includes the number of calls and a time. Once the number
 of call attempts has been handled within a timer cycle, no further attempts are allowed until the
 timer expires. e.g. 5 calls in 60 seconds. It is expected that only one of the timer choices will be
 supported.";;
REGISTERED AS {q823Attribute 23};

controlTmCircuitEndPointSubgroup **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.InstancePointerOrName;
MATCHES FOR EQUALITY, SUBSTRINGS, ORDERING;
BEHAVIOUR
 instancePointerAndNameBehaviour,
 controlTmCircuitEndPointSubgroupBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the controlled circuit subgroup. The SUBSTRINGS and ORDERING matching rules only
 apply if this Attribute is specified as a name."
 ;;
REGISTERED AS {q823Attribute 58};

creatorIdentity **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.CreatorIdentity;
MATCHES FOR EQUALITY;
BEHAVIOUR
 creatorIdentityBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the entity which created the object instance";;
REGISTERED AS {q823Attribute 24};

dccGroupId **ATTRIBUTE**
DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 25};

destinationAspect **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.DestinationAspect;
MATCHES FOR EQUALITY;
BEHAVIOUR
destinationAspectBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the destination aspects (e.g. HTR) for which this control is valid. If this Attribute has a NULL value, the traffic control is valid for all destination aspects.";;
REGISTERED AS {q823Attribute 26};

destinationCode **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.DestinationCode;
MATCHES FOR EQUALITY, SUBSTRINGS, ORDERING;
BEHAVIOUR
destinationCodeBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the country code, or/and area code, or/and exchange code or/and other location number to which object instance applies";;
REGISTERED AS {q823Attribute 27};

destinationType **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.DestinationType;
MATCHES FOR EQUALITY;
BEHAVIOUR
destinationTypeBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the type of destination. It is either the nature of address in a seven bit string according to Recommendation Q.763, or the type of destination as an enumerated list.";;
REGISTERED AS {q823Attribute 28};

dispositionOfCalls **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.DispositionOfCall;
MATCHES FOR EQUALITY;
BEHAVIOUR
dispositionOfCallBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies whether the calls will be skipped to the next available circuit subgroup (if the value is NULL) or will be cancelled (if the value is treatment)";;
REGISTERED AS {q823Attribute 29};

htrDestinationId **ATTRIBUTE**
DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 30};

incomingSeizures **ATTRIBUTE**
DERIVED FROM seizure;
BEHAVIOUR
incomingSeizuresBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the number of incoming seizures on the circuit subgroup.";;
REGISTERED AS {q823Attribute 31};

incomingTraffic **ATTRIBUTE**
DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
incomingTrafficBehaviour **BEHAVIOUR**

DEFINED AS

"identifies the number of incoming calls in the exchange for which reception of at least one digit has been acknowledged.";;

REGISTERED AS {q823Attribute 32};

incomingTrafficUsage ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.Integer;

BEHAVIOUR

incomingTrafficUsageBehaviour BEHAVIOUR

DEFINED AS

"identifies the incoming traffic carried in erlang sec. Typically, this is provided via the sampling method.";;

REGISTERED AS {q823Attribute 33};

internalTraffic ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

internalTrafficBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of internal calls (seizures) in the exchange.";;

REGISTERED AS {q823Attribute 34};

leakyBucket ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.LeakyBucket;

MATCHES FOR EQUALITY;

BEHAVIOUR

leakyBucketBehaviour BEHAVIOUR

DEFINED AS

"identifies the leaky bucket which includes the bucket size (maximum allowed counter value) and decrement per time unit. If the counter exceeds the defined maximum size, the call attempt is cancelled. If the counter is less than or equal the maximum size, the call attempt is allowed and the counter is incremented. The counter is decremented at defined intervals making it possible for new calls to be accepted. The NE will provide the bucketSize if it is not provided. It is expected that only one of the timer choices will be supported.";;

REGISTERED AS {q823Attribute 35};

level1ResponseCategories ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-

ASN1Module.ResponseCategories;

MATCHES FOR EQUALITY;

BEHAVIOUR

level1ResponseCategoriesBehaviour BEHAVIOUR

DEFINED AS

"response categories for reservation level 1";;

REGISTERED AS {q823Attribute 36};

level2ResponseCategories ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-

ASN1Module.ResponseCategories;

MATCHES FOR EQUALITY;

BEHAVIOUR

level2ResponseCategoriesBehaviour BEHAVIOUR

DEFINED AS

"response categories for reservation level 2";;

REGISTERED AS {q823Attribute 37};

newTmCircuitEndPointSubgroups ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.NewTmCircuitEndPointSubgroups;

MATCHES FOR EQUALITY;

BEHAVIOUR

newTmCircuitEndPointSubgroupsBehaviour BEHAVIOUR

DEFINED AS

"Identifies in sequence the circuit subgroups which have idle capacity to complete calls.";;

REGISTERED AS {q823Attribute 38};

noCircuitsAvailable ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

noCircuitsAvailableBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of bids resulting in unsuccessful call due to the fact that no free circuits leading to the observed destination was available; i.e. overflow resulting on the final circuit subgroup of that destination.";;

REGISTERED AS {q823Attribute 39};

numberOfAvailCircuits ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721:1992":gauge;

BEHAVIOUR

numberOfAvailCircuitsBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of circuits that can carry traffic including the ones currently carrying traffic. Whether this value is provided as snapshot or as a mean value is left to the implementation as due to the normally high frequency of changes to the circuits, both methods are equivalent.";;

REGISTERED AS {q823Attribute 40};

observedDestinationId ATTRIBUTE

DERIVED FROM trafficManagementObjectRdn;

REGISTERED AS {q823Attribute 41};

originationAspect ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.OriginationAspect;

MATCHES FOR EQUALITY;

BEHAVIOUR

instancePointerAndNameBehaviour,

originationAspectBehaviour BEHAVIOUR

DEFINED AS

"Identifies the origin and calling parties' category (according to Recommendation Q.763) for which the control is valid. If the value of this attribute is empty sequence, the traffic control is valid for all origination aspects.";;

REGISTERED AS {q823Attribute 42};

originatingTraffic ATTRIBUTE

DERIVED FROM "CCITT Rec. X.721:1992":counter;

BEHAVIOUR

originatingTrafficBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of originating calls (seizures) in the exchange for which at least one digit has been acknowledged.";;

REGISTERED AS {q823Attribute 43};

outgoingBids ATTRIBUTE

DERIVED FROM bids;

BEHAVIOUR

outgoingBidsBehaviour BEHAVIOUR

DEFINED AS

"identifies the number of successful or unsuccessful attempts to seize a circuit in the circuit subgroup";;

REGISTERED AS {q823Attribute 44};

outgoingSeizures ATTRIBUTE

DERIVED FROM seizure;

BEHAVIOUR

outgoingSeizuresBehaviour BEHAVIOUR

DEFINED AS
 "identifies the number of outgoing bids which succeeded in obtaining a circuit within a circuit subgroup. ";
REGISTERED AS {q823Attribute 45};

outgoingTraffic ATTRIBUTE
DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
outgoingTrafficBehaviour BEHAVIOUR
DEFINED AS
 "identifies the number of outgoing calls (seizures) from the exchange which has successfully seized a circuit.";;
REGISTERED AS {q823Attribute 46};

outgoingTrafficUsage ATTRIBUTE
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.Integer;
BEHAVIOUR
outgoingTrafficUsageBehaviour BEHAVIOUR
DEFINED AS
 "identifies the outgoing traffic carried in erlang sec. Typically, this is provided via the sampling method." ;
REGISTERED AS {q823Attribute 47};

overflow ATTRIBUTE
DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
overflowBehaviour BEHAVIOUR
DEFINED AS
 "identifies the number of outgoing bids overflowing from this circuit subgroup. It will not include calls affected by cancel rerouted overflow, tar from and cancel from controls.";;
REGISTERED AS {q823Attribute 48};

percentage ATTRIBUTE
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.Percentage;
MATCHES FOR EQUALITY;
BEHAVIOUR
percentageBehaviour BEHAVIOUR
DEFINED AS
 "identifies the percentage of calls that shall be affected as a result of control activation. It is expected that only one of the choices of percentage for the strength will be supported.";;
REGISTERED AS {q823Attribute 49};

returnAction ATTRIBUTE
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.ReturnAction;
MATCHES FOR EQUALITY;
BEHAVIOUR
returnActionBehaviour BEHAVIOUR
DEFINED AS
 "Identifies the disposition of overflowing traffic from the new CircuitEndPointSubgroups.";;
REGISTERED AS {q823Attribute 50};

routingAspect ATTRIBUTE
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.RoutingAspect;
MATCHES FOR EQUALITY;
BEHAVIOUR
routingAspectBehaviour BEHAVIOUR
DEFINED AS
 "Identifies the routing aspect (direct routed traffic or alternate routed traffic) for which this control is valid. If the Attribute value is NULL, the traffic control is applied all routing aspects.";;
REGISTERED AS {q823Attribute 51};

scrAffectedTrafficId **ATTRIBUTE**
 DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 52};

seizure **ATTRIBUTE**
 DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
seizureBehaviour **BEHAVIOUR**
DEFINED AS
 "a bid for a circuit in a circuit subgroup which succeeds in obtaining a circuit in that circuit subgroup.
 This term is defined in Annex A/E.410.";;
REGISTERED AS {q823Attribute 54};

stateIndicatorId **ATTRIBUTE**
 DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 55};

terminatingTraffic **ATTRIBUTE**
 DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
terminatingTrafficBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies the number of terminating calls (seizures to lines) in the exchange.";;
REGISTERED AS {q823Attribute 56};

timeExpiration **ATTRIBUTE**
 WITH ATTRIBUTE SYNTAX
 Q823-TM-ASN1Module.Timer;
 MATCHES FOR EQUALITY;
BEHAVIOUR
timeExpiringBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the time after which the control will automatically be deleted";;
REGISTERED AS {q823Attribute 57};

tmCircuitEndPointSubgroupList **ATTRIBUTE**
 WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.TmCircuitEndPointSubgroupList;
BEHAVIOUR
tmCircuitEndPointSubgroupListBehaviour **BEHAVIOUR**
DEFINED AS
 "Identifies the circuit subgroups for which this object instance applies. If the value is empty set, the
 object instance applies to all circuit subgroups.";;
REGISTERED AS {q823Attribute 59};

tmSurveillance **ATTRIBUTE**
 WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.TrueFalse;
 MATCHES FOR EQUALITY;
BEHAVIOUR
tmSurveillanceBehaviour **BEHAVIOUR**
DEFINED AS
 "identifies whether the object instance is being surveilled for traffic management purposes. Setting the
 value of this Attribute to TRUE will result in the creation of an instance of a subclass of currentData,
 named from this object instance and having a value of integer '1' for its scannerId attribute (naming
 attribute). If the creation of currentData object is not successful, the set operation will fail and the value
 of this attribute will remain unchanged. If the value of this attribute was already TRUE before the set
 operation, the operation will succeed but will have no effect on related object.
 Setting the value of this attribute to FALSE will result in the deletion of an instance of a subclass of
 currentData, named from this object instance and having a value of integer '1' for its scannerId attribute
 (naming attribute). If the deletion of currentData object is not successful, the set operation will fail and
 the value of this attribute will remain unchanged. If the value of this attribute was already FALSE before
 the set operation, the operation will succeed but will have no effect on related object.";;
REGISTERED AS {q823Attribute 60};

trafficControlId **ATTRIBUTE**
DERIVED FROM trafficManagementObjectRdn;
REGISTERED AS {q823Attribute 61};

trafficManagementObjectRdn **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.NameType;
MATCHES FOR EQUALITY;
BEHAVIOUR
trafficManagementObjectRdnBehaviour **BEHAVIOUR**
DEFINED AS
"It is the object identifier";
REGISTERED AS {q823Attribute 62};

transitTraffic **ATTRIBUTE**
DERIVED FROM "CCITT Rec. X.721:1992":counter;
BEHAVIOUR
transitTrafficBehaviour **BEHAVIOUR**
DEFINED AS
"identifies the number of transit calls (seizures) in the exchange.";;
REGISTERED AS {q823Attribute 63};

treatment **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.Treatment;
MATCHES FOR EQUALITY;
BEHAVIOUR
instancePointerAndNameBehaviour,
treatmentBehaviour **BEHAVIOUR**
DEFINED AS
"Identifies how the traffic flow impacted by the cancellation will be treated
(e.g. announcement).";
REGISTERED AS {q823Attribute 64};

triggerThreshold **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Q823-TM-ASN1Module.IntegerValue;
MATCHES FOR EQUALITY;
BEHAVIOUR
triggerThresholdBehaviour **BEHAVIOUR**
DEFINED AS
"Identifies the trigger threshold value that will instantiate an automatic control.";;
REGISTERED AS {q823Attribute 65};

11.6 Definición de acciones

stateIndicatorAction **ACTION**
BEHAVIOUR
stateIndicatorActionBehaviour **BEHAVIOUR**
DEFINED AS "to trigger the emission of the ACTION REPLY containing bit map showing the condition of the NE during the granularityPeriod. The bit position and their definitions are as follows:
bit-0: exchangeCongestionLevel1: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 1.
bit-1: exchangeCongestionLevel2: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 2.
bit-2: congestionLevel1Received: This bit is equal to 1 if during the granularityPeriod the NE receives CL1 Indicator from any adjacent exchanges.
bit-3: congestionLevel2Received: This bit is set equal to 1 if during the granularityPeriod when the NE received CL2 indicator from any adjacent exchanges.
bit-4: scrTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of SCR control.
bit-5: accTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of acc control.

bit-6: protectiveControlActive: This bit is equal to 1 if during the granularityPeriod one or more instances of protective controls (i.e. cancelTo, cancelFrom, skip, acc, scr, cancelRerouted) is present in the NE.

bit-7: expansiveControlActive: This bit is equal to 1 if during the granularityPeriod when one or more instances of expansive controls (i.e. tarTo, tarFrom) is present in the NE.

bit-8: destinationControlActive: This bit is equal to 1 during the granularityPeriod when one or more instances of destination controls (i.e. destinationCodeControl functions, adc) are present in the NE.

bit-9: htrDestinationActive: This bit is equal to 1 during the granularityPeriod when one or more instances of htrDestination are present in the NE.

bit-10: circuitEndPointSubgroupAddedOrDeleted: This bit is equal to 1 during the granularityPeriod when a circuit subgroup is added or deleted in the NE.

bit-11: accTransmissionInhibited: This bit is equal to 1 if during the granularityPeriod the administrativeState of accTrigger is set to locked.

bit-12: adcTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of adc control";;

MODE CONFIRMED;

WITH REPLY SYNTAX Q823-TM-ASN1Module.StateIndicatorBitMap;

REGISTERED AS {q823Action 1};

11.7 Definición de modificaciones

stateIndicatorNotification NOTIFICATION
BEHAVIOUR

stateIndicatorNotificationBehaviour BEHAVIOUR

DEFINED AS "This notification is automatically emitted at the end of granularityPeriod, and provides a bitmap showing the condition of the NE during the granularityPeriod. The bit position and their definitions are as follows:

bit-0: exchangeCongestionLevel1: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 1.

bit-1: exchangeCongestionLevel2: This bit is equal to 1 if during the granularityPeriod the NE experiences congestion level 2.

bit-2: congestionLevel1Received: This bit is equal to 1 if during the granularityPeriod the NE receives CL1 Indicator from any adjacent exchanges.

bit-3: congestionLevel2Receieved: This bit is set equal to 1 during the granularityPeriod when the NE receivedCL2 indicator from any adjacent exchanges.

bit-4: scrTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of SCR control.

bit-5: accTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of acc control.

bit-6: protectiveControlActive: This bit is equal to 1 if during the granularityPeriod one or more instances of protective controls (i.e. cancelTo, cancelFrom, skip, acc, scr, cancelRerouted) are present in the NE.

bit-7: expansiveControlActive: This bit is equal to 1 during the granularityPeriod when one or more instances of expansive controls (i.e. tarTo, tarFrom) are present in the NE.

bit-8: destinationControlActive: This bit is equal to 1 during the granularityPeriod when one or more instances of destination controls (i.e. destinationCodeControl functions, adc) are present in the NE.

bit-9: htrDestinationActive: This bit is equal to 1 during the granularityPeriod when one or more instances of htrDestination are present in the NE.

bit-10: circuitEndPointSubgroupAddedOrDeleted: This bit is equal to 1 during the granularityPeriod when a circuit subgroup is added or deleted in the NE.

bit-11: accTransmissionInhibited: This bit is equal to 1 if during the granularityPeriod the administrativeState of accTrigger is set to locked.

bit-12: adcTriggered: This bit is equal to 1 if during the granularityPeriod the administrativeState is unlocked and autoActivated is TRUE for at least one instance of adc control";;

WITH INFORMATION SYNTAX Q823-TM-ASN1Module.StateIndicatorBitMap;

REGISTERED AS {q823Notification 1};

11.8 Módulo tipos definidos de ASN.1

-- Extensibility rules

-- As per ISO 8824-DAM on extensibility rules, the productions that are of extensible types are to

-- be indicated by including three (3) ellipses (...) in their type descriptions.

Q823 – TM-ASN1Module {ccitt(0) recommendation(0) q(17) q823(823) asn1Module(2) q823ASN1Module(0)}

DEFINITIONS IMPLICIT TAGS ::=

BEGIN

-- EXPORTS everything

IMPORTS

Attribute,

ObjectInstance

FROM

CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)}

AdministrativeState,

Count,

ObservedValue

FROM

Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1}

HistoryRetention

FROM

Q822-PM-ASN1Module {ccitt(0) recommendation(0) q(17) q822(822) asn1Module(2) q822ASN1Module(0)}

NameType,

FROM

ASN1DefinedTypesModule {ccitt recommendation m(13) gnm(3100) informationModel(0) asn1Modules(2) asn1DefinedTypesModule(0)};

q823-InformationModel OBJECT IDENTIFIER ::= {ccitt(0) recommendation(0) q(17) q823(823) informationModel(0)}

q823ObjectClass OBJECT IDENTIFIER ::= {q823-InformationModel managedObjectClass(3)}

q823Package OBJECT IDENTIFIER ::= {q823-InformationModel package(4)}

q823NameBinding OBJECT IDENTIFIER ::= {q823-InformationModel nameBinding(5)}

q823Attribute OBJECT IDENTIFIER ::= {q823-InformationModel attribute(6)}

q823Action OBJECT IDENTIFIER ::= {q823-InformationModel action(7)}

q823Notification OBJECT IDENTIFIER ::= {q823-InformationModel notification(8)}

--value assignments for origin aspect extension

extendOrigin OBJECT IDENTIFIER ::= {q823-InformationModel standardSpecificExtension(0) extendOrigin(0)}

originated Origin ::= originExtension : {extendOrigin 1}

transit Origin ::= originExtension : {extendOrigin 2}

inboundTerminating Origin ::= originExtension : {extendOrigin 3}

--value assignment for destination aspect extension

**extendDestination OBJECT IDENTIFIER ::= {q823-InformationModel standardSpecificExtension(0)
extendDestination(1)}**

hardToReach DestinationAspect ::= destinationExtension : {extendDestination 1}

nonHardToReach DestinationAspect ::= destinationExtension : {extendDestination 2}

-- default value definition

defaultAdministrativeState AdministrativeState ::= unlocked

defaultAllTmCircuitEndPointSubgroups TmCircuitEndPointSubgroupList ::= {}

defaultAllDestinationCodes DestinationCode ::= ""

defaultDestinationAspect DestinationAspect ::= definedDestinationAspect : null

defaultOriginAspect OriginationAspect ::= {}

defaultRoutingAspects RoutingAspect ::= null

--initial value definitions

initialCount Count ::= 0

initialGauge ObservedValue ::= integer: 0

initialInteger INTEGER ::= 0

-- permitted value definitions

PermittedState ::= AdministrativeState(locked | unlocked)

PermittedHistoryRetention ::= HistoryRetention(1)

PermittedTarFromReturnAction ::= ReturnAction (WITH COMPONENTS {return, cancelTreatment})

--supporting productions

**ActivationThresholds ::= SEQUENCE {
 level1 [0] ThresholdLevel,
 level2 [1] ThresholdLevel OPTIONAL}**

**AdcTriggerType ::= ENUMERATED {
 centralized (0),
 decentralized (1),
 both (2) -- both centralized and decentralized --
}**

**AdcType ::= ENUMERATED {
 centralized (0),
 decentralized (1)}**

AssocAccAffectedTraffic ::= ObjectInstance -- points to a accAffectedTraffic

AssocOwnerDccGroup ::= ObjectInstance -- points to dccGroup object instance

AssocScrAffectedTraffic ::= ObjectInstance -- points to scrAffectedTraffic

Category ::= BIT STRING (SIZE(8)) -- value according to Recommendation Q.763

```
CongestionLevel ::= ENUMERATED {
    mcl0 (0),
    mcl1 (1),
    mcl2 (2),
    mcl3 (3)}
```

```
ContinuousTimer ::= SEQUENCE {
    calls [0] INTEGER,
    perTimeUnit [1] Timer
}
```

```
CreatorIdentity ::= CHOICE {
    sourceCls SourceClass,
    sourceName GraphicString
}
```

```
DefinedOrigin ::= ENUMERATED {
    originated (0),
    transit (1),
    inboundTerminating (2)
}
```

```
DestinationAspect ::= CHOICE {
    definedDestinationAspect DefinedDestinationAspect,
    destinationExtension OBJECT IDENTIFIER
}
```

```
DefinedDestinationAspect ::= ENUMERATED {
    null (0), --all destination aspects
    htr (1),
    nhtr (2)
}
```

```
DestinationCode ::= GraphicString(FROM("0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" | "A" | "B" | "C" | "D" | "E" | "F" | "#" | "*"))
```

```
DestinationType ::= CHOICE {
    natureOfAddress [0] NatureOfAddress,
    desttype [1] DestType,
    -- "...these ellipses defined in ASN.1 amendment are used here to indicate that this is --
    -- an extensible type and additional choices may be added in the future--
}
```

```
DestType ::= ENUMERATED {
    international (0),
    national (1),
    local (2),
    other (3)
}
```

```
DispositionOfCall ::= CHOICE {
    skip NULL,
    treatment Treatment
}
```

```
InstancePointerOrName ::= CHOICE {
    objectInstance [0] ObjectInstance,
    symbolic [1] NameType
}
```

```
Integer ::= INTEGER(0 .. MAX)
```


IntegerValue ::= INTEGER(0 .. 100)

LeakyBucket ::= SEQUENCE {
 bucketSize [0] **INTEGER OPTIONAL**, -- *the NE will define the bucketSize if not provided*
 calls [1] **INTEGER**,
 perTimeUnit [2] **Time (WITH COMPONENTS{scale1, scale2})**
}

NatureOfAddress ::= BIT STRING (SIZE(7)) -- *value according to Recommendation Q.763*

NewTmCircuitEndPointSubgroups ::= SEQUENCE OF InstancePointerOrName

Origin ::= CHOICE {
 definedOrigin [0] **DefinedOrigin**,
 namedOrigin [1] **InstancePointerOrName**, --*it points to or names an origin.*
 originExtension [2] **OBJECT IDENTIFIER**
}

OriginationAspect ::= SEQUENCE {
 callingPartyCategory [0] **Category OPTIONAL**,
 origin [1] **Origin OPTIONAL**
}

Percentage ::= CHOICE {
 granularity12p5 [0] **ENUMERATED {**
 perc0 (0), --0%--
 perc12p5 (2), --12.5%--
 perc25 (4), --25%--
 perc37p5 (6), --37.5%--
 perc50 (8), --50%--
 perc62p5 (10), --67.5%--
 perc75 (12), --75%--
 perc87p5 (13), --87.5%--
 perc100 (15) --100%--
 },
 granularity10 [1] **ENUMERATED {**
 perc0 (0) -- 0%--,
 perc10 (1) -- 10%--,
 perc20 (2) -- 20%--,
 perc30 (3) -- 30% --,
 perc40 (4) -- 40%--,
 perc50 (5) -- 50%--,
 perc60 (6) -- 60%--,
 perc70 (7) -- 70%--,
 perc80 (8) -- 80%--,
 perc90 (9) -- 90%--,
 perc100 (10) -- 100%--
 }
}

ResponseCategories ::= SET OF SEQUENCE {
 originationAspect [0] **OriginationAspect**,
 destinationAspect [1] **DestinationAspect**,
 routingAspect [2] **RoutingAspect**,
 additionalCriteria [3] **OBJECT IDENTIFIER OPTIONAL**,
 percentage [4] **Percentage**
}

ReturnAction ::= CHOICE {
 return [0] **NULL**,
 skip [1] **NULL**,
 cancelTreatment [2] **Treatment**
}

```

RoutingAspect ::= ENUMERATED {
    directRoutedTraffic    (0),
    alternateRoutedTraffic (1),
    null                   (2) -- direct and alternate routed traffic
}

SourceClass ::= ENUMERATED {
    tnmos    (0),
    otherNE  (1),
    thisNE   (2),
    other    (3)
}

StateIndicatorBitMap ::= BIT STRING {
    exchangeCongestionLevel1    (0),
    exchangeCongestionLevel2    (1),
    congestionLevel1Received     (2),
    congestionLevel2Received     (3),
    scrTriggered                 (4),
    accTriggered                 (5),
    protectiveControlActive      (6),
    expansiveControlActive       (7),
    destinationControlActive     (8),
    htrDestinationActive         (9),
    circuitEndPointSubgroupAddedOrDeleted(10),
    accTransmissionInhibited     (11),
    adcTriggered                 (12)
}

ThresholdLevel ::= CHOICE {
    percentageOfCircuits [0]  INTEGER(0 .. 100),
    noOfCircuits         [1]  INTEGER
}

TmCircuitEndPointSubgroupList ::= SET OF InstancePointerOrName

Treatment ::= CHOICE {
    other [0]  OBJECT IDENTIFIER,
    announcementNumber [1] INTEGER,
    congestionTone [2]  NULL
}

Timer ::= CHOICE {
    blockAllCalls [0]  NULL,
    scale1 [1]  ENUMERATED {
        sec0 (0) -- 0 sec --,
        sec0p1 (1) -- 0.1 sec --,
        sec0p25 (2) -- 0.25 sec --,
        sec0p5 (3) -- 0.5 sec --,
        sec1 (4) -- 1 sec --,
        sec2 (5) -- 2 sec --,
        sec5 (6) -- 5 sec --,
        sec10 (7) -- 10 sec --,
        sec15 (8) -- 15 sec --,
        sec30 (9) -- 30 sec --,
        sec60 (10) -- 60 sec --,
        sec120 (11) -- 120 sec --,
        sec300 (12) -- 300 sec --,
        sec600 (13) -- 600 sec--
    },
    scale2 [2]  INTEGER(0 .. 600000) --millisec. scale
}

TrueFalse ::= BOOLEAN
END

```

APÉNDICE I

Servicio de gestión de la RGT de gestión de tráfico

Este apéndice identifica las funciones de gestión de la red de tráfico que han de incorporarse en la Recomendación M.3200 que son tratadas en la presente Recomendación y las que caen fuera del alcance de esta Recomendación. Véase el cuadro I.1

Cuadro I.1/Q.823 – Flujos de información Q3

	NE	OS
Funciones de gestión de red		Tratadas en (Rec.)
1 Funciones de supervisión de situación		
Informe de la disponibilidad de servicio de los elementos de red	<-----	M.3100
Informe por demanda de la situación de los controles	<-----	Q.823
Informe de la situación de ocupado/reposo de un haz de circuitos	<-----	Indirectamente
Informe de la situación de congestión de la central	<-----	Q.823
Informe del recibo de una señal ACC	<-----	Q.823
Informe por demanda de la situación HTR de los destinos	<-----	Q.823
Adición/supresión manual de la situación HTR del destino	----->	Q.823
Informe de la situación de congestión de la red de señalización por canal común	<-----	Fuera de alcance
Informe del recibo de señales de gestión de señalización por canal común	<-----	Fuera de alcance
2 Funciones de supervisión de la calidad de funcionamiento		
Informe programado de datos/parámetros de haz de circuitos con carácter	<-----	Q.822, Q.823
Informe por demanda de datos/parámetros de haz de circuitos	<-----	Q.822, Q.823
Informe programado de mediciones de carga de central	<-----	Q.822, Q.823
Informe por demanda de mediciones de carga de central	<-----	Q.822, Q.823
Informe programado de mediciones de congestión de central	<-----	Q.822, Q.823
Informe por demanda de mediciones de congestión de central	<-----	Q.822, Q.823
Informe programado de medición de carga CCS	<-----	Fuera de alcance
Informe por demanda de medición de carga CCS	<-----	Fuera de alcance
Informe programado de medición de congestión CCS	<-----	Fuera de alcance
Informe por demanda de medición de congestión CCS	<-----	Fuera de alcance
Informe programado de datos sobre calidad de funcionamiento de los controles	<-----	Q.822, Q.823
Informe por demanda de datos sobre calidad de funcionamiento de los controles	<-----	Q.822, Q.823
3 Funciones de control		
Aplicación/modificación/supresión de control manual	----->	Q.823
Establecimiento/modificación/supresión de un control automático	----->	Q.823
Activación/desactivación de un control automático	----->	Q.823
Aplicación/modificación/supresión de un anuncio grabado especial	----->	Fuera de alcance

Cuadro I.1/Q.823 – Flujos de información Q3 (fin)

	NE	OS
Funciones de gestión de red		Tratadas en (Rec.)
4 Funciones administrativas		
Establecimiento/cambio/supresión de un calendario de mediciones	---->	X.746, Q.823
Establecimiento/cambio/actualización de una base de datos de gestión de red	---->	Fuera de alcance
Establecimiento/cambio/supresión de un umbral para informe de situación, informe de datos y determinación HTR	---->	Q.822, Q.823
Establecimiento/cambio/supresión de calendarios para información de situación y datos	---->	Q.822, Q.823
Comunicación por demanda de información sobre la tabla de encaminamientos		
a) qué subhaces de circuitos forman parte de qué destinos	<----	Fuera de alcance
b) qué destinos son encaminados por un determinado subhaz de circuitos	<----	Fuera de alcance

APÉNDICE II

Correlación entre los controles de tráfico de la Recomendación E.412 y las clases de objeto tratadas en esta Recomendación

El cuadro II.1 da la correlación entre los controles de tráfico que se definen en la Recomendación E.412, y la clase de objeto definida en esta Recomendación.

Cuadro II.1/Q.823

Control de tráfico E.412	Tratado por clase(s) de objeto(s) en Rec. Q.823	Observaciones
Bloqueo de código	destinationCodeControl, dccGroup	Ninguna
Espaciamiento de llamadas	destinationCodeControl, dccGroup	Ninguna
Cancelación de encaminamiento directo	cancelTo	Ninguna
Cancelación de encaminamiento alternativo hacia	cancelTo	Ninguna
Cancelación de encaminamiento alternativo desde	cancelFrom	Ninguna
Direccionalización de circuitos	–	Fuera de alcance
Rechazo/ocupación/bloqueo de circuito	–	Fuera de alcance
Salto de ruta	skip	Ninguna
Encaminamiento alternativo temporal por haz de circuitos/destino	tarTo, tarFrom	Ninguna
Cancelación de desbordamiento reencaminado	cancelRerouted	Ninguna
Control automático de congestión	acc, accTrigger, accAffectedTraffic	Ninguna
Reserva selectiva de circuitos	scr, scrAffectedTraffic	Ninguna
Control automático de destino	adcTrigger, adc	Ninguna

APÉNDICE III

Diagrama de transición de estados para controles automáticos y manuales

Véanse las figuras III.1 y III.2.

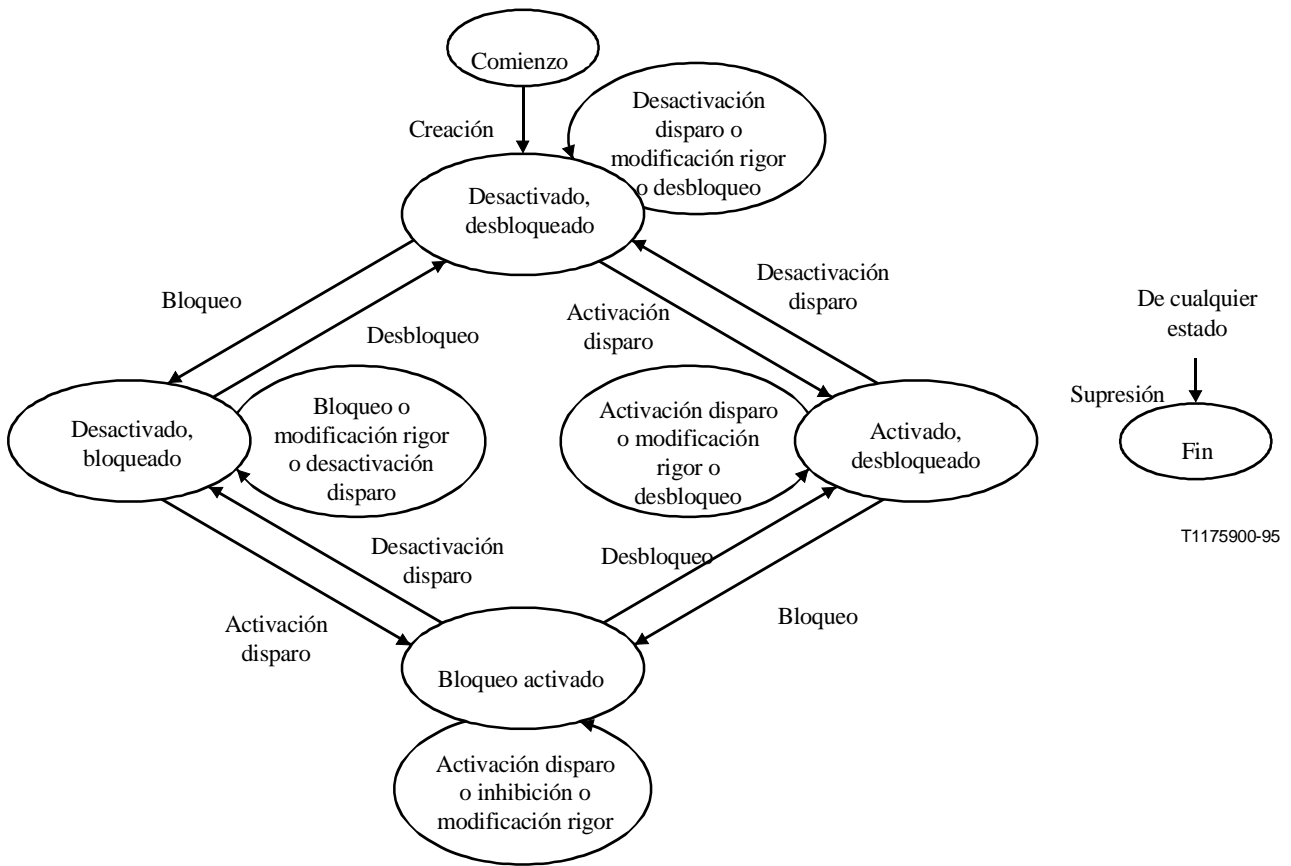


Figura III.1/Q.823 – Diagrama de transición de estados de control automático

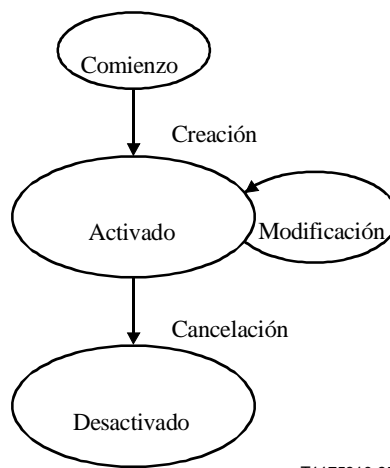


Figura III.2/Q.823 – Diagrama de transición de estados para controles automáticos y manuales

APÉNDICE IV

Diferenciación del tráfico con encaminamiento directo y con encaminamiento alternativo para los controles cancelación hacia y cancelación desde

A fin de utilizar eficazmente los recursos de red durante un periodo de gran actividad, el gestor de red hace uso de los controles cancelación hacia (Cancel-To) y cancelación desde (Cancel-From) en los subhaces de circuitos. Cuando aplica estos controles, el gestor de red debería tener la posibilidad de diferenciar y controlar selectivamente el tráfico con encaminamiento directo o el tráfico con encaminamiento alternativo en ambos tipos de controles. La figura IV.1 explica los diversos tipos de tráfico y cómo puede un gestor de red aplicar selectivamente los diferentes controles para cada tipo de tráfico.

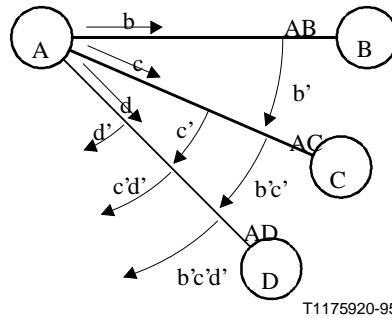


Figura IV.1/Q.823

En la figura, se representan los siguientes tipos de tráfico:

- El tráfico b es tráfico ofrecido directo al subhaz de circuitos AB.
- El tráfico c es tráfico ofrecido directo al subhaz de circuitos AC.
- El tráfico d es tráfico ofrecido directo al subhaz de circuitos AD.
- El tráfico b' es un desbordamiento del tráfico b del subhaz de circuitos AB, que es ahora de encaminamiento alternativo al subhaz de circuitos AC.
- El tráfico c' es un desbordamiento del tráfico c del subhaz de circuitos AC, que es ahora de encaminamiento alternativo al subhaz de circuitos AD.
- El tráfico d' es un desbordamiento del tráfico d del subhaz de circuitos AD, que es ahora de encaminamiento alternativo al subhaz de circuitos siguiente.
- El tráfico b'c' es un desbordamiento del tráfico b' del subhaz de circuitos AC, que es ahora de nuevo de encaminamiento alternativo al subhaz de circuitos AD.
- El tráfico c'd' es un desbordamiento del tráfico c' del subhaz de circuitos AD, que es de nuevo de encaminamiento alternativo al subhaz de circuitos siguiente.
- El tráfico b'c'd' es un desbordamiento del tráfico b'c' del subhaz de circuitos AD, que es de nuevo de encaminamiento alternativo al subhaz de circuitos siguiente.

Obsérvese que el tráfico b' y el tráfico c' es tráfico que se ha desbordado sólo una vez (es decir, tráfico con encaminamiento directo que se ha desbordado); en tanto que el tráfico b'c' y el tráfico c'd' es tráfico que se ha desbordado más de una vez (es decir, tráfico con encaminamiento alternativo que se ha desbordado).

El gestor de red debe tener la flexibilidad de controlar cada tipo del tráfico arriba enumerado utilizando el control cancelación hacia o cancelación desde; especialmente el tráfico que se ha

desbordado sólo una vez y el tráfico que se ha desbordado más de una vez. El cuadro IV.1 ilustra como puede conseguirse.

Cuadro IV.1/Q.823

Tráfico a controlar	Tipo de control
Tráfico b	Cancelación hacia directa en el subhaz de circuitos AB
Tráfico c	Cancelación hacia directa en el subhaz de circuitos AC
Tráfico d	Cancelación hacia directa en el subhaz de circuitos AD
Tráfico b'	Cancelación desde directa en el subhaz de circuitos AB
Tráfico c'	Cancelación desde directa en el subhaz de circuitos AC
Tráfico d'	Cancelación desde directa en el subhaz de circuitos AD
Tráfico b'c'	Cancelación desde alternativa en el subhaz de circuitos AC
Tráfico c'd'	Cancelación desde alternativa en el subhaz de circuitos AD
Tráfico b'c'd'	Cancelación desde alternativa en el subhaz de circuitos AD

Obsérvese que si los subhaces de circuitos AB, AC y AD forman también parte de otras tablas de encaminamiento, los correspondientes desbordamientos en estas tablas de encaminamiento también serán afectados.

APÉNDICE V

Utilización de los datos históricos de la Recomendación Q.822 para mediciones de tráfico

V.1 Introducción

Este apéndice presenta información práctica de la aplicación del modelo de información de gestión de calidad de funcionamiento de la Recomendación Q.822 para la recopilación de mediciones de tráfico. Expone mecanismos para realizar informes autónomos y recuperación a petición de mediciones.

V.2 Visión general del modelo de información de gestión de calidad de funcionamiento

El modelo de gestión de calidad de funcionamiento permite el informe sumario de los parámetros de calidad de funcionamiento utilizando el simpleScanner (escáner simple).

El modelo de objeto para la gestión de calidad de funcionamiento se muestra en la figura 1/Q.822.

Los datos encontrados en las instancias de clases de objeto currentData (datos actuales) (o sus subclases) se componen de mediciones que se hallan en fase de acumulación y cambio durante el intervalo. Al final de éste se inicializan las instancias de objeto currentData. Por tanto, los datos recopilados deben ser inmediatamente comunicados o almacenados para que haya persistencia de los intervalos concluidos. El modelo soporta la retención del último intervalo concluido como instancias de objeto historyData (datos históricos) (o sus subclases).

Al final de cada intervalo de calidad de funcionamiento puede emitirse un scanReport (informe de exploración) directamente por el objeto currentData. Sin embargo, esto daría lugar a una notificación separada para cada instancia currentData que se supervise.

Es más eficaz agrupar las mediciones de todas las instancias `currentData` acumuladas durante el mismo periodo. El modo recomendado es utilizar el `simpleScanner` para sincronizar las instancias `historyData`. El objeto gestionado `simpleScanner` explora los atributos comunes que han de observarse en cada uno de los objetos seleccionados según un calendario especificado y produce un informe agregado.

Las notificaciones generadas por el escáner pueden ser procesadas por un discriminador de retransmisión de eventos (EFD, *even forwarding discriminator*) para informar al sistema gestor.

La generación de informes resumizados PM se representa en la figura B.5/Q.822. La ilustración muestra que el `simpleScanner` puede utilizarse para explorar atributos de los objetos `historyData` y emitir un `scanReport` al EFD. El EFD a su vez envía un informe `m-EVENT` de los datos resumizados siempre que se cumplan las aserciones de valor establecidas (por ejemplo, `eventType = scanReport`).

V.3 Configuración inicial

Esta subláusula trata el proceso en el que se configura el mecanismo de recopilación.

Se instancian las siguientes clases de objeto:

- 1) subclases de `currentData`, por ejemplo, `tmTmCircuitEndPointSubgroupCurrentData`;
- 2) subclases de `historyData`, por ejemplo, `xtpsgHistoryData`;
- 3) `simpleScanner`.

V.3.1 Datos actuales

El intervalo de recopilación se especifica en el atributo `granularityPeriod`, que es heredado del escáner. `currentData` es una subclase del escáner. Ejemplos típicos del valor son cinco minutos o sesenta segundos.

El lote condicional `historyRetentionPkg` se necesita para que pueda establecerse el número apropiado de intervalos de `historyData`. Para fines de gestión de tráfico, este valor se limita a 1 a fin de minimizar el uso de recursos de memoria.

V.3.2 Datos históricos

El periodo de retención (duración) se especifica en el atributo `historyRetention` (retención de datos históricos). Cuando se crea una nueva instancia de `historyData`, puede suprimirse la instancia existente más antigua.

El lote condicional `historyDataSuspectIntervalPkg` es necesario, ya que es de utilidad para saber si las mediciones son sospechosas.

V.3.3 Escáner simple

Se recomienda fijar como se indica los siguientes atributos heredados del escáner y del escáner homogéneo (`homogeneousScanner`) para obtener el COMPORTAMIENTO deseado.

- 1) `scannerId` (heredado de la Recomendación X.739:scanner)
El `scannerId` contiene un valor que se utiliza para identificar una instancia de esta clase de objeto gestionado (utilizada para denominación).
- 2) `administrativeState` (heredado de la Recomendación X.739:scanner)
El `administrativeState` (definida en la Recomendación X.731) se utiliza para suspender o reanudar la función de exploración. Si el estado administrativo tiene el valor DESBLOQUEADO (UNLOCKED), el escáner tiene administrativamente permitido efectuar exploraciones. Inicialmente, el estado debería ser fijado a DESBLOQUEADO más que a BLOQUEADO para que pueda permitirse esa exploración.

- 3) granularityPeriod (heredado de la Recomendación X.739:scanner)
El atributo granularityPeriod indica el tiempo que transcurre entre exploraciones sucesivas. Éste debería fijarse, por ejemplo, a cinco minutos, para obtener una exploración automática e informar al final de cada periodo.
- 4) scope (ámbito) (heredado de la Recomendación X.738:homogeneousScanner)
Se especifica el lote de selección delimitado (scoped). El escáner utiliza la clase de objeto gestionado identificada en el atributo objeto gestionado de base y comprueba todos los objetos gestionados dentro de los niveles indicados por el atributo ámbito (scope). El baseManagedObject que debería ser el nodo de comienzo para seleccionar el objeto gestionado es el managedElement (elemento gestionado) (elemento de red tal como el conmutador de circuito). El valor del atributo ámbito (scope), por ejemplo, se pondrá al nivel individual 2. Esto implica que todos los objetos contenidos en el segundo nivel de la jerarquía de denominación con relación al managedElement son candidatos para la selección. Esta selección es restringida aún más por el scanningFilter (filtro de exploración) como se explica a continuación.
- 5) scanningFilter (heredado de la Recomendación X.738:homogeneousScanner)
El escáner comprueba las instancias de los objetos gestionados dentro de los niveles indicados aplicando los criterios en el atributo scanningFilter.
El filtro puede configurarse para que seleccione entre los siguientes tipos de datos:
 - a) subhaz de circuitos;
 - b) destino;
 - c) central.Por ejemplo, el filtro puede configurarse para seleccionar objetos de datos históricos para las entidades supervisadas subhaz de circuitos, destino y central (objectClass = tmCircuitEndPointSubgroupHistoryData U objectClass = destinationHistoryData U objectClass = exchangePerformanceHistoryData).
- 6) numericAttributeIdArray (de la Recomendación X.738:simpleScanner)
Debe fijarse el numericAttributeIdArray. Es la lista ordenada de atributos cuyos valores han de ser explorados. Los valores estarán contenidos en el numericValueArray.
- 7) supressObjectInstance (heredado de la Recomendación X.738:simpleScanner)
El atributo supressObjectInstance se pone a VERDADERO a fin de que la identificación de instancia de objeto no se incluya en la respuesta. Se reducirá así el número de bytes en el mensaje.

NOTA – Debe evitarse el valor VERDADERO en supressObjectInstance si se incluye el paquete de supresión de cero de currentData.

V.4 Informe autónomo

El mecanismo de informe autónomo se ilustra en la figura V.1: Informe autónomo de mediciones.

El escáner puede ser invocado cada periodo de granularidad de manera autónoma. Los informes autónomos se generan periódicamente mediante fijación apropiada del granularityPeriod del simpleScanner, por ejemplo, cada cinco minutos. Se explorarán los datos históricos apropiados para cada objeto supervisado. Los valores de atributo apropiado se insertarán en el numericValueArray (formación de valores numéricos) de la NOTIFICACIÓN scanReport.

El `numericValueArray` puede tener una secuencia de enteros o números reales. El orden exacto de los identificadores de atributo se conocen del `numericAttributeIdArray` (formación de atributos numéricos).

La notificación se envía al EFD.

La NOTIFICACIÓN `scanReport` es evaluada por el EFD. Debería provisionarse la construcción del discriminador para el envío de las notificaciones apropiadas. Los informes resumizados se remitirán al gestor como informes `m-EVENT`.

V.5 Informe a petición

El mecanismo de informe a petición (`polled reporting`) se ilustra en la figura E.2: Informe a petición de mediciones.

Un mecanismo de informe a petición puede obtenerse poniendo a cero el `granularityPeriod` del `simpleScanner`. Esto neutralizará la exploración autónoma. El gestor (sistema de operaciones) puede iniciar una exploración a petición invocando la acción `activateScanReport` (ACCIÓN) al escáner `simple`, lo cual provocará la actuación del escáner una sola vez (recopilación de los datos y respuesta con resultados). La acción se ejecutará utilizando parámetros de ámbito (`scope`) y `scanningFilter` los valores de los atributos como en el caso de informe autónomo.

El informe resumizado se enviará al gestor como una respuesta `m-ACCIÓN`.

V.6 Cambios al informe

El gestor podrá suspender la generación autónoma de los informes del `simpleScanner` poniendo el `administrativeState` a bloqueado (`locked`), lo que interrumpirá bruscamente el empleo de la exploración. Un planteamiento más elegante consiste en poner el `administrativeState` a cierre (`shutdown`). Éste se cerrará después de concluir el intervalo en curso y emitir el informe.

El gestor podrá cambiar el ámbito, filtro y `numericAttributeIdArray` para efectuar el contenido del informe. Podrán también cambiarse otros parámetros para cambiar el comportamiento del mecanismo de informe, por ejemplo, `granularityPeriod`.

V.7 Conclusiones

Los mecanismos normalizados de la especificación de gestión de calidad de funcionamiento de la Recomendación Q.822 pueden utilizarse en unión del `simpleScanner` de la Recomendación X.738 para sumarizar e informar de las mediciones de tráfico. El informe a petición y el informe autónomo pueden soportarse utilizando los dos mecanismos descritos.

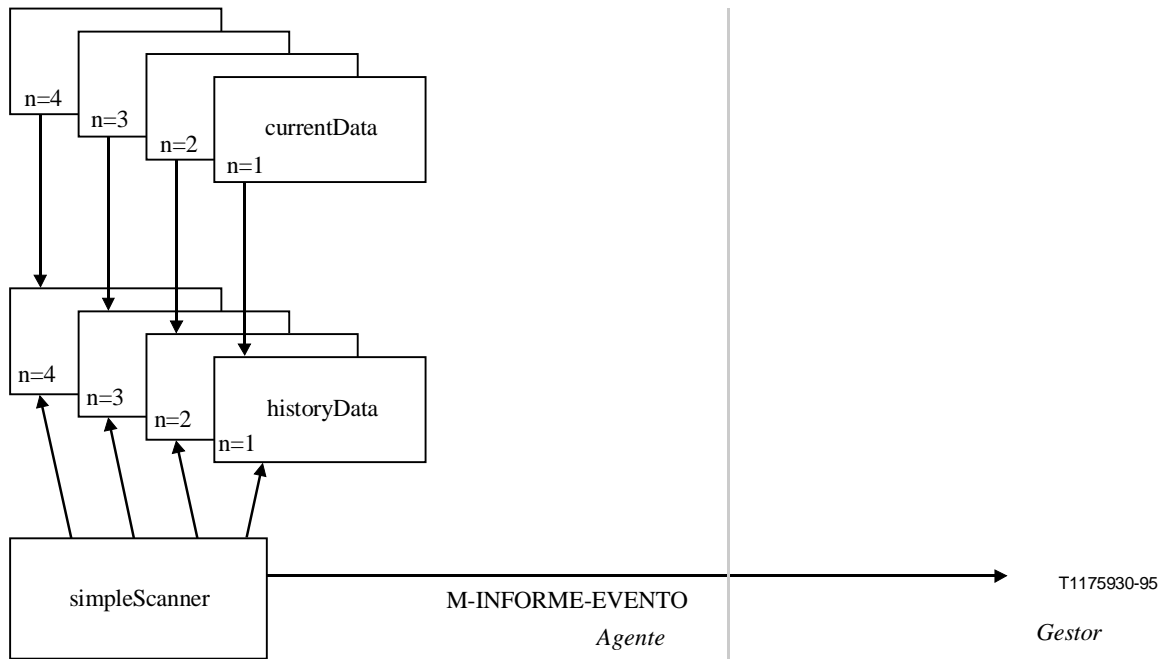


Figura V.1/Q.823 – Informe autónomo de mediciones

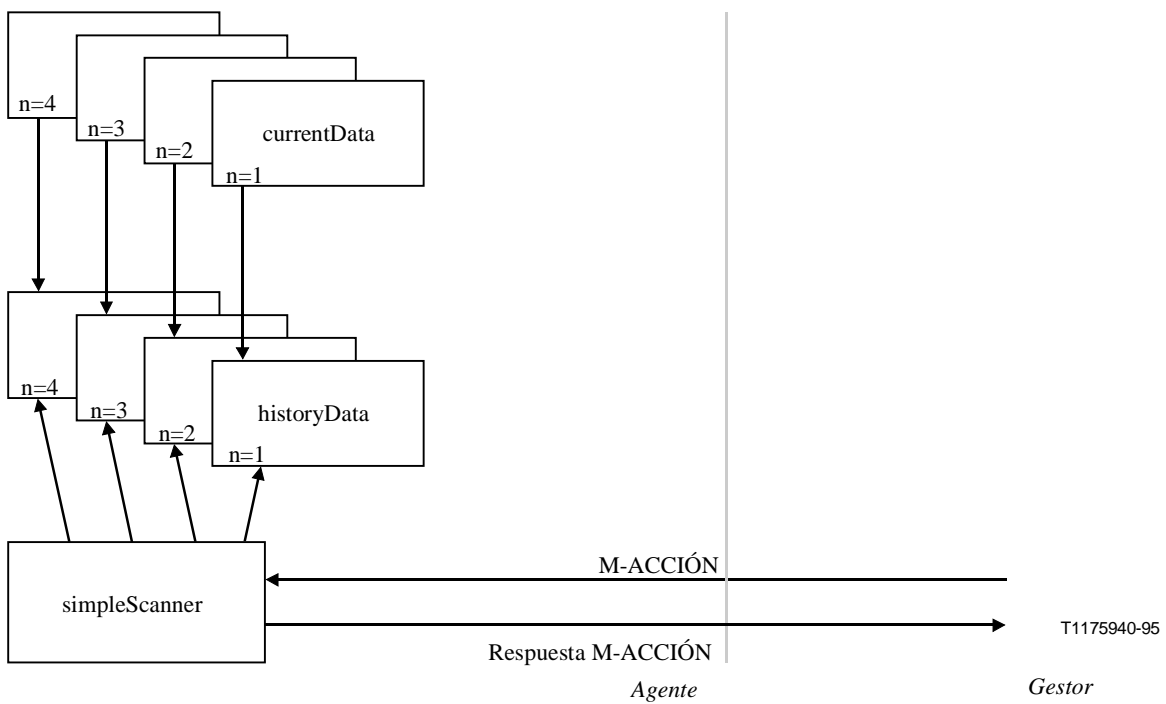


Figura V.2/Q.823 – Informe a petición de mediciones

APÉNDICE VI

Definición de objeto subhaz de punto extremo de circuitos

Hasta que la siguiente definición de objeto `circuitEndPointSubgroup` se pase a la Recomendación M.3100, se incluye aquí para una mejor comprensión y compleción del modelo Q.823.

```
circuitEndPointSubgroup      MANAGED OBJECT CLASS
DERIVED FROM "Recommendation X.721 :1992":top;
CHARACTERIZED BY
  circuitEndPointSubgroupPackage PACKAGE
  BEHAVIOUR
    circuitEndPointSubgroupBehaviour BEHAVIOUR
DEFINED AS
  "The set of circuit endpoints within a group that directly interconnects one exchange with another having
  common values for the attributes listed in the package. Note that the term exchange includes PBX where
  applicable. -- Annex A/E.410 defines circuit subgroup.
  ;;
ATTRIBUTES
circuitEndPointSubgroupId      GET,
numberOfCircuits              GET,
labelOfFarEndExchange        GET,
signallingCapabilities       GET,
informationTransferCapabilities GET,
circuitDirectionality       GET,
transmissionCharacteristics  GET,
userLabel                    GET;
  ;;
NOTIFICATIONS
"Recommendation X.721:1992":attributeValueChange,
"Recommendation X.721:1992":objectCreation,
"Recommendation X.721:1992":objectDeletion,
REGISTERED AS { m3100ObjectClass x};
circuitEndPointSubgroupId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR
    "Recommendation X.721 :1992": rDNIdBehaviour,
    -- The above behaviour is defined as part of discriminatorId in Rec. X.721
    ttpIdBehaviour BEHAVIOUR
  DEFINED AS "The circuit subgroup Id is an attribute type whose distinguished value can be used as a
  RDN when naming an instance of the circuit subgroup object class.";;
REGISTERED AS {m3100Attribute x};
numberOfCircuits ATTRIBUTE
  WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.NumberOfCircuits;
  MATCHES FOR EQUALITY, ORDERING;
  DEFINED AS "The number of circuits in a circuit subgroup.";;
REGISTERED AS {m3100Attribute x};
labelOfFarEndExchange ATTRIBUTE
  WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.UserLabel;
  MATCHES FOR EQUALITY, SUBSTRINGS;
  BEHAVIOUR
    labelOfFarEndExchangeBehaviour BEHAVIOUR
  DEFINED AS "This attribute type assigns a user friendly name to the Far End Exchange terminating this
  circuit subgroup.";;
```

REGISTERED AS {m3100Attribute x};

signallingCapabilities ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.SignallingCapabilities;

MATCHES FOR EQUALITY;

BEHAVIOUR

signallingCapabilitiesBehaviour BEHAVIOUR

DEFINED AS "The attribute type specifies the signalling types supported by the circuit subgroup.";;

REGISTERED AS {m3100Attribute x};

informationTransferCapabilites ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.InformationTransferCapabilities;

MATCHES FOR EQUALITY;

BEHAVIOUR

informationTransferCapabilitiesBehaviour BEHAVIOUR

DEFINED AS "The attribute type specifies the different service types such as speech, 64 Kbs unrestricted data supported by the circuit subgroup.";;

REGISTERED AS {m3100Attribute x};

circuitDirectionality ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.CircuitDirectionality;

MATCHES FOR EQUALITY;

BEHAVIOUR

circuitDirectionalityBehaviour BEHAVIOUR

DEFINED AS "The attribute type specifies the directionality of the circuits in the circuit subgroup.";;

REGISTERED AS {m3100Attribute x};

transmissionCharacteristics ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.TransmissionCharacteristics;

MATCHES FOR EQUALITY;

BEHAVIOUR

transmissionCharacteristicsBehaviour BEHAVIOUR

DEFINED AS "The attribute type specifies the different transmission characteristics such as satellite, echo control supported or not supported by the circuit subgroup. The bit positions are set to indicate if a particular characteristic is supported.";;

REGISTERED AS {m3100Attribute x};

ASN.1 productions:

NumberOfCircuits ::= INTEGER

SignallingCapabilities ::= ENUMERATED {isup(0), isup2(1),ccittNo5(2),r2(3),ccittNo6(4), tup(5)

-- "... " these ellipses defined in ASN.1 amendment are used here to indicate that this is --

-- an extensible type and additional enumerations may be added in future --}

CircuitDirectionality ::= ENUMERATED {onewayOut(0), onewayIn(1), twoway(2)}

InformationTransferCapabilities ::= ENUMERATED {speech(0),

audio3pt1(1),audio7(2),audioComb(3),digitalRestricted56(4), digitalUnrestricted64(5)

-- "... " these ellipses defined in ASN.1 amendment are used here to indicate that this is --

-- an extensible type and additional enumerations may be added in future --}

TransmissionCharacteristics ::= BITSTRING {satellite(0), dCME(1), echoControl(2)}

--creation and deletion of CircuitEndPointSubgroup using the following name binding is outside the scope of this Recommendation.

circuitEndPointSubgroup-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS circuitEndPointSubgroup;
NAMED BY
SUPERIOR OBJECT CLASS "M.3100":managedElement AND SUBCLASSES;
WITH ATTRIBUTE circuitEndPointSubgroupId;
CREATE WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE;

APÉNDICE VII

Correspondencia de funciones NM con unidades funcionales

El cuadro VII.1 ofrece una correspondencia de funciones NM (enumeradas en el apéndice I) con las unidades funcionales.

Cuadro VII.1/Q.823

Funciones de gestión de red	Unidades funcionales (cláusula 8)
1 Funciones de supervisión de situación	
Informe de la disponibilidad de servicio de los elementos de red	H
Informe por demanda de la situación de los controles	H, I o J
Informe de la situación de ocupado/reposo de un haz de circuitos	Indirectamente – Véase datos/parámetros de haz de circuitos por demanda
Informe de la situación de congestión de la central	H, I o J
Informe del recibo de una señal ACC	I o J
Informe por demanda de la situación HTR de los destinos	G, H
Adición/supresión manual de la situación HTR del destino	G
Informe de la situación de congestión de la red de señalización por canal común	Fuera de alcance
Informe del recibo de señales de gestión de señalización por canal común	Fuera de alcance
2 Funciones de supervisión de la calidad de funcionamiento	
Informe programado de datos/parámetros de haz de circuitos	H, D, ii)
Informe por demanda de datos/parámetros de haz de circuitos	H, D, I
Informe programado de mediciones de carga de central	C, ii)
Informe por demanda de mediciones de carga de central	C, I
Informe programado de mediciones de congestión de central	C, ii)
Informe por demanda de mediciones de congestión de central	C, I
Informe programado de medición de carga CCS	Fuera de alcance
Informe por demanda de medición de carga CCS	Fuera de alcance
Informe programado de medición de congestión CCS	Fuera de alcance
Informe por demanda de medición de congestión CCS	Fuera de alcance
Informe programado de datos sobre calidad de funcionamiento de los controles	B, ii)
Informe por demanda de datos sobre calidad de funcionamiento de los controles	B, i)

Cuadro VII.1/Q.823 (fin)

Funciones de gestión de red	Unidades funcionales (cláusula 8)
3 Funciones de control	
Aplicación/modificación/supresión de control manual	A
Establecimiento/modificación/supresión de un control automático	A
Aplicación/modificación/supresión de un anuncio grabado especial	Fuera de alcance
4 Funciones administrativas	
Establecimiento/cambio/supresión de un calendario de mediciones	K
Establecimiento/cambio/actualización de una base de datos de gestión de red	Fuera de alcance
Establecimiento/cambio/supresión de un umbral para informe de situación, informe de datos y determinación HTR	ii)
Establecimiento/cambio/supresión de calendarios para información de situación y datos	ii)
Comunicación por demanda de información sobre la tabla de encaminamientos	Fuera de alcance

APÉNDICE VIII

Soporte de la funcionalidad de auditoría utilizando la Recomendación Q.823

VIII.1 Introducción

Este apéndice describe la utilización del modelo de esta Recomendación, combinado con el modelo de la función de gestión conocimiento de gestión (Recomendación X.750) para soportar las capacidades de auditoría. Las necesidades funcionales de auditoría del gestor se tratan en VIII.2. Los mecanismos que pueden utilizarse para satisfacer estas necesidades se describen en VIII.3.

VIII.2 Requerimientos funcionales

Las auditorías son necesarias para asegurar que el gestor y el agente tienen la misma visión de los datos de referencia (datos que no sean de mediciones). El gestor solicita una auditoría al elemento de red de los datos de la base de información de gestión (MIB, *management information base*) (diferentes de los datos de vigilancia comunicados periódicamente) a fin de sincronizar el conocimiento de la MIB en el gestor con el estado del agente en ese momento. Estas auditorías son iniciadas por el gestor por una o más de las siguientes razones:

- a) en respuesta a anomalías (tales como que faltan datos de medición de un haz troncal cuando el agente tenga planeado enviarlos) detectadas en los datos de medición recibidos;
- b) en respuesta a la creación, supresión y cambios en los valores de información de gestión;
- c) en respuesta a información de alerta tal como congestión o actividades de gestión proporcionada autónomamente por el agente (por ejemplo, clase de objeto stateIndicator), y
- d) a fin de soportar la sincronización periódica del conocimiento de gestión compartido entre el gestor y el agente.

Los mecanismos para la recuperación de datos del agente deben soportar la personalización por el gestor a fin de cumplir uno o más de los siguientes criterios: tipos de datos, volumen de datos, frecuencia de recuperación. Esta recuperación de datos debe soportar auditorías de todos los datos de

referencia (auditoría total) así como la recuperación selectiva de los datos de referencia cambiados desde la última auditoría (auditoría de cambios).

Estos requerimientos, salvo la auditoría de cambios, pueden satisfacerse utilizando los mecanismos descritos a continuación. Se describe a continuación un posible planteamiento para soportar auditoría de cambios si el gestor la necesita.

VIII.3 Tipos de auditoría

VIII.3.1 Auditoría total/parcial

Una auditoría total recupera toda la información en la MIB del NE para la clase de objeto seleccionada. Una auditoría parcial recupera un subconjunto de estos objetos sobre la base de criterios de filtrado aplicados a los atributos de objeto. Los diferentes mecanismos para solicitar información de auditorías se describen en VIII.4.

Esta auditoría solicitada por el OS:

- a) cuando un OS desea construir una visión inicial de la MIB del NE,
- b) cuando el OS desea validar su visión de la MIB con el NE.

Al recibir la información de auditoría, es responsabilidad del OS construir la MIB inicial o determinar y resolver las diferencias incrementales.

VIII.3.2 Auditoría de cambios

Además de la auditoría total/parcial, el OS puede también desear obtener información de auditoría sólo de los cambios efectuados en la MIB del NE desde la última auditoría. Esta necesidad puede satisfacerse mediante el uso de notificaciones existentes tales como creación, supresión o cambio de valor de atributo emitidas desde los objetos de datos de referencia. Esta información puede ser capturada por el OS para determinar y resolver las diferencias incrementales.

Si se desea una verdadera auditoría de cambios, debe entonces hacerse el registro cronológico de las notificaciones de supresión y cambios de valor de atributo. El simpleScanner puede entonces utilizarse para recuperar del registro cronológico especificando del lote timeSelection en el escáner.

VIII.4 Mecanismos

VIII.4.1 Utilización del servicio PT-OBTENCIÓN (PT-GET)

El gestor puede iniciar una petición utilizando la combinación del servicio PT-OBTENCIÓN en la Recomendación X.730 y las características de alcance y filtrado disponibles en el CMISE (Recomendaciones X.710 y X.711). A fin de aplicar esta capacidad, se necesita el soporte del perfil ISO ISP 11183-Part 2, enhanced communications functions. Esta parte del perfil de gestión de red hace obligatorias las unidades funcionales (específicamente la selección de objetos gestionados, respuesta múltiple, filtrado y cancelación-obtención). Además, en el momento del establecimiento de la asociación, estas unidades funcionales del CMIS deben ser negociadas para su utilización en la asociación.

Con este mecanismo, el gestor puede emitir una única petición para recuperar datos sujetos a diferentes criterios. Se dan a continuación ejemplos de las condiciones (no exhaustivas) que el gestor puede solicitar que el agente aplique a fin de determinar si debe emitirse una respuesta. Una respuesta separada será enviada por el agente para cada objeto gestionado que cumpla las condiciones expuestas en la petición. En cualquier momento durante el cual se envíen múltiples respuestas, el gestor puede cancelar la petición emitiendo una petición de obtención de cancelación.

El parámetro CMISFilter se define en términos de aserción de valor de atributo. Es una expresión booleana de verdadero o falso. Con relación al modelo de esta Recomendación, pueden utilizarse aserciones para recuperar datos en los objetos gestionados de control u objetos de datos de referencia existentes. Utilizando esta capacidad, se obtienen valores de atributos de estos objetos gestionados.

A Auditoría de los controles

El siguiente CMISFilter construye una recuperación selectiva permitida basada en el tipo de controles. Obsérvese que pueden combinarse varias de estas condiciones si el gestor ha de recuperar información relativa a más de un tipo de controles.

- | | | |
|----|---|--|
| a) | Espaciamiento de llamadas | {objectClass = destinationCodeControl} |
| b) | Cancelación de encaminamiento directo/alternativo | {objectClass = cancelTo} |
| c) | Cancelación de encaminamiento alternativo desde | {objectClass = cancelFrom} |
| d) | Salto de ruta | {objectClass = skip} |
| e) | TAR hacia | {objectClass = tarTo} |
| h) | TAR desde | {objectClass = tarFrom} |
| g) | Cancelación de desbordamiento reencaminado | {objectClass = cancelRerouted} |
| h) | ACC | {objectClass = acc} |
| i) | Reserva selectiva de circuitos | {objectClass = scr} |
| j) | Destino automático | {objectClass = adc} |
| k) | Todos los controles de protección manuales | {objectClass = cancelFrom OR
objectClass = cancelTo OR
objectClass = skip} |
| l) | Controles de expansión | {objectClass = tarTo OR
objectClass = tarFrom} |
| m) | Todos los controles | {present trafficControlId} |
| n) | Todos los controles automáticos activos | {present trafficControlId AND
administrativeState = unlocked AND
autoActivated = true} |

B Datos de referencia

Pueden utilizarse los siguientes tipos de aserciones para recuperar datos que son designados como datos de referencia en el modelo.

Obsérvese que esta lista no es exhaustiva, sino que sólo se proporciona como ejemplo.

- | | | |
|----|--|---|
| a) | Subhaces de circuitos | {objectClass = circuitSubGroupEndPoint} |
| b) | CSG que son supervisados | {objectClass = circuitSubGroupEndPoint AND
tmSurveillance = true} |
| c) | CSG con un extremo distante específico | {objectClass = circuitSubGroupEndPoint AND
labelOfFarEndExchange = <value of the specific
exchange >} |
| d) | Destino HTR | {objectClass = htrDestination} |
| e) | Destino observado | {objectClass = observedDestination} |

Este mecanismo puede utilizarse para una auditoría total.

VIII.4.2 Utilización de un escáner simple

El mecanismo antes descrito da lugar a una única petición seguida por cierto número de respuestas para cada uno de los objetos seleccionados. Según la condición especificada, el número de respuestas vinculadas a la petición puede ser muy grande. Para reducir el número de respuestas, puede utilizarse otro mecanismo que proporcionará la respuesta combinándolo en un solo informe respuestas aplicables a múltiples objetos. El apéndice V describe cómo puede utilizarse un escáner simple para recuperar datos de los objetos de datos historyData. La capacidad de auditoría puede ser soportada utilizando el mismo mecanismo. En este caso, los objetos gestionados sujetos a auditoría se obtienen configurando el lote de selección deseado en el mecanismo anterior, como se describe en el apéndice V. Esto se efectuaría mediante petición (polling) (por oposición al informe autónomo).

Debe señalarse que, aunque este mecanismo es de mucha utilidad al notificar respuestas cuando se realiza la auditoría total, algunas implementaciones pueden no soportar el envío y el recibo de más de 10 000 octetos de unidades de datos de protocolo de sesión (los datos codificados de aplicación habrán de tener unos pocos octetos menos para permitir la información de encabezamiento procedente de las capas de sesión y de presentación). Este escenario es posible porque la ISP 11183-Part 1 requiere una implementación para soportar como mínimo solamente 10 K octetos para enviar y recibir datos. En ese caso, la respuesta puede ser proporcionada por medio de múltiples respuestas de enlace.

VIII.4.3 Utilización del objeto descubrimiento

El primer mecanismo descrito en VIII.3.1 exige que el agente soporte las características de delimitación de ámbito (scoping) y de filtrado en el CMISE en la interfaz. Con el mecanismo de VIII.3.2, aun cuando el gestor no exija el envío de estas características en la petición, estas capacidades forman parte del escáner simple en el agente. La Recomendación X.750 define una clase de objeto gestionado denominada discoveryObject (objeto descubrimiento). Este objeto gestionado soporta un gestor para obtener conocimiento de gestión cuando el agente no soporta la característica de delimitación de ámbito y de filtrado.

La definición discoveryObject de la Recomendación X.750 incluye una acción mlTSearch. Utilizando esta acción, el gestor obtiene las instancias de los diversos objetos gestionados. Opcionalmente, el gestor puede también solicitar el atributo objectClass para estas instancias. Este método es de utilidad si el gestor sólo requiere los nombres de las instancias y la clase. Sin embargo, si también se requieren los valores de otros atributos para la auditoría, son necesarias nuevas PT-GET (PT-OBTENCIÓN).

El gestor puede emitir peticiones m-Obtención individuales para recuperar datos para la totalidad o un subconjunto de las instancias incluidas en las respuestas del discoveryObject. En otro caso, el gestor puede crear un objeto simpleScanner que incluya la lista de objetos gestionados de la cual deben recuperarse datos, junto con los atributos cuyos valores deberían incluirse en el informe.

VIII.5 Conclusiones

Este apéndice describe cómo puede soportarse la capacidad de auditoría con el modelo existente. Combinando las características del CMISE y los modelos de información disponibles en las Recomendaciones de la UIT-T, se soportan auditorías de los datos de referencia.

Ejemplos de errores de direccionamiento en la Recomendación Q.823

IX.1 Introducción

Al implementar una interfaz Q3 utilizando los requerimientos de protocolo en las Recomendaciones Q.811 y Q.812 y un modelo de información específico de la aplicación tal como el de la Recomendación Q.823, pueden aparecer errores a diferentes niveles. Estos errores pueden clasificarse en dos tipos:

- generados por la máquina de protocolo a diversos niveles, incluidos ACSE y CMISE;
- errores genéricos definidos en el CMISE, pero detectados por el usuario del CMISE (aplicación);
- errores específicos de la aplicación y de los recursos.

Este apéndice describe cómo pueden aplicarse los errores genéricos definidos en el CMISE al modelo de gestión de Tráfico y servicios descritos en esta Recomendación. Se describe por medio de un ejemplo de escenario y no se considera que incluya todos los tipos de errores.

IX.2 Errores generados por la máquina de protocolo

Estos errores surgen a consecuencia de la violación de las transiciones de estado descritas para la máquina de protocolo o de las reglas para utilizar los servicios del protocolo.

Por ejemplo, en la capa de transporte se dictan provisiones para detectar PDU de transporte fuera de secuencia y procedimientos para la recuperación tras errores. En la capa de presentación, si los datos codificados recibidos no pertenecen al conjunto acordado de sintaxis abstracta (negociado a través del conjunto de contextos de definición), se generará un aborto. Además, si no se proporcionan reglas de extensibilidad, pueden producirse abortos de presentación si un sistema envía por ejemplo un valor enumerado (que se incluye en un documento revisado) a otro sistema que no ha sido sobregarado para reconocer este nuevo valor.

Un ejemplo de violación de una regla que puede causar un error a la aplicación es el requerimiento en el CMIP de que debe establecerse una asociación previamente al envío de PDU de CMIP. Si la aplicación emite una PDU de CMIP sin una asociación, ésta será errónea.

La subcláusula IX.3 trata la utilización de errores definidos en el CMISE para su uso por una aplicación.

IX.3 Utilización de errores CMISE

A fin de ilustrar la utilización de errores genéricos definida en el CMISE, considérese el siguiente ejemplo. El modelo de gestión de Tráfico define un `tmCircuitEndPointSubgroup` como una subclase del `circuitEndPointSubgroup` definido en la Recomendación M.3100. La subclase incluye un atributo denominado `tmSurveillance`. El comportamiento de esta clase de objeto tiene una limitación de integridad que satisface las siguientes condiciones previas y posteriores.

Si `tmSurveillance` se pone a verdadero, debe existir entonces un `tmCircuitEndPointSubgroupCurrentData`. La condición previa para crear los datos actuales es que `tmSurveillance` debe poder ponerse a verdadero. Si se suprimen los datos actuales, el atributo debe entonces ponerse a falso. En otras palabras, la condición posterior para suprimir el objeto datos actuales es que el valor del atributo `tmSurveillance` debe ser falso. Si existe una discordancia en este comportamiento, es decir, si el valor de `tmSurveillance` es verdadero sin un objeto datos actuales,

entonces la limitación de integridad es violada y ésta será una implementación no conforme. Esto también se cumple en sentido contrario.

Supongamos que el gestor está interesado en recoger datos de vigilancia para subhaces de punto extremo de circuitos. Puede emitirse una petición CMIP de fijar el valor de un atributo en uno de los siguientes métodos.

- 1) una petición de modificar uno o más valores de uno o más atributos en un objeto gestionado único utilizando una operación fijación no confirmada;
- 2) una petición de modificar uno o más valores de uno o más atributos en un objeto gestionado único utilizando una operación fijación confirmada;
- 3) una petición de modificar uno o más valores de uno o más atributos en múltiples objetos utilizando delimitación de ámbito (scoping) (independientemente de que sea con o sin filtrado) utilizando una operación no confirmada;
- 4) una petición de modificar uno o más valores de uno o más atributos en múltiples objetos utilizando delimitación de ámbito (scoping) (independientemente de que sea con o sin filtrado) utilizando una operación confirmada.

Los dos últimos casos pueden todavía dividirse en dos modos de sincronización: atómica y del mejor esfuerzo.

A continuación se describen los errores que pueden surgir al fijar el atributo `tmSurveillance`.

En el primer caso, si el sistema es incapaz de crear el objeto datos actuales, no se proporcionará una respuesta de errores. Sin embargo si se elige este modo de fijación, el discriminador de retransmisión de evento debe configurarse de manera que se emitan notificaciones de creación. La ausencia de notificaciones de creación indicará que los datos actuales no se crearon. Es también posible emitir una continuación de lectura del atributo para terminar este resultado. La forma de mostrar este error en una pantalla de operador cae fuera del alcance de la interfaz máquina-máquina descrita en esta Recomendación. Otra opción es hacer que el `tmCircuitEndPointSubgroup` envíe una notificación de cambio de valor de atributo con los valores antiguo y nuevo del atributo `tmSurveillance`.

Si el atributo `tmSurveillance` ya es verdadero y hay presente una instancia de datos actuales, la solicitud de una puesta a verdadero no tendrá efecto. Debe especificarse el comportamiento esperado cuando los datos actuales ya existan.

En el segundo caso, es posible una respuesta de error a la petición fijación. La respuesta de error puede utilizar uno de los códigos adecuados (exceso denegado, clase `InstanceConflict`, `noSuchObjectClass`, `noSuchObjectInstance`, `processingFailure` sin ninguna información adicional, en el supuesto de que no haya plantillas de parámetros definidas para esta clase de objeto, `setListError` con o sin el valor proporcionado en la petición). Si la operación no se realiza debido a limitaciones de recursos, puede emitirse una PDU RO-rechazo con el código de error de `resourceLimitation`. El recibo de este error y/o el no recibo de la notificación de creación de objeto de los datos actuales debe indicar que se proporciona un mensaje de error al operador. También esta vez la traducción del código de error en la PDU a la interfaz amigable para el usuario cae fuera del alcance de la RGT.

El tercer caso es similar al primero, excepto que el atributo `tmSurveillance` en múltiples objetos ha de fijarse si se crean los correspondientes objetos de datos actuales. Debido a que ésta es una petición no confirmada, la serie de notificaciones de creación de objetos será recibida para los objetos de datos actuales creados. Estas notificaciones incluirán los nombres del objeto datos actuales (incluyendo el nombre del `circuitEndPointSubgroup` que los contiene). Estas notificaciones pueden entonces utilizarse para determinar si se han creado todos los objetos de datos actuales. Esto supone que el gestor tiene conocimiento de los nombres de los subhaces de punto extremo de circuitos para los cuales se requieren. Sobre la base de las notificaciones de creación, el gestor puede ahora

proporcionar al operador un mensaje amigable para el usuario (fuera del alcance de esta Recomendación) que indique dónde se encontraron los errores. El operador puede entonces solicitar que se le devuelva un cometido. Esto significaría que la traducción entre la interfaz de usuario y la interfaz CMIP habría de suprimir todas las instancias de datos actuales para las que se han recibido notificaciones de creación. Cuando se emite la petición supresión, las constricciones de integridad exigirán que el atributo de `tmSurveillance` en el objeto contenido se ponga a falso. (Esto debería indicarse en la vinculación de nombre de las clases de objeto de datos actuales consideradas.)

El cuarto caso es similar al caso 2. Las respuestas vinculadas con notificaciones `setListError` y/o notificaciones de creación de objeto pueden ser analizadas para determinar los subgrupos de punto extremo de circuitos para los cuales no han sido creados objetos de datos actuales.

Las citadas descripciones de los casos 3 y 4 son aplicables si la operación especificó el parámetro de sincronización al mejor esfuerzo o si no se incluyó en la petición ningún valor para este parámetro. Se desea que los objetos de datos actuales sean creados sólo si todos ellos son creables, debería entonces especificarse la sincronización atómica. La forma de implementar esto en términos de bloqueo de la información hasta que hayan concluido todas las operaciones cae fuera del alcance del modelo. Además, si existe un fallo de enlace de comunicación durante cualquiera de los casos, se necesitarán entonces procedimientos de recuperación para asegurar la coherencia de la MIB. También esto cae fuera del alcance del modelo.

Otro ejemplo de dónde pueden aparecer errores se refiere a la gestión de funciones no presentes en un determinado tipo de elemento de red. Por ejemplo, el atributo `destinationAspect` incluye valores enumerados correspondientes a funciones tales como difícil de alcanzar (`hard to reach`). En cualquier definición de atributo, pueden proporcionarse valores para abarcar una variedad de tipos de elementos de red con variadas capacidades. Si se emite una petición de fijación para fijar el valor correspondiente, por ejemplo difícil de alcanzar, y el NE no soporta esta funcionalidad, se emitirá entonces el valor `invalidAttribute` de error CMISE (suponiendo que ésta es una petición confirmada). Las restricciones sobre los valores soportados en una implementación de un producto deben especificarse en el formulario de MOCS. Éste se utilizaría entonces para entender por qué el valor atributo no válido se recibe aun cuando éste es un valor válido definido en la norma.

IX.4 Utilización de errores específicos

Además de los errores genéricos definidos por el CMISE, existe como parte de la definición CMIP la posibilidad de incluir errores específicos de los recursos. Estos errores específicos tendrán que especificarse utilizando plantillas de parámetros e identificarse en el modelo de información (por ejemplo, vinculados a un atributo para indicar que un determinado error con la sintaxis considerada podría ser emitido cuando se modifica un atributo).

El siguiente ejemplo se compuso para ilustrar este aspecto.

Supongamos que un elemento de red está concebido para soportar el control "cancelación hacia" (`Cancel To`), y que, debido a la limitación de recursos, el suministrador especifica el máximo número de objetos `cancelTo` que pueden ser soportados por el NE. La petición creación fracasará si, de resultas de la creación, se supera el número permitido de objetos gestionados `cancelTo`. Puede proporcionarse un rechazo genérico denominado `resourceLimitation` (limitación de recursos). Por otra parte, en la Recomendación M.3100 se define un error específico denominado parámetro `createError`. Si este parámetro se incluye en la vinculación de nombre de la clase de objeto `cancelTo`, pueden entonces proporcionarse un error de fallo de procesamiento acompañado por un entero que represente el máximo número de objetos `cancelTo` soportables por el elemento de red. La interfaz de usuario puede entonces traducir esto a una cadena significativa y visualizarla al operador.

IX.5 Conclusión

Como se ha indicado antes, la mayoría de las condiciones de error pueden ser soportadas por los errores genéricos definidas en el CMISE especificando constricciones de integridad. Además, pueden incluirse errores específicos, pero deberían incluirse en las plantillas para la clase de objeto, atributo, acción o notificación gestionado. Errores tales como temporización (time out) y fallo del enlace caen fuera del alcance del modelo de información. Estas condiciones han de determinarse mediante especificaciones de implementación de la interfaz. No se han identificado en el modelo errores específicos.

APÉNDICE X

Jerarquía de control

A falta de una jerarquía de control en la Recomendación E.412, cuando hay múltiples controles simultáneamente activos en la central, se aplican las siguientes reglas a fin de determinar la jerarquía de los controles:

- Como el destino es determinado antes de la búsqueda de subhaces de circuitos, los controles de destino tienen precedencia sobre cualesquiera controles de subhaz de circuitos.
- Los controles de subhaz de circuitos se aplican entonces en dos etapas: etapa 1: antes de buscar un miembro en reposo en el subhaz de circuitos; y etapa 2: después de desbordar de todos los circuitos del subhaz de circuitos pero antes de la búsqueda del subhaz de circuito alternativo en la cadena de encaminamiento. Los controles en la etapa 1 tiene precedencia sobre los controles en la etapa 2.

Se considerará la siguiente jerarquía descendente de controles de Tráfico durante el proceso de tratamiento de la llamada:

- 1) Control de indicativo de destino
- 2) Control automático de destino
- 3) TAR hacia)
- 4) Cancelación hacia)
- 5) Salto de ruta) Etapa 1
- 6) Reserva selectiva de circuitos)
- 7) ACC)

- 8) Cancelación de desbordamiento reencaminado)
- 9) TAR desde) Etapa 2
- 10) Cancelación desde)

Cuando no se especifica en este apéndice la prioridad entre controles, se considera asunto local.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Red telefónica y RDSI
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión
Serie H	Transmisión de señales no telefónicas
Serie I	Red digital de servicios integrados
Serie J	Transmisiones de señales radiofónicas y de televisión
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	Mantenimiento: 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
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Equipos terminales y protocolos para los 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