

# **Análisis de redes y sistemas de comunicaciones**

POLITEXT

Xavier Hesselbach Serra  
Jordi Altés Bosch

# Análisis de redes y sistemas de comunicaciones

EDICIONS UPC

La presente obra fue galardonada en el octavo concurso  
"Ajut a l'elaboració de material docent" convocado por la UPC.

Primera edición: octubre 2002

Diseño de la cubierta: Manuel Andreu

© Los autores, 2002

© Edicions UPC, 2002  
Edicions de la Universitat Politècnica de Catalunya, SL  
Jordi Girona Salgado 31, 08034 Barcelona  
Tel.: 934 016 883 Fax: 934 015 885  
Edicions Virtuals: [www.edicionsupc.es](http://www.edicionsupc.es)  
E-mail: [edicions-upc@upc.es](mailto:edicions-upc@upc.es)

Producción: CPET (Centre de Publicacions del Campus Nord)  
La Cup. Gran Capità s/n, 08034 Barcelona

Depósito legal: B-41612-2002  
ISBN: 84-8301-611-7

Quedan rigurosamente prohibidas, sin la autorización escrita de los titulares del copyright, bajo las sanciones establecidas en las leyes, la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares de ella mediante alquiler o préstamo públicos.

## Prólogo

Esta obra que está en sus manos es fruto del reto de cubrir un hueco bibliográfico en la descripción de las redes de comunicaciones y en la aplicación de herramientas para su análisis.

En el ánimo de los autores se encuentra ofrecer una visión descriptiva y general de las tecnologías de red actuales, contemplando la tan frecuentemente olvidada vertiente analítica, indispensable tanto para la comprensión del funcionamiento de los sistemas como para su correcto dimensionado.

Es por este motivo que este libro se estructura del siguiente modo: un capítulo inicial para presentar los conceptos y definiciones fundamentales en el área de las redes telemáticas, que permite introducir en el siguiente la descripción de los principales métodos y algoritmos de las modernas tecnologías (que por avanzadas que sean, siempre tienen fecha de caducidad). Se dedica un capítulo a las herramientas de análisis basadas en la teoría de colas, cuyos modelos permiten el estudio de todas estas tecnologías, con instrumentos de trabajo perennes en el tiempo. No se pasan por alto los métodos de acceso tradicionales, ni tampoco las redes de banda ancha, con especial atención a los problemas específicos a los que están enfocadas y las soluciones que aportan.

Confiamos en que los esfuerzos dedicados a la culminación de este libro sean útiles tanto a estudiantes de los últimos cursos de ingeniería especializada en telemática como a jóvenes estudiantes de doctorado, que buscan comprender los fundamentos de las redes de comunicaciones y conocer las principales herramientas de análisis para abordar el problema de estudio del comportamiento de un determinado sistema o red.

Las tecnologías de red nacen, se desarrollan y acaban por desaparecer. Algunas, como Ethernet, parecen revitalizarse día a día, lejos de seguir la suerte de muchas otras contemporáneas suyas que ya están casi olvidadas. Sin embargo, los conceptos de red perduran. El lector encontrará en estas páginas implementaciones y análisis, las primeras sufriendo el paso del tiempo, pero las segundas mostrando metodologías de análisis que se va aplicando y se podrán aplicar a las tecnologías existentes y las que están por llegar en el futuro.

## Índice

<b>PRÓLOGO</b> .....	<b>7</b>
<b>ÍNDICE</b> .....	<b>9</b>
<b>1. INTRODUCCIÓN</b> .....	<b>13</b>
1.1 DEFINICIONES BÁSICAS .....	13
1.1.1 <i>La codificación de canal</i> .....	13
1.1.2 <i>La modulación de la señal</i> .....	15
1.1.3 <i>Los medios de transmisión</i> .....	16
1.1.4 <i>El multiplexado de la señal</i> .....	20
1.1.5 <i>Clasificación de la transmisión según su sentido</i> .....	23
1.1.6 <i>Transmisión síncrona/asíncrona</i> .....	23
1.1.7 <i>Capacidad de canal</i> .....	23
1.2 CONCEPTOS BÁSICOS DE REDES DE COMUNICACIONES .....	24
1.2.1 <i>Servicios orientados y no orientados a conexión</i> .....	26
1.2.2 <i>Tipos de redes según su capacidad de cobertura</i> .....	26
1.2.3 <i>Clasificación de las topologías de red</i> .....	27
1.2.4 <i>Clasificación de los tipos de conmutación de datos</i> .....	28
1.2.5 <i>Clasificación del tráfico en clases</i> .....	30
1.3 CONCEPTOS BÁSICOS DE TELETRÁFICO.....	31
<b>2. ARQUITECTURAS DE COMUNICACIONES</b> .....	<b>33</b>
2.1 PERSPECTIVA HISTÓRICA .....	33
2.2 CONCEPTOS DE ARQUITECTURAS DE COMUNICACIONES .....	34
2.3 ANALOGÍA DE UNA ARQUITECTURA DE COMUNICACIONES.....	37
2.4 EL MODELO DE REFERENCIA OSI DE LA ISO .....	39
2.4.1 <i>Introducción</i> .....	39
2.4.2 <i>Terminología OSI</i> .....	41
2.4.3 <i>La capa física</i> .....	42
2.4.4 <i>La capa de enlace de datos</i> .....	43
2.4.5 <i>La capa de red</i> .....	51
2.4.6 <i>La capa de transporte</i> .....	56
2.4.7 <i>La capa de sesión</i> .....	59
2.4.8 <i>La capa de presentación</i> .....	63
2.4.9 <i>La capa de Aplicación</i> .....	71

<b>3. INTRODUCCIÓN AL ANÁLISIS MEDIANTE TEORÍA DE COLAS.....</b>	<b>81</b>
3.1 INTRODUCCIÓN.....	81
3.2 PROCESOS DE POISSON.....	83
3.2.1 <i>Definición de proceso de Poisson</i> .....	83
3.2.2 <i>Propiedades</i> .....	84
3.2.3 <i>Distribución de las llegadas en un proceso de Poisson</i> .....	86
3.2.4 <i>Propiedad de superposición</i> .....	87
3.2.5 <i>Propiedad de descomposición</i> .....	87
3.3 CADENAS DE MARKOV.....	88
3.3.1 <i>Sistemas de tiempo discreto y sistemas de tiempo continuo</i> .....	89
3.3.2 <i>Cadenas de Markov de tiempo continuo</i> .....	89
3.3.3 <i>Ecuación de futuro</i> .....	91
3.3.4 <i>Procesos de nacimiento y muerte</i> .....	92
3.3.5 <i>Ejemplo</i> .....	93
3.3.6 <i>Procesos de nacimiento y muerte en régimen permanente</i> .....	95
3.3.7 <i>Estudio mediante flujos</i> .....	95
3.3.8 <i>Cálculo de las probabilidades de estado de los procesos de nacimiento y muerte</i> .....	96
3.4 FÓRMULA DE LITTLE.....	97
3.5 LA NOTACIÓN DE KENDALL Y LOS MODELOS DE COLAS.....	97
3.6 LA COLA M/M/1.....	98
3.6.1 <i>Modelo de cola</i> .....	98
3.6.2 <i>Probabilidades de estado</i> .....	99
3.6.3 <i>Número medio de unidades en el sistema</i> .....	100
3.6.4 <i>Tiempo medio de permanencia de una unidad en el sistema</i> .....	100
3.6.5 <i>Ejemplo numérico</i> .....	101
3.7 LA COLA M/M/ $\infty$ .....	102
3.7.1 <i>Modelo de cola</i> .....	102
3.7.2 <i>Probabilidades de estado</i> .....	103
3.8 LA COLA M/M/M. ERLANG C.....	103
3.8.1 <i>Modelo de cola</i> .....	103
3.8.2 <i>Probabilidades de estado</i> .....	104
3.9 LA COLA M/M/M/M. ERLANG B.....	105
3.9.1 <i>Modelo de cola</i> .....	105
3.9.2 <i>Probabilidades de estado</i> .....	106
3.9.3 <i>Situación de bloqueo. Función de Erlang B</i> .....	107
3.9.4 <i>Aplicación típica: Dimensionado de un sistema</i> .....	108
3.9.5 <i>Relación de recurrencia</i> .....	108
3.9.6 <i>Ejemplo</i> .....	109
3.9.7 <i>Ejemplo de dimensionado</i> .....	109
3.10 PROBABILIDAD DE DEMORA EN ERLANG C. RELACIÓN CON ERLANG B.....	110
3.10.1 <i>Expresión analítica</i> .....	110
3.10.2 <i>Número de elementos en cola</i> .....	111
3.10.3 <i>Ejemplo</i> .....	111
3.11 LA COLA M/G/1.....	113
3.11.1 <i>La fórmula de Pollaczek-Khinchine</i> .....	113
3.11.2 <i>Ejemplo 1: Servicio exponencial</i> .....	114
3.11.3 <i>Ejemplo 2: Servicio determinista</i> .....	114
3.12 DIVERSOS ESCENARIOS DE ESTUDIO.....	115

<b>4. REDES DE ÁREA LOCAL.....</b>	<b>117</b>
4.1 PROTOCOLOS DE ACCESO ALEATORIO.....	117
4.2 ALOHA .....	118
4.2.1 Algoritmo de acceso .....	118
4.2.2 Análisis .....	118
4.3 S-ALOHA.....	122
4.3.1 Algoritmo de acceso .....	122
4.3.2 Análisis .....	122
4.3.3 S-Aloha con población finita.....	125
4.4 REDES CSMA .....	126
4.5 REDES CSMA/CD.....	129
4.6 ALGORITMO DE BACK-OFF EXPONENCIAL BINARIO TRUNCADO .....	130
4.7 ESTÁNDARES IEEE PARA REDES LOCALES.....	131
4.8 EL ESTÁNDAR IEEE 802.2 (LLC) .....	133
4.9 EL ESTÁNDAR IEEE 802.3 (ETHERNET).....	136
4.9.1 Introducción .....	136
4.9.2 Subcapa MAC de IEEE 802.3. Servicio y protocolo .....	136
4.9.3 Configuraciones topológicas de la tecnología Ethernet.....	137
4.9.4 Definición de trama en una red Ethernet .....	139
4.9.5 Fast-Ethernet, Gigabit Ethernet .....	139
4.9.6 Consideraciones en la compatibilidad entre Ethernet de diversas capacidades.....	141
4.9.7 Incremento de capacidad a 1000 Mbit/s .....	142
4.9.8 La solución de Metro Ethernet Forum .....	144
<b>5. REDES PÚBLICAS DE DATOS.....</b>	<b>145</b>
5.1 LA RED DIGITAL DE SERVICIOS INTEGRADOS .....	145
5.1.1 Introducción .....	145
5.1.2 Arquitectura de la RDSI .....	147
5.1.3 Tipos de acceso a la RDSI de banda estrecha.....	149
5.2 LA RED FRAME RELAY .....	151
5.2.1 Objetivos.....	151
5.2.2 Características básicas.....	152
5.2.3 La conmutación .....	153
5.2.4 Arquitectura de protocolos.....	155
5.2.5 Acceso a una red Frame Relay.....	156
5.2.6 El nivel físico .....	156
5.2.7 El nivel de enlace.....	157
5.2.8 Parámetros de contrato.....	158
5.2.9 Control de congestión.....	159
<b>6. LA RED DE BANDA ANCHA .....</b>	<b>163</b>
6.1 INTRODUCCIÓN.....	163
6.1.1 Problemática .....	163
6.1.2 Caracterización de los servicios.....	163
6.2 TÉCNICAS DE MULTIPLEXADO PARA LA B-ISDN.....	165
6.3 MODELO DE PROTOCOLOS DE LA B-ISDN .....	166
6.4 DIVISIÓN JERÁRQUICA ATM .....	167
6.5 TRANSPORTE Y CONMUTACIÓN.....	167
6.6 ESTABLECIMIENTO DE CONEXIONES .....	168

6.7 INTERFASES Y FORMATOS DE CELDA.....	170
6.8 LA CAPA DE ADAPTACIÓN .....	171
6.8.1 AAL1 .....	173
6.8.2 AAL2.....	173
6.8.3 AAL3/4.....	174
6.8.4 AAL5.....	175
6.9 GESTIÓN DE TRÁFICO.....	176
6.9.1 Introducción .....	176
6.9.2 Definición de parámetros.....	177
6.9.3 Notación .....	178
6.9.4 Control de Admisión de la Conexión.....	179
6.9.5 Función de UPC.....	179
6.9.6 Conformación de tráfico.....	181
6.9.7 El servicio ABR.....	182
6.9.8 Comportamiento de los conmutadores ante congestión.....	183
6.9.9 Comportamiento de las fuentes .....	184
6.9.10 La repartición de los recursos disponibles ABR.....	185
6.9.11 Cálculo del caudal justo mediante criterio Max-Min.....	186
6.9.12 Ejemplo práctico: Adaptador de red VMA-200 de Fore Systems.....	188
<b>BIBLIOGRAFÍA .....</b>	<b>191</b>

## 1. Introducción

Este capítulo presenta brevemente los conceptos y términos más básicos relacionados con las redes y sistemas de comunicación. Algunos de los conceptos se desarrollarán más ampliamente a lo largo del libro.

### 1.1 Definiciones básicas

Comencemos por algunas definiciones básicas en el campo de la transmisión de datos:

*Transmisión analógica*: aquella que usa señales que toman valores continuos (de un conjunto infinito de valores) a lo largo del tiempo (normalmente también continuo).

*Transmisión digital*: aquella que usa señales que toman valores discretos (de un conjunto finito de valores) a lo largo del tiempo (dividido usualmente en unidades elementales iguales).

*Codificación (de canal)*: mecanismo de conversión de una señal digital a otra también digital más adecuada a un propósito determinado; generalmente para obtener alguna ventaja en su proceso de transmisión. Así, por ejemplo, puede desearse:

- a) reducir el número de cambios en la señal transmitida (ancho de banda mínimo)
- b) incluir información de temporización o sincronización
- c) eliminar la existencia de componente continua
- d) incluir cierta inmunidad al ruido o interferencia o, equivalentemente, cierta capacidad de detección/corrección de errores

#### 1.1.1 La codificación de canal

De los muchos métodos de codificación existentes, presentaremos solamente dos para ilustrar la idea de la codificación.

Supongamos una secuencia digital de datos a transmitir, es decir, de “ceros” y “unos” cuya duración individual es el tiempo de bit ( $T_b$ ).

La codificación *NRZ* (*Non Return to Zero*), que asigna un nivel de tensión bajo, digamos de  $-V$  Volt, a los “ceros” y un nivel de tensión alto, digamos de  $+V$  Volt, a los “unos”.

La codificación *Manchester* diferencial. Este mecanismo puede describirse mediante las dos reglas siguientes: a) siempre existe transición (cambio de nivel bajo a alto o viceversa) en la mitad del bit; b) existe una transición adicional al inicio del bit, si éste es un “cero”, no existiendo en caso de ser un “uno”.

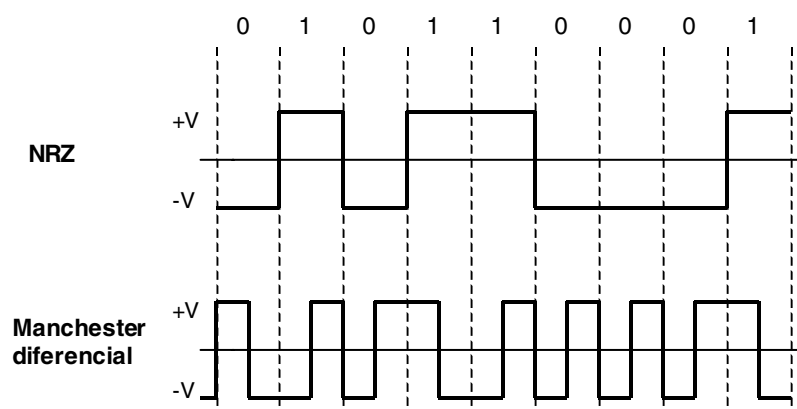


Fig. 1.1 Secuencia de bits codificada según NRZ y Manchester diferencial

Veamos algunas características de ambos esquemas de codificación. Usando la codificación NRZ, la componente continua de la señal transmitida (es decir, su valor medio a largo término) fluctúa en función de la proporción de “ceros” y “unos” de la secuencia a transmitir. Por el contrario, usando la codificación *Manchester* diferencial podemos ver, tras cierta reflexión, que la componente continua es siempre nula, independientemente de la proporción de “0”s y “1”s de la secuencia original. Este hecho permite el acoplamiento de las señales así codificadas mediante transformadores o condensadores (que proporcionan un aislamiento muchas veces deseado entre los diferentes elementos del sistema de transmisión).

Otra ventaja de la ausencia de componente continua en las transmisiones que usan códigos Manchester diferenciales es la eliminación de fenómenos de corrosión electrolítica en los conectores y de fallos, por tanto, en éstos (Recordemos que dos metales distintos puestos en contacto a través de cierta humedad ambiental pueden originar fenómenos de descomposición electroquímica).

Pero quizás la mayor ventaja de usar el código Manchester sea la de la autosincronización (*self-clocking*) a nivel de bit que proporciona. En efecto, usando códigos NRZ, largas secuencias de “ceros” o “unos” consecutivos hacen difícil en el receptor determinar exactamente cuantos bits iguales se han transmitido debido a las inevitables diferencias entre los relojes (los que definen el  $T_b$ ) de transmisión y recepción.

La solución a ese problema pasaría por utilizar una línea de transmisión adicional con la información de reloj del transmisor o relojes extremadamente precisos (los llamados relojes atómicos, de cesio o rubidio) y caros en los lados transmisor y receptor. Por el contrario, las señales codificadas según el esquema Manchester, independientemente de la secuencia original, siempre presentan transiciones frecuentes (al menos cada  $T_b$  s y, a veces, cada  $T_b/2$  s) que permiten al receptor ajustar continuamente el reloj con el que muestrea las señales recibidas.

Por último, también puede verse, tras cierta reflexión, que la información que transporta el código Manchester diferencial está asociada a las transiciones y no a los valores, bajo o alto, de la señal. Es decir, podríamos intercambiar sin ningún problema los hilos que llevan la señal. (Esto es especialmente interesante en los actuales sistemas de cableado de red que usan líneas pares trenzados.)

Las ventajas del código Manchester diferencial respecto al código NRZ tienen su precio. Mientras el ritmo máximo de variación de las señales codificadas en NRZ es una vez cada  $T_b$  s, dicho ritmo llega a ser el doble en las señales codificadas en Manchester diferencial, es decir, cada  $T_b/2$  s. Dicho de otro modo, el espectro de potencia de las señales codificadas en Manchester diferencial se extiende a frecuencias más altas. Dado que las líneas de transmisión responden peor a frecuencias elevadas, al usar códigos Manchester tendremos más problemas (errores) al decodificar en recepción las señales, o bien deberemos utilizar líneas de transmisión de mayor calidad (y, por tanto, mayor coste), o bien limitar las distancias alcanzables.

Existen muchos otros códigos, cada uno de ellos orientado a obtener alguna ventaja durante la transmisión de señales digitales a través de las líneas de transmisión. Los dos códigos presentados, sin embargo, ilustran de forma bastante clara la idea y los objetivos que hay detrás de la función de codificación.

La codificación descrita aquí se suele llamar *codificación de canal* para distinguirla de la *codificación de fuente*. Esta última se aplica en un nivel superior, está íntimamente relacionada con la clase de información a enviar (datos, sonidos, imágenes, ...) y su función principal es la *compresión* (reducción de la cantidad de bits necesaria para representar una información).

### 1.1.2 La modulación de la señal

Se puede definir la *modulación* como el mecanismo de conversión de una señal, digital o no, en otra analógica con el fin de obtener alguna ventaja en su transmisión o de separarla de otras señales dentro de un mismo medio de transmisión.

La modulación se basa en la utilización de una señal *portadora*, generalmente una onda senoidal de frecuencia más elevada o variación más rápida que la de la señal original. La modulación consiste en la modificación de al menos uno de los parámetros (amplitud, frecuencia o fase) de la portadora como

función de la señal original. Así tendremos los siguientes tipos de modulación (para fuentes analógicas o digitales, respectivamente):

- a) AM (*Amplitude Modulation*) o ASK (*Amplitude Shift Keying*)
- b) FM (*Frequency Modulation*) o FSK (*Frequency Shift Keying*)
- c) PM (*Phase Modulation*) o PSK (*Phase Shift Keying*)

o, cuando se modifican simultáneamente los parámetros de amplitud y fase:

- d) QAM (*Quadrature and Amplitude Modulation*)

Una característica del proceso de modulación es que, desde el punto de vista espectral, se produce una translación en frecuencia (aparte de probable modificación) del espectro de la señal original; esto no sucede en los procesos de codificación (donde sólo se produce una modificación, ensanchamiento o reducción, del espectro original). Por ello, a veces, se verá la palabra *broadband* (banda ancha) asociada a los sistemas donde se utilizan modulaciones, y *baseband* (banda base) donde se utilizan codificaciones.

### 1.1.3 Los medios de transmisión

Los *medios de transmisión* son aquellas estructuras físicas que soportan la propagación de las ondas electromagnéticas asociadas a los bits (o a señales analógicas, también) a enviar de un punto geográfico a otro. Constituyen, por tanto, la infraestructura más básica de toda red de comunicaciones. Con el fin de establecer algún criterio de cara a su clasificación y presentación, diremos que pueden ser *guiantes* y *no-guiantes*.

En los *medios guiantes*, el campo electromagnético asociado a la información transportada está confinado alrededor de la estructura física del medio. Están realizados a base de:

- a) materiales de elevada conductividad eléctrica (cobre, aluminio, aleaciones, recubrimientos de plata, oro, etc.) o elevada conductividad óptica (vidrio silíceo)
- b) materiales dieléctricos, no necesarios desde el punto de vista de transmisión pero sí para ejercer funciones de soporte mecánico, de aislamiento o de protección (polietileno, nylon, teflón, PVC, papel, encerado, cauchos, etc.).

Los tipos de medios guiantes son:

- a) línea de pares paralelos
- b) línea de pares trenzados
- c) línea coaxial
- d) fibra óptica

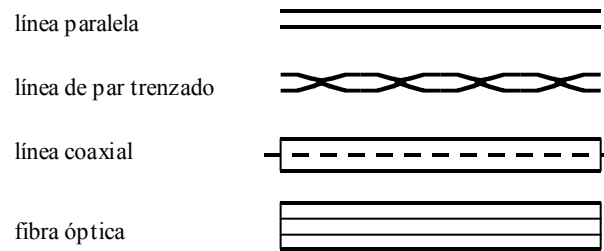


Fig. 1.2 Distintos tipos de medios físicos de transmisión “guiantes”

En los *medios no-guiantes*, el campo electromagnético asociado a la información transportada no está confinado y se extiende por el material de soporte que llamaremos *éter* (el espacio aéreo o el vacío). En realidad, no existe medio de transmisión y estamos hablando de transmisión por ondas de radio (desde las de frecuencia más baja -*VLF: Very Low Frequency*- hasta las de frecuencia más elevada - $\mu W$ : *micro-Waves*-) o por ondas luminosas (usualmente infrarrojas o láser).

Se pueden describir cada uno de los medios de transmisión:

- a) *Línea de pares paralela*: formada por dos conductores que discurren próximos y paralelamente. De bajo coste, ofrecen características de transmisión muy limitadas (cifras meramente orientativas podrían ser: tasa de transmisión < 20 Kbit/s; distancia < 50 m). Sus principales desventajas son la captación de ruido, el acoplamiento con otras líneas próximas (diafonía) y las pérdidas por radiación a frecuencias elevadas. Ejemplos de utilización: el interfaz local de datos EIA RS-232C, el bucle de abonado telefónico (último tramo).
- b) *Línea de par trenzado*: formada por dos conductores trenzados (retorcidos) sobre sí mismo. Su geometría cambiante reduce la captación de ruido, las pérdidas por radiación y la diafonía entre pares cercanos. De bajo coste, ofrece mejores características de transmisión que la línea de pares paralela (algunas cifras orientativas son: tasa de transmisión < 1 Mbit/s.; distancia < 100 m). Existen versiones de dicho tipo de línea sin pantalla (conductor cilíndrico que rodea el par), llamados *UTP (Unshielded Twisted Pair)*, y con ella, llamados *STP (Shielded Twisted Pair)*. Estos últimos ofrecen mejores características a un mayor coste. Ejemplos de utilización: la planta de abonado de la red telefónica, los cableados estructurados en edificios para comunicaciones de voz y datos.
- c) *Línea coaxial*: formada por un conductor cilíndrico y otro conductor en el interior que discurre axialmente al primero. Su geometría hace que el campo electromagnético quede prácticamente confinado en su interior, reduciéndose grandemente la captación de ruido e interferencias así como las pérdidas por radiación. El material dieléctrico existente entre ambos conductores, necesario para mantener la estructura mecánica de la línea, influye notablemente en sus características. Existen multitud de líneas coaxiales disponibles en el mercado con prestaciones que aumentan con el grosor de los conductores y la bondad del

material dieléctrico (cifras meramente orientativas podrían ser: tasa de transmisión de 10 Mbit/s a 1 Gbit/s; distancias de 200 m a varios km). Ejemplos de utilización: distribución local de señales de TV y radio en FM, enlaces de alta capacidad (troncales) en redes extensas de telefonía y datos, conexión de terminales y controladores a ordenadores grandes (en desuso), redes locales Ethernet (en desuso).

- d) *Fibra óptica*: formada por una fibra (cilindro muy delgado) de vidrio síliceo cuyo índice de refracción disminuye de forma abrupta o suave en sentido radial, desde el centro hacia la periferia. Por ella se propaga, conducida por fenómenos de reflexión o refracción, una onda electromagnética de frecuencia muy elevada (en el rango de las ondas luminosas, visibles o no) actuando como *portadora*. Existen varios tipos de fibra óptica:
1. *Multi-modo (de índice en escalón)*: constan de un núcleo y de una vaina, que rodea al primero, cuyos índices de refracción son distintos (mayor el del núcleo que el de la vaina). Para ciertos ángulos de incidencia de las ondas luminosas se produce el fenómeno de “reflexión total” en la frontera núcleo-vaina, responsable del guiado de las ondas en el sentido aproximadamente longitudinal de la fibra. En estas fibras el diámetro del núcleo es mucho mayor que la longitud de onda ( $\lambda$ ) de la onda luminosa, por lo que existen varios caminos posibles a lo largo de la fibra, es decir, múltiples modos de propagación. Esto hace que en el extremo receptor se superpongan ondas luminosas que han viajado distancias ligeramente distintas y, por tanto, con retrasos también distintos, originándose una dispersión o ensanchamiento de los pulsos de luz transmitidos que llegan a confundirse unos con otros cuando su tasa supera cierto valor (interferencia intersimbólica). Algunas de las fibras de este tipo más usuales son las denominadas 62,5/125 y las 50/125 (números que expresan, en  $\mu\text{m}$ , el diámetro del núcleo y la vaina respectivamente). Unas cifras orientativas en cuanto a prestaciones podrían ser: tasa de transmisión  $< 200$  Mbit/s; distancias  $< 2$  km.

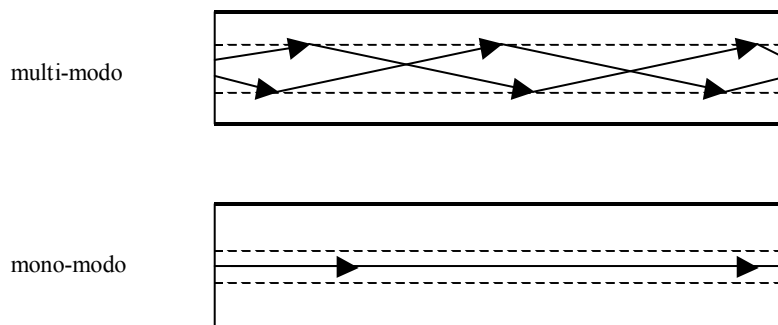


Fig. 1.3 Concepto de los modos de propagación en las fibras ópticas

2. *Mono-modo*: en estas fibras el diámetro del núcleo es de unas pocas veces la  $\lambda$  de la onda luminosa que se propaga por ellas (del orden de 8 a 12  $\mu\text{m}$ ). Esto hace que las ondas se propaguen de forma más rectilínea, básicamente por un único camino o modo de

propagación. En el extremo receptor tendremos, por tanto, un menor ensanchamiento de los pulsos que se traducirá en mayores prestaciones: tasa de transmisión de 1 a 10 Gbit/s; distancias de 10 a 50 km (cifras orientativas).

3. A medio camino entre los dos anteriores tipos de fibra tenemos las *multi-modo de gradiente de índice*. Como su denominación indica, el índice de refracción disminuye de forma suave desde el centro hacia la periferia de la fibra, siendo la luz guiada por fenómenos de refracción que tienden a curvar los haces luminosos hacia el centro de la fibra cuando aquellos se alejan de éste.

Los elementos usados para la generación y la detección de la portadora óptica son principalmente dispositivos de estado sólido (uniones semiconductoras). En el proceso de transmisión (conversión de la señal eléctrica en óptica) se utilizan:

- a) *diodos LED (light-emitting diode, diodo electro-luminiscente)*: de bajo coste, pero ofreciendo una fuente de luz de baja pureza espectral (emiten luz en un rango relativamente amplio de  $\lambda$ s), lo que aumenta la dispersión de los pulsos recibidos. Suelen emplearse con las fibras más sencillas multi-modo.
- b) *diodos LASER (Light Amplification by Stimulated Emission of Radiation, amplificación de luz mediante la emisión estimulada de radiación)*: más delicados y costosos, pero ofreciendo una fuente de luz de mayor pureza espectral (más mono-cromática). Suelen emplearse con las fibras mono-modo para obtener las prestaciones más elevadas.

Las frecuencias de emisión de los dispositivos anteriores caen en la región del espectro de luz no visible del infrarrojo, siendo la banda más usada la de  $\lambda=850$  nm. Más recientemente se vienen usando las bandas de menor frecuencia de 1350 y 1500 nm, que presentan mejores características de propagación.

En el proceso de recepción (conversión de la señal óptica en eléctrica) se usan diversos tipos de fotodiodos y fototransistores.

Los medios de transmisión de fibra óptica presentan actualmente las mejores prestaciones, en cuanto a capacidad y distancia, para el transporte de la información. Otra característica favorable es su prácticamente total ausencia de radiación y captación de cualquier energía electromagnética (eléctrica u óptica). Su principal desventaja es su coste debido, en gran parte, no a la fibra en sí sino a la complejidad de unión de las fibras (soldaduras y conectores) y a una instalación y mantenimiento más complicados por su fragilidad. Algunos ejemplos de aplicación serían: enlaces troncales de elevada capacidad y distancia en redes extensas de telefonía y datos, redes metropolitanas de difusión de video y datos, enlaces entre redes locales, etc.

En cuanto a los *medios no-guiados*, se caracterizan por un campo electromagnético de elevada frecuencia, actuando a modo de portadora de la información, que se extiende libremente por todo el medio (el espacio o éter) en el que se encuentran inmersos el emisor y el receptor. Un limitado confinamiento –guiado– de dicho campo se consigue por medio de los dispositivos de acoplo al medio (antenas o lentes).

Algunas características específicas de este tipo de medio de transmisión son que existe un *único* medio que debe ser compartido por todos los procesos de transmisión que se desarrollan en una cierta zona

espacial que, a su vez, no tiene límites claramente definidos; la transmisión por este tipo de medios está sujeta a mayores problemas (de ruido, interferencia, atenuación, dispersión por caminos múltiples, etc.) que cualquier otro medio; su capacidad está limitada, es decir, no puede aumentarse sencillamente como sucede en los otros tipos de medio (para los que siempre pueden instalarse más líneas de par trenzado o fibras ópticas, por ejemplo); en relación con lo anterior, la capacidad disponible debe repartirse cuidadosamente entre las diferentes transmisiones asignándoles fracciones del espectro y del espacio difíciles de planificar y mantener.

Existen infinidad de ejemplos de aplicaciones de este tipo de medios:

- a) Ondas de radio con frecuencias por debajo de 1GHz: difusión de información de audio y vídeo, telemetría/telemando, servicios de voz y datos en sistemas de telefonía móvil celular, etc.
- b) Ondas de radio por encima de 1 GHz: enlaces punto a punto y de difusión de información, tanto vía satélite como terrestres, redes locales sin hilos, etc.
- c) Ondas en el rango de frecuencias óptico: enlaces entre puntos con visibilidad directa entre ellos de corta y media, redes locales sin hilos, telemando, etc.

#### 1.1.4 El multiplexado de la señal

El Multiplexado es el método por el cual pueden transmitirse varias señales (*canales*) por un único medio físico de transmisión. Existen básicamente dos métodos:

- a) *FDM (Frequency División Multiplexing)*: multiplexado por división en la frecuencia. Las diversas señales (analógicas o digitales), que deben ser de banda limitada, *modulan* portadoras distintas suficientemente separadas en frecuencia y se transmite la suma de todo ello.

Como se vio, la modulación supone una traslación del espectro de las señales. De este modo las distintas señales, aunque simultáneas en el tiempo, quedan separadas desde el punto de vista frecuencial.

La recuperación, a la recepción, de las distintas señales originales (el *demultiplexado*) se lleva a cabo mediante filtros paso-banda y demoduladores. Puede decirse que las técnicas de FDM son básicamente analógicas y que conllevan cierto coste y complejidad (la integración a gran escala de circuitos analógicos es más costosa que la de circuitos digitales, los filtros no son ideales, etc.).

Algunos ejemplos de utilización de técnicas de FDM son: la inmensa mayoría de las comunicaciones que se llevan a cabo sobre medios “no-guantes” (por ejemplo, difusión de radio y TV, comunicaciones móviles de telefonía, mensajería, etc.); también se utiliza en medios “guantes” (por ejemplo, las redes de cable de distribución de TV, algunas redes locales realizadas sobre cables coaxiales de banda ancha, antiguamente, las conversaciones telefónicas a través de cables submarinos, etc. y, más recientemente, las técnicas de *WDM (Wavelength División Múltiple)*, es decir, el uso de varias portadoras

luminosas (colores), utilizadas para aumentar la capacidad, ya de por sí elevada, de las fibras ópticas.

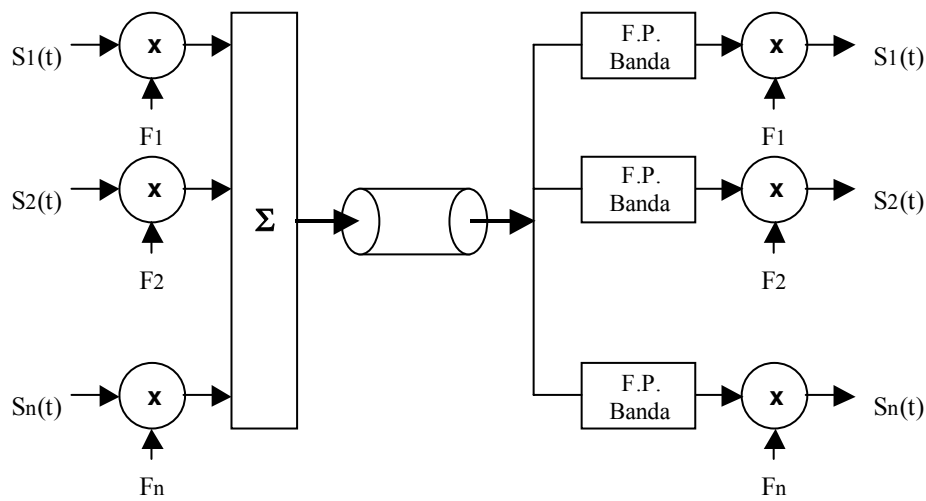


Fig. 1.4 Esquema del proceso de multiplexado FDM

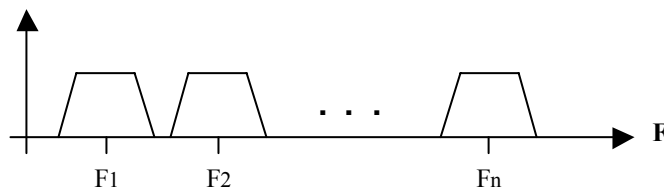


Fig. 1.5 Separación en la frecuencia de las señales multiplexadas mediante FDM

- b) *TDM (Time División Multiplexing)*: multiplexado por división en el tiempo. Las diversas señales (digitales) se “trocean” en grupos de bits o bytes que se transmiten a mayor velocidad que la original, unos tras otros y siguiendo un proceso cíclico. En este caso cada una de las señales utiliza todo el ancho de banda disponible en el medio de transmisión pero nunca coincidiendo en el tiempo. La estructura de bits que transporta un “trozo” de cada una de las señales a multiplexar, y que se repite cíclicamente en el tiempo, se denomina *trama*. Dentro de ella, el grupo de bits perteneciente a una misma señal se denomina *ranura (slot)*. Dado que una trama debe contener los “trozos” de información de todos los  $N$  canales a multiplexar en el mismo tiempo original, ésta tendrá una velocidad (en bit/s)  $N$  veces superior al de las señales originales (en realidad algo superior

dado que las tramas suelen incluir bits adicionales para la sincronización, la supervisión y el control de errores). Puede decirse que las técnicas de TDM son básicamente digitales y para señales digitales, y su coste y complejidad son reducidos (dado el menor coste de los circuitos integrados digitales). Un ejemplo de utilización de técnicas de TDM se encuentra en la inmensa mayoría de los enlaces troncales de alta velocidad en redes de área extensa de telefonía y datos y el acceso a la RDSI.

Atención especial merece el esquema de digitalización y multiplexado de los canales de voz en las redes telefónicas: los canales analógicos de voz, con un ancho de banda de aproximadamente 3,4 kHz, se digitalizan a 8.000 muestras/s, codificándose cada muestra con 8 bits; esto nos da el canal digital básico de 64 kbit/s de toda red telefónica y, por extensión, de numerosas redes de datos. En los enlaces troncales el primer nivel de multiplexado (temporal) se realiza tomando 32 canales de 64 kbit/s cada uno (de los que 30 se usan para tráfico de voz y los 2 restantes para funciones de señalización y supervisión). Esto constituye el llamado enlace E1, cuya velocidad es de 2,048 Mbit/s (de uso en Europa y otras partes del mundo excepto en los EUA, Canadá y Japón).

El *acceso básico* a la RDSI, también llamado 2B+D, está construido por el multiplexado de 2 canales B (para tráfico de voz y datos) de 64 kbit/s cada uno y un canal D (para señalización usuario-red y datos a baja velocidad) de 16 kbit/s. El *acceso primario* a la RDSI se realiza alrededor de un enlace E1 con algunas modificaciones.

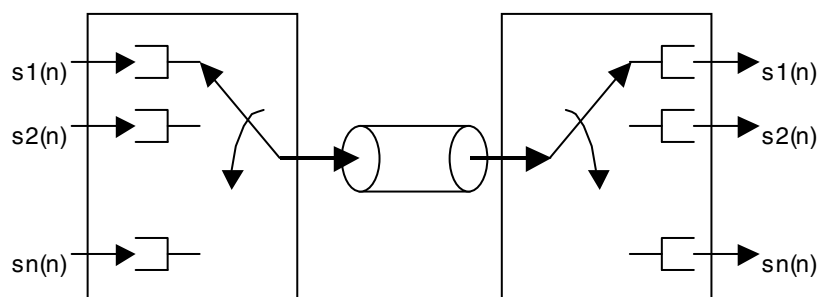


Fig. 1.6 Esquema del multiplexado mediante TDM

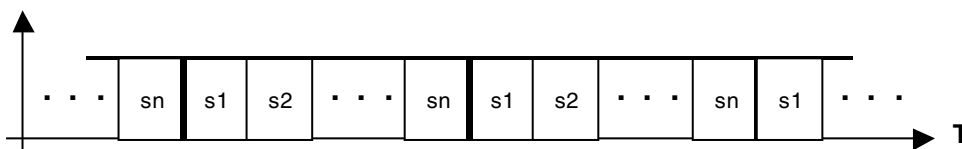


Fig. 1.7 Separación en el tiempo de las señales multiplexadas mediante TDM

### 1.1.5 Clasificación de la transmisión según su sentido

En función de los parámetros “sentido” de una comunicación y tiempo, las transmisiones pueden clasificarse en:

- a) *Simplex*: cuando se dan siempre en un solo sentido; por ejemplo, aquellas asociadas a los servicios de difusión (TV, radio) y de mensajería (busca-personas).
- b) *Semi-duplex*: cuando pueden darse en ambos sentidos pero no al mismo tiempo; por ejemplo, las comunicaciones entre radioaficionados y las de los *walkie-talkies*.
- c) *Full-duplex*: cuando pueden darse en ambos sentidos y simultáneamente en el tiempo; por ejemplo, la telefonía (convencional y móvil).

### 1.1.6 Transmisión síncrona/asíncrona

Considerando la transmisión de caracteres (bytes), se dice que la transmisión es asíncrona cuando éstos pueden transmitirse en cualquier instante, aisladamente o en grupos, y no hay ninguna coordinación temporal estricta entre emisor y receptor (para ello cada carácter lleva asociada información de autosincronización, que puede consistir en los llamados *bit de inicio* –usualmente un bit “0”- y *bit de parada* –usualmente uno o más bits “1”).

Por el contrario, la transmisión es síncrona cuando siempre existe una coordinación temporal precisa entre emisor y receptor y los caracteres se transmiten contiguamente sin ningún bit adicional de sincronización (aun en el caso de que no hayan caracteres a transmitir, la sincronía se mantiene y es frecuente que se envíen continuamente bits llamados de “relleno”).

### 1.1.7 Capacidad de canal

La capacidad de canal es una medida de la cantidad de información que puede llevar un canal de comunicaciones; por ejemplo, el ancho de banda (en Hz) constituiría una medida analógica y la tasa de bit (en bit/s) una medida digital. Detrás de este concepto existe un aparato científico-tecnológico muy importante y extenso (cuya presentación ocuparía más de un libro como el presente). Se darán, por tanto, unas breves “pinceladas” sobre dicho tema.

Los parámetros relacionados con el concepto de capacidad de canal son:

$R$ , velocidad de transmisión de datos (en bit/s)

$B$ , ancho de banda de la señal transmitida (en Hz) que, para aprovechar el canal, se hace coincidir con el de éste

$N$ , potencia de ruido (en W), usualmente considerado de tipo gaussiano (térmico, aleatorio, de espectro de potencia plano)

$P_e$ , probabilidad de error de bit

$R_m$ , velocidad de codificación/modulación (en baudios o símbolos/s); está relacionada con  $R$  de la siguiente forma  $R = R_m \log_2 A$ , donde  $A$  es el número de símbolos o señales distintas (formas de onda, niveles de tensión, etc.) que tiene el esquema de codificación/modulación empleado. En general, si se desean transmitir  $n$  bits en cada símbolo se requiere  $A=2^n$  símbolos.

Deben mencionarse algunas propiedades relacionadas con el concepto de capacidad de canal:

Idealmente, por un canal de ancho de banda  $B$  Hz puede transmitirse a  $2B$  baudios (utilizando, por ejemplo, *pulsos paso-bajo*).

En total ausencia de ruido, por un canal de ancho de banda  $B$  finito y empleando señales de potencia finita puede transmitirse a velocidad arbitrariamente alta ( $R \rightarrow \infty$ ) usando un esquema de codificación con muchos niveles ( $A \rightarrow \infty$ ).

*Teorema de Nyquist:* Dada una señal analógica de ancho de banda  $B$  que deba ser digitalizada para su transmisión, es suficiente que sea muestreada a  $2B$  muestras/s para poder restituirla con toda fidelidad. Por ejemplo, las señales de voz del servicio de telefonía, que no pretende ser de alta fidelidad, sino simplemente de alta inteligibilidad, son usualmente muestreadas a 8 Kmuestras/s lo que teóricamente garantiza que dichas señales puedan tener un ancho de banda de hasta 4 kHz.

*Relación de Shannon:*  $C = B \log_2 \left( 1 + \frac{S}{N} \right)$  (bits/s). Esta importante relación nos indica la capacidad máxima de un canal de comunicación de ancho de banda  $B$  en presencia de ruido gaussiano (de potencia  $N$ ) y empleando señales de potencia  $S$ . Dicha relación es una cota superior para cualquier esquema de codificación/modulación realizable actualmente o en el futuro, por sofisticado que sea.

Véase un ejemplo de aplicación de la anterior ley a la determinación de la capacidad máxima de transmisión digital del canal telefónico convencional usando módems: las frecuencias utilizables en el canal telefónico van aproximadamente desde 300 a 3.400 Hz, es decir, su ancho de banda  $B$  es de 3,1 kHz; la relación señal a ruido en dicho canal es del orden de 35 dB; por tanto su capacidad máxima, en este ejemplo, sería  $C \approx 36.044$  bit/s. Debe indicarse que en esta expresión, el valor de  $S/N$  debe indicarse en valores lineales, esto es,  $35 \text{ dB} = 3162,28$ ).

## 1.2 Conceptos básicos de redes de comunicaciones

Una *red de comunicaciones* es un conjunto de medios de transmisión y conmutación para el envío de información entre puntos separados geográficamente. Esta definición resulta extremadamente general y en la actualidad existe un gran número de implementaciones diferentes que responden a necesidades específicas, tales como redes de acceso de datos, troncales, inalámbricas, redes de voz, etc.)

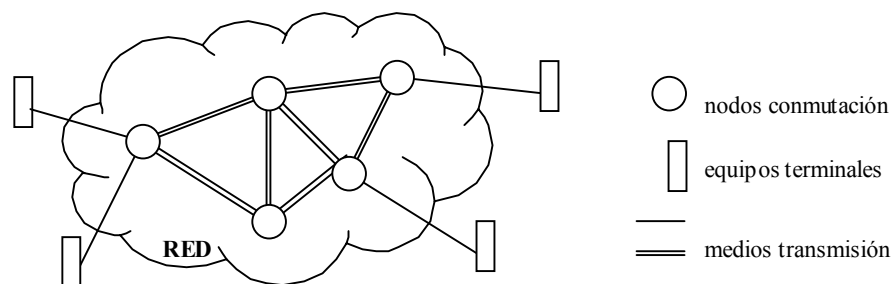


Fig. 1.8 Elementos de las redes y sistemas de comunicaciones

Un *sistema de telecomunicación* es un conjunto de medios físicos, lógicos y organizativos que permiten ofrecer un servicio de telecomunicación; es decir, incluye a la red, los terminales y los recursos técnicos y humanos adicionales para su mantenimiento, gestión y explotación.

Un *servicio de telecomunicación* es el valor añadido que percibe el usuario; por ejemplo: telefonía, facsímil (fax), transmisión datos (módem) entre computadores. Obsérvese la diferencia entre servicio y red; los tres servicios citados anteriormente son habitualmente soportados por la red telefónica tradicional (aunque esta fuera inicialmente diseñada y optimizada para soportar el servicio de telefonía) y cabría pensar en redes más adecuadas para soportar los dos últimos servicios. En relación con los servicios, en ocasiones se habla de:

- a) *Teleservicios*: coincidiendo básicamente con la definición genérica de servicio que se ha dado.
- b) *Servicios portadores*: aquellos que se ofrecen entre puntos de acceso a la red, no entre terminales. Retomando los ejemplos anteriores, puede decirse que el servicio portador de la red de telefónica tradicional es aquel que permite el envío, entre puntos situados prácticamente en cualquier lugar del globo terráqueo, de señales analógicas en el rango de frecuencias comprendido entre 300 y 3.400 Hz, en modo *full-duplex*, con retardo inferior a unos 200 ms y con una distorsión y adición de ruido determinados. Evidentemente este servicio portador es el más adecuado para ofrecer el teleservicio de telefonía aunque, con el uso de módems, se adapte al ofrecimiento de otros teleservicios, como se ha comentado anteriormente.
- c) *Servicios suplementarios (o de valor añadido)*: no son servicios en sí mismos, sino más bien mejoras o complementos de algún teleservicio. Por ejemplo, todas las facilidades de marcación abreviada, desviación de llamadas, cobro revertido, indicación de llamada en espera, conferencia a tres, identificación del llamante, etc., asociadas al teleservicio de telefonía constituyen servicios suplementarios.

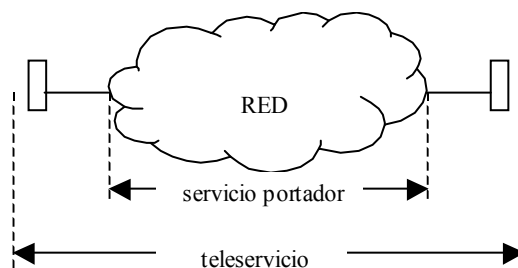


Fig. 1.9 Servicio portador y teleservicio

### 1.2.1 Servicios orientados y no orientados a conexión

Los servicios también pueden clasificarse según la forma en que se establecen y finalizan. Así puede hablarse de:

- a) servicios *orientados a conexión*: aquellos que requieren una fase de establecimiento (antes de estar disponibles) y una de finalización. Por ejemplo, el servicio de telefonía requiere de una fase de establecimiento (en la que se especifica la dirección del destino, se analiza la viabilidad de establecer el servicio, se reservan los recursos necesarios en la red, se notifica al destinatario, etc.). A esta fase le sigue la de transferencia (en la que se ofrece el servicio propiamente) y, después, la fase de liberación (en la que se desocupan los recursos reservados en la red, se almacenan los datos de facturación del usuario, etc.).
- b) servicios *no orientados a conexión*: aquellos que están disponibles sin ningún procedimiento previo y carecen de fases. Por ejemplo, las redes de datos IP (*Internet Protocol*), donde los *datagramas*, conteniendo la dirección de destino, se envían por la red sin ningún proceso previo ni posterior.

### 1.2.2 Tipos de redes según su capacidad de cobertura

Aunque puedan establecerse multitud de criterios, las redes se clasifican tradicionalmente según tres parámetros: velocidad de acceso, distancia cubierta y tipo de propiedad/explotación. Así tenemos:

- a) *WAN (Wide Area Networks)*: Las redes de área extensa comúnmente se caracterizan por tener una velocidad de acceso moderada (de 1 a 64 kbit/s; o hasta 2 Mbit/s.), distancias cubiertas grandes (de 100 a 20.000 km) y propiedad pública (en el sentido de que los territorios por donde se despliega la red son varios, de propietarios distintos, incluidas las administraciones públicas. Antes del fenómeno de la “liberalización” de hace algunos años en España, la propiedad y explotación eran enteramente públicas). Ejemplos de este tipo de redes son: la red telefónica tradicional y la RDSI, las redes públicas de datos con estándar de acceso X.25 (en retroceso), internet, etc. En algunos casos, se comprende también a los

troncales de red, que pueden alcanzar velocidades de los centenares de Gbit/s mediante tecnologías tales como DWDM.

- b) *LAN (Local Area Networks)*: Las redes de área local tradicionales se caracterizan por tener velocidades de acceso elevadas (de 0,2 a 16 Mbit/s; o hasta 1000 Mb/s si se incluyen las variantes más recientes), distancias cubiertas reducidas (de 200 m a 5 km) y propiedad/explotación privada (en el sentido de que la infraestructura de red usualmente pertenece a la organización a la que da sus servicios, se despliega en territorios de su propiedad y es explotada por la misma). Ejemplos de este tipo de redes son: *Ethernet* (IEEE 802.3), *Token Ring* (IEEE 802.5), *AppleTalk* (norma propietaria en declive), etc.
- c) *MAN (Metropolitan Area Networks)*: Las redes de área metropolitana se caracterizan por tener velocidades de acceso muy elevadas (de 30 a 150 Mbit/s y en la actualidad hasta los 10 Gbit/s), distancias cubiertas medianas (10 a 50 km, las correspondientes a una ciudad y su área de influencia) y propiedad/explotación a medio camino entre lo público y lo privado. Dichas redes, aparecidas en la década de 1990, eran tecnológicamente avanzadas y su lugar actualmente es algo difuso, confundiéndose con las nuevas redes de área local y extensa de alta velocidad. Ejemplos de este tipo de redes son: *DQDB (Dual Queue Dual Bus)* (IEEE 802.6), *FDDI (Fiber Distributed Data Interface)*, etc.

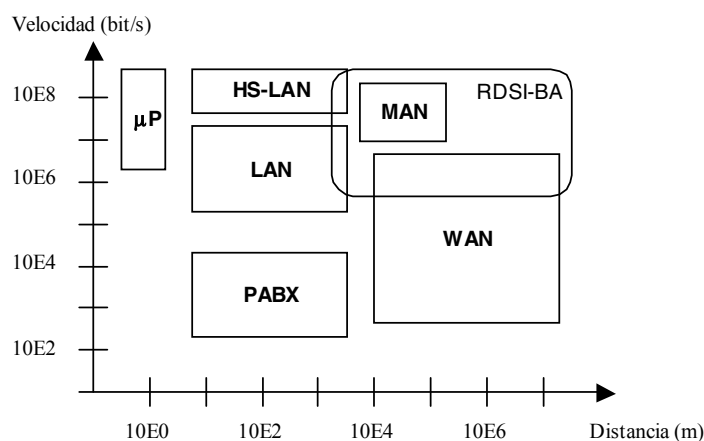


Fig. 1.10 Ámbito de aplicación de los distintos tipos de redes según los parámetros de velocidad y cobertura

### 1.2.3 Clasificación de las topologías de red

La *topología (de red)* es la disposición lógica de los elementos (enlaces, nodos) de una red. Así pueden definirse diversos modelos de topologías básicas:

- a) *Malla*: Los distintos nodos están más o menos densamente unidos entre sí por enlaces directos (en general, de forma arbitraria y sin seguir ninguna jerarquía particular). Cuando cualquier

nodo está unido directamente a todos los demás mediante un enlace directo, se dice que la red presenta una topología de *mallá completa*.

- b) *Estrella*: Los distintos nodos están unidos a un único nodo central.
- c) *Árbol*: Los distintos nodos están distribuidos en forma de ramificaciones sucesivas a partir de un único nodo raíz.
- d) *Bus*: Todos los nodos están unidos por un único enlace común.
- e) *Anillo*: Los nodos están unidos en cadena, uno tras otro, cerrándose ésta sobre si misma (de manera circular).

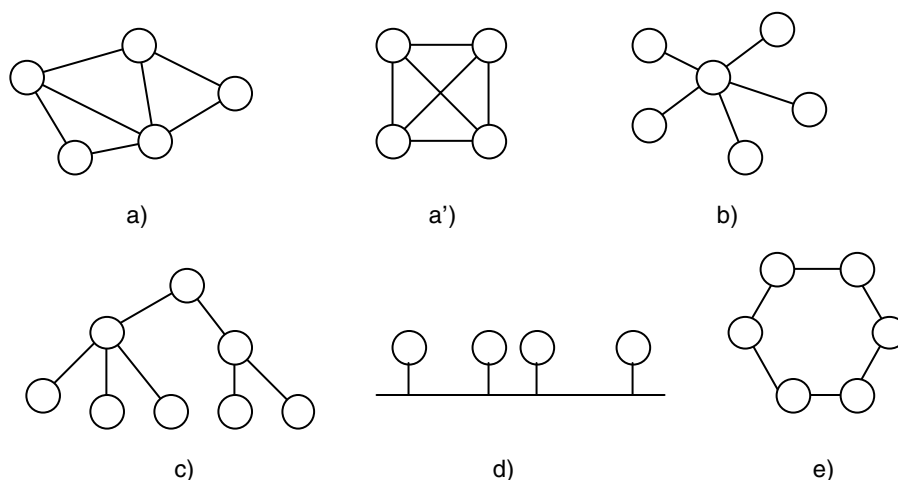


Fig. 1.11 Topologías de red a) malla, a') malla completa de 4 nodos, b) estrella, c) árbol, d) bus, e) anillo

#### 1.2.4 Clasificación de los tipos de conmutación de datos

La conmutación permite la entrega de información desde un nodo origen hasta un destino a través de un medio compartido, gracias a los nodos intermedios que actúan como elementos activos en el encaminamiento de la información, dirigiendo los datos. Existen diversas técnicas para llevar a cabo esta tarea, como son la conmutación de circuitos, de paquetes y de celdas, así como variantes específicas que a continuación son descritas.

- a) *Conmutación de circuitos*: Modo de operación de una red en el que la comunicación entre dos terminales se produce a través de caminos establecidos al inicio de la comunicación, que no varían durante ésta y que se dedican en exclusiva a ella. Los caminos se construyen al inicio de la comunicación, por conmutación electro-mecánica o electrónica en los nodos intermedios, concatenando una serie de circuitos físicos (por ejemplo, una línea de par trenzado dentro de un cable de pares, un canal en un sistema de FDM o TDM, un canal de radiofrecuencia en el

espacio o un canal luminoso en una fibra óptica). Las características más relevantes de las comunicaciones establecidas en este modo de operación son: a) su ancho de banda ( $B$ ), o velocidad de transmisión ( $R$ ), es fijo; b) su retardo ( $T$ ) es bajo y constante. Dada la reserva exclusiva de recursos de red que implica, la conmutación de circuitos resultará rentable cuando se éstos aprovechen enteramente, es decir, para flujos continuos de datos. El ejemplo más claro e importante de este modo de operación lo constituye la red telefónica bajo los estándares del ITU-T (con sus circuitos digitales full-duplex a 64 kbit/s dedicados a cada llamada telefónica).

- b) *Conmutación de paquetes*: Modo de operación de una red en el que la comunicación entre dos terminales se produce mediante los paquetes (unidades de información formadas por grupos de bits) que viajan a través de caminos, establecidos inicialmente o no, fijos o variables a lo largo de la comunicación, compartidos por paquetes de otras comunicaciones. El camino que sigue cada paquete se realiza, tras su análisis, por su conmutación digital en los nodos intermedios. Las características más relevantes de las comunicaciones establecidas en este modo de operación son: a) su ancho de banda ( $B$ ) no está generalmente prefijado, es decir, la red proporciona capacidad según los paquetes asociados a cada comunicación vayan llegando (según su demanda) y en función de la disponibilidad de recursos; b) su retardo ( $T$ ) es variable y mayor que en la conmutación de circuitos. Esta variabilidad en  $B$  y  $T$  es debida a que no se reservan recursos en exclusiva a las comunicaciones; por lo tanto, los tiempos de procesamiento y conmutación y la capacidad disponible varían continuamente en función de las comunicaciones y paquetes que existen en un momento dado. Algunos ejemplos de este modo de operación se encuentran en las redes de datos públicas o privadas que operan bajo estándares como X.25 (o *Frame Relay*) del ITU-T, TCP/IP de la comunidad internet, etc.

Existen dos grandes familias dentro de la conmutación de paquetes clásica:

1. Conmutación de paquetes en *modo "circuito virtual"*: El camino que siguen todos los paquetes pertenecientes a una comunicación se determina en su inicio y permanece invariable a lo largo de la comunicación como en la conmutación de circuitos. Aun así, sólo se determina el camino, no se hace una reserva de recursos (al menos no en exclusiva) como sucedía en la conmutación de circuitos. De ahí el nombre de circuito virtual.
  2. Conmutación de paquetes en *modo "datagrama"*: El camino que sigue cualquier paquete se determina individualmente, es decir, se determina en el momento en que un paquete llega a un nodo intermedio y solamente tiene validez para ese paquete en particular. De hecho, a nivel de la red, no existe la noción de comunicación (entendida como una asociación lógica en el tiempo entre dos terminales). Dicho de otro modo, cada paquete (que lleva toda la información necesaria para su "viaje" a través de la red) constituye en sí una comunicación.
- c) *Conmutación de celdas*: Es una variante evolucionada de la conmutación de paquetes en la que éstos son de longitud pequeña y fija (y usualmente llamados celdas). Su pequeña longitud permite una "granularidad" muy fina a la hora de asignar recursos a las comunicaciones y, por tanto, un buen aprovechamiento del ancho de banda. El tamaño fijo de las celdas facilita el uso de técnicas de conmutación muy rápidas (realizadas directamente por *hardware*). El ejemplo más relevante de este modo de operación se encuentra en las redes de banda ancha basadas en la tecnología ATM (*Asynchronous Transfer Mode*) (ver más detalles en el capítulo 6).

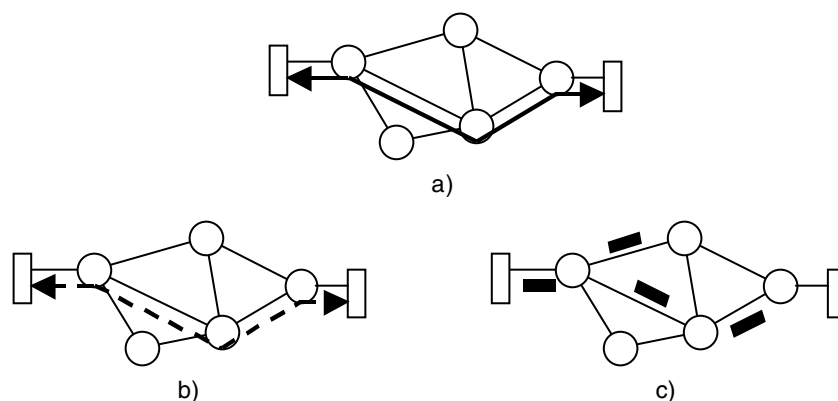


Fig. 1.12 Distintos modos de operación de las redes: a) conmutación de circuitos, b) conmutación de paquetes en modo circuito virtual, c) idem, en modo datagrama

Tabla 1.1 Algunas características y diferencias de los distintos modos de conmutación en las redes

Conmutación de CIRCUITOS	Conmutación de PAQUETES modo DATAGRAMA	Conmutación de PAQUETES modo CIRCUITO VIRTUAL
Circuito dedicado en exclusiva	Circuito no dedicado (compartido)	Circuito no dedicado (compartido)
Ancho de banda Fijo	Uso dinámico del ancho de banda	Uso dinámico del ancho de banda
Retardo de establecimiento de la conexión	No hay retardo de establecimiento de la conexión	Retardo de establecimiento de la conexión
Retardo bajo y fijo (no hay almacenamiento, solo propagación)	Retardo mayor y variable (por almacenamiento y conmutación)	Retardo mayor y variable (por almacenamiento y conmutación)
Ruta establecida inicialmente e invariable	Ruta establecida para cada paquete	Ruta establecida inicialmente e invariable
Fiabilidad alta (sólo desconexiones imprevistas)	Fiabilidad baja (posibles pérdidas y desordenamientos de paquetes)	Fiabilidad alta (sólo desconexiones imprevistas)
No existen cabeceras de red (durante la conexión)	Cabeceras de red grandes en cada paquete	Cabeceras de red más pequeñas en cada paquete
Para tráfico continuo	Para tráfico discontinuo (a ráfagas)	Para tráfico discontinuo (a ráfagas)

### 1.2.5 Clasificación del tráfico en clases

Con el único fin de perfilar los distintos tipos de tráfico que puedan ser transportados por una red, se pueden establecer una posible clasificación con las siguientes 3 clases de tráfico:

1. *Clase I (tiempo real)*: de mensajes largos o continuos, en tiempo real (no tolera retardos superiores, digamos, a 200 ms), admite errores, admite bloqueo, muy interactivo: hombre-hombre (telefonía, videoconferencia), adecuado para la conmutación de circuitos.
2. *Clase II (interactivo, racheado)*: de mensajes cortos y racheados (a ráfagas), interactivo (tolera cierto retardo, de 1 a 3 s.), no suele admitir errores, admite cierto bloqueo, interactivo: hombre-máquina, máquina-máquina (terminal de datos, cajero electrónico, navegación web), adecuado para la conmutación de paquetes.
3. *Clase III (pesado, diferido)*: de mensajes muy largos, no requiere tiempo real (tolera retardos elevados, hasta de minutos), no admite errores, no admite bloqueo, no interactivo: máquina-máquina (transferencia de archivos, correo electrónico), adecuado para la conmutación de paquetes o circuitos.

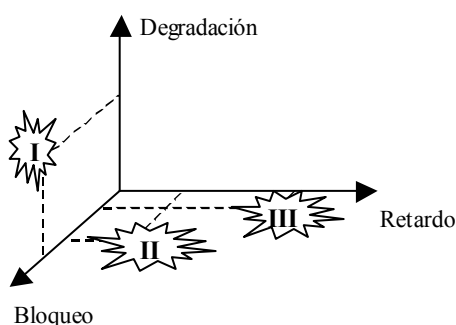


Fig. 1.13 Representación simbólica de las clases de tráfico según los parámetros de degradación, retardo y bloqueo (imposibilidad de establecer una conexión en un momento dado)

### 1.3 Conceptos básicos de teletráfico

La *intensidad de tráfico*, o *tráfico* simplemente, es una medida muy útil de la carga o grado de ocupación de los recursos (línea de transmisión, órgano de conmutación, etc.) de una red.

Tres definiciones alternativas de este concepto son:

1. Ocupación media de un recurso en el tiempo (enlace de comunicaciones, canal en un sistema FDM, ranura en un sistema TDM, elemento de un nodo de conmutación, etc.).
2. Número medio de recursos ocupados (de un grupo de ellos: líneas en un cable de pares, procesadores en un sistema, etc.)
3. Dado un recurso que es solicitado por sus usuarios con una frecuencia promedio de  $\lambda$  peticiones/s y siendo el tiempo medio de ocupación de dicho recurso en cada solicitud de  $T$  s, se define el tráfico ( $A$ ) cursado por dicho recurso como

$$A = \lambda T \quad (1.1)$$

Dicha cantidad es adimensional aunque, en honor al ingeniero A. K. Erlang, que desarrolló la teoría de teletráfico a principios del siglo XX, se suele expresar en Erlang (Er). Véanse con algunos ejemplos el significado de esta importante medida:

- a) Después de observar la actividad en una línea de comunicaciones, concluimos que ésta se encuentra ocupada el 40% del tiempo. Entonces dicha línea soporta un tráfico de 0,4 Er.
- b) Los proveedores del servicio de telefonía manejan estadísticas acerca de la actividad de sus abonados con el fin de dimensionar sus redes. Así, por ejemplo, se dice que el tráfico ofrecido por un usuario doméstico es de unos 0,02 Er (es decir, la proporción del tiempo, a lo largo de un día promedio, en el que el teléfono de dicho usuario estaría activo en llamadas, suponiendo que todas son realizadas). En ese mismo sentido, el tráfico ofrecido por una cabina telefónica situada en la vía pública es de 0,1 a 0,4 Er.
- c) El tráfico cursado a través de un cable de pares trenzados de 100 pares es de 25 Er. Entonces puede decirse que 25 pares están ocupados el 100% del tiempo y el resto desocupados o también que todos los pares están ocupados en un 25% de su tiempo.
- d) En una central local de telefonía se reciben 2.100 peticiones de llamadas por hora (en la hora del día con mayor tráfico u *hora cargada*, *busy hour*). Si el promedio de la duración de una llamada telefónica es de 3 min, puede decirse que la central soporta un tráfico de

$$A = \lambda T = \frac{2100 \text{ llamadas}}{1 \text{ h}} \frac{1 \text{ h}}{60 \text{ min}} \frac{3 \text{ min}}{1 \text{ llamada}} = 105 \text{ Er}$$

Obsérvese que el *tráfico cursado* es aquel que supone una ocupación real de los recursos. El *tráfico ofrecido* es aquel que se ofrece a un grupo finito de recursos, pero sin que haya la certeza de que todas las peticiones puedan satisfacerse (cursarse) en cualquier momento; es, por tanto, aquel que hipotéticamente se cursaría si el número de recursos fuera infinito. El *tráfico perdido* es aquel que se debe a todas aquellas peticiones que no han encontrado algún recurso libre y, por tanto, han sido rechazadas. Finalmente, también se define el *tráfico demorado* como aquel que se debe a todas aquellas peticiones que no han encontrado algún recurso libre inmediatamente y, por tanto, son mantenidas en espera hasta que eso suceda. Es aquel tráfico que acaba materializándose, pero con demora.

El *caudal (throughput)* es un término equivalente al de tráfico cursado, normalmente usado en el entorno de redes de área local y denotado por la letra  $S$ . Por ejemplo, un analizador de protocolos conectado a una red Ethernet que indique un caudal de  $S=0,35$  quiere decir que la ocupación promedio del bus en transmisiones es del 35% del tiempo o que el tráfico que cursa es de 0,35 Er.

Una definición alternativa del caudal es

$$S = \frac{\{\text{bits de datos transm. / s}\}}{R} \quad (1.2)$$

donde  $R$  es la tasa de transmisión en bruto en el canal.

## 2. Arquitecturas de comunicaciones

### 2.1 Perspectiva histórica

La idea de red de comunicaciones como infraestructura para ofrecer algún servicio de comunicación de información quizás se remonte muchos siglos atrás. Las redes basadas en el fenómeno de la electricidad se remontan a mediados del siglo XIX, con la invención del telégrafo. Pero las redes de comunicación de datos, digitales, surgen a partir de la invención del ordenador moderno, hacia la década de 1960 (aunque existieron máquinas de calcular bastante antes: el ábaco de los árabes, la máquina de Leibniz y ya en el siglo XX, el computador ENIAC, basado en válvulas de vacío).

Los primeros ordenadores con transistores eran equipos grandes y costosos en posesión de unas pocas empresas (los llamados *centros de cálculo*). Ofrecían los servicios a clientes que se desplazaban físicamente para entregar los datos de entrada y recoger los datos de salida. Es decir, al principio el acceso a los ordenadores era *local*, a través de terminales relativamente rudimentarios conectados a ellos (lectores de tarjetas perforadas con los programas y datos, impresores en papel continuo, teclados y pantallas de rayos catódicos, a lo sumo).

Con el fin de mejorar los servicios ofrecidos a sus clientes, los centros de cálculo habilitaron terminales en las ubicaciones de aquéllos para permitir el acceso *remoto*, usando módems y la red analógica de telefonía (la única realmente extendida en aquella época). Con el tiempo los ordenadores empezaron a conectarse entre sí para compartir datos y capacidad de proceso entre ellos, así como terminales de entrada/salida más sofisticados.

Ya tenemos *las redes de ordenadores* o de comunicación de datos. La conectividad entre ordenadores y terminales requirió la adición de *hardware* (los llamados *front-end* de comunicaciones) y *software* (los protocolos de comunicación). Cada fabricante de ordenadores escogió su propio camino, es decir, escogió su propia *arquitectura de comunicaciones*. Pero antes de definir con más exactitud eso, se hará un breve repaso a la historia de las redes de datos.

Postrimerías de la década de 1960: En EUA, ARPA (*Advanced Research Projects Agency*) promueve el desarrollo de una red específica para la comunicación entre los ordenadores de centros federales militares y de investigación. Dicha red debía ser robusta ante fallos de algunos de sus elementos (topología en *malla* con caminos redundantes) y adaptada al tráfico generado por los ordenadores (el *paquete* como unidad de información). Esa red, llamada *ARPANet*, fue el embrión de lo que hoy conocemos como *Internet*.

Principios de la década de 1970: IBM presenta su arquitectura de comunicaciones SNA (*System Network Architecture*). Poco después Digital presenta la suya DNA (*Digital Network Architecture*); Xerox, en sus laboratorios de Palo Alto (California, EUA) desarrolla una red de ámbito local y topología en *bus* bautizada como *Ethernet* para conectar estaciones de trabajo a elevada velocidad; en Canadá, DATAPAC significa una de las primeras redes públicas de datos.

Principios de la década de 1980: IBM, en sus laboratorios de Zurich (Suiza), desarrolla la red *Token Ring*, otra red de ámbito local con topología en anillo y filosofía de acceso diferente al de la red *Ethernet*. En 1985, la red SITA desarrollada por y para las compañías de transporte aéreo está ampliamente extendida (algunos datos de explotación correspondientes al año 1985 eran: cuenta con enlaces a 9,6 y 14,4 kbit/s, algunos de ellos vía satélite, conecta unas 250 compañías aéreas de unos 150 países, mueve  $5 \times 10^9$  mensajes, relacionados con reservas de vuelos en tiempo real, de 80 octetos de longitud media al año con un retardo típico de 3 s, mueve también  $4 \times 10^8$  mensajes, relacionados con partes de vuelo y meteorología, de 200 octetos de longitud media al año con un retardo mayor).

En España: A principios de la década de 1970 la CTNE (Compañía Telefónica Nacional de España) presenta su red pública de datos IBERPAC; la compañía influye de forma notable en la definición, a nivel internacional en el CCITT, en la definición del estándar X.25 para dicho tipo de redes; en 1982 la red sufre un cambio de tecnología con la adopción de equipos propios de fabricación nacional (los sistemas TESYS); algunas cifras de explotación al año 1983 eran: velocidades de acceso de los terminales en modo paquete de 300 a 1200 bit/s, 24 centros de conmutación, 103 concentradores de tráfico, 1 centro de gestión de red, 16.300 terminales (con un crecimiento anual de 2000 terminales/año), tráfico total de unos 800 paquetes/s, el 80% del tráfico se da entre las 9 y 15 horas del día, el tiempo medio de transferencia de los paquetes es  $< 1$  s para el 86% de ellos y  $< 2$  s para el 95% de ellos.

Año 1978: La ISO (*International Standards Organization*) promueve el desarrollo de un *modelo de referencia* para el desarrollo de una arquitectura de comunicaciones para la *interconexión de sistemas abiertos* (OSI: *Open Systems Interconnection*).

Mediados de la década de 1990: “Explosión” de Internet y de las comunicaciones móviles celulares.

## 2.2 Conceptos de arquitecturas de comunicaciones

Las arquitecturas de comunicaciones permiten ordenar la estructura necesaria para la comunicación entre equipos mediante una red de modo que puedan ofrecerse servicios añadidos al simple transporte de información, algunos tan importantes como la corrección de datos o la localización del destinatario en un medio compartido. En esta arquitectura, deben definirse ante todo algunos conceptos esenciales:

- a) *Proceso de aplicación*: cualquier proceso (programa de aplicación en ejecución) en un sistema informático que ofrezca alguna utilidad al usuario.
- b) *Sistema final*: sistema informático donde residen procesos de aplicación; en ciertos contextos se le llama acertadamente *host* (anfitrión). Son los antiguamente llamados *mainframes*, las estaciones de trabajo, los PCs, etc.

- c) *Sistema intermedio*: sistema que, en general, no posee aplicaciones de usuario y actúa como nodo de conmutación e interconexión en las redes; son los repetidores, puentes (*bridges*) y encaminadores (*routers, gateways*) cada uno de ellos con funcionalidades específicas.
- d) *Protocolo de comunicación*: Conjunto de reglas para el intercambio de información y de definiciones de los formatos de los mensajes para la interacción fructífera entre dos o más entidades. Por ejemplo, el popular protocolo IP, base de Internet.

Para estudiar las arquitecturas de comunicaciones es conveniente pensar que los objetos en comunicación son los procesos de aplicación y no los sistemas (finales) donde se alojan. Bajo esta premisa, ya puede intuirse que la tarea de poner en comunicación dichos procesos puede llegar a ser muy compleja.

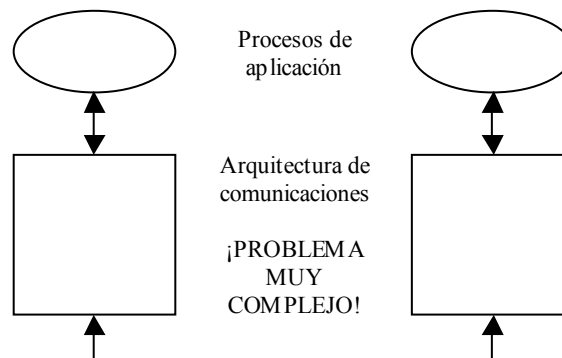


Fig. 2.1 La arquitectura de comunicaciones como estructura compleja para el intercambio de información entre procesos de aplicación

En toda comunicación entre equipos finales a través de un medio, existen diversas cuestiones a las que buscar solución con el fin de alcanzar el establecimiento de conexión. Véanse algunos aspectos a resolver:

- a) identificación de orígenes / destinos (*direccionamiento*)
- b) control de los errores de transmisión (algoritmos de detección y corrección)
- c) pérdidas de secuencia (numeración y reordenamiento)
- d) diferencias de velocidad, saturaciones (control de flujo)
- e) diferencias de longitud (segmentación y reensamblaje)
- f) optimización de costes (multiplexado, concatenación)

- g) diferencias de representación de la información (sintaxis común, conversión)
- h) seguridad, es decir, privacidad, autenticidad (criptografía)
- i) gestión del acceso a recursos compartidos (protocolos de acceso, priorización)
- j) determinación del mejor camino a seguir (encaminamiento)

El número de problemas a resolver para permitir una comunicación correcta entre los procesos de aplicación es demasiado elevado como para atacarlos globalmente. Es mejor aplicar el principio de “divide y vencerás”.

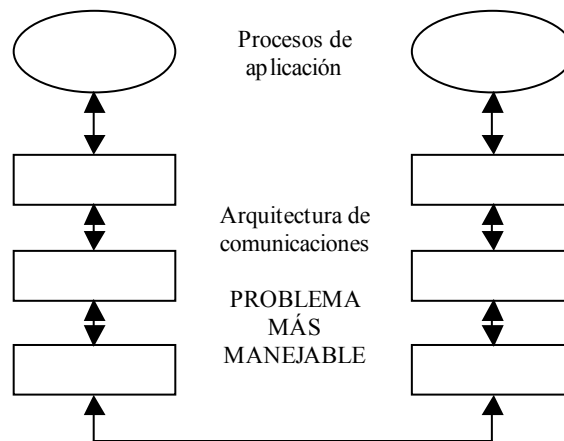


Fig. 2.2 División en bloques o conjuntos de tareas en una arquitectura de comunicaciones

Por eso, la mayoría de arquitecturas de comunicaciones están estructuradas en lo que suelen llamarse *capas* o *niveles*. Las principales ventajas de una estructuración de ese tipo son:

- a) la *modularidad* o independencia entre tareas. Permite resolver el problema general en pequeños problemas, más simples y resolubles individualmente.
- b) permitir varias alternativas para una misma tarea. Poder disponer de 2 protocolos, elegibles por la entidad de nivel superior, en función de los requisitos necesarios.
- c) la facilidad de cambios parciales. Por ejemplo, cambiar un protocolo por otro sin afectar al resto del funcionamiento. Una aplicación de ello podría ser la migración a una nueva versión de protocolo desde una anterior (pasar de IP versión 4 a IP versión 6, sin variar el resto de la torre de protocolos, obteniendo por tanto el beneficio del nuevo estándar en las funciones de las que específicamente sea responsable).

### 2.3 Analogía de una arquitectura de comunicaciones

Para ilustrar mejor el concepto y la utilidad de una arquitectura de comunicaciones, se presentará a continuación una analogía ampliamente usada en la literatura sobre el tema.

Supongamos dos pensadores (filósofos) que viven en países distintos, con lenguas nativas distintas y que están interesados en establecer un intenso debate dialéctico. Ellos representarán los procesos de aplicación que desean intercambiar información al más alto nivel, sin preocupaciones secundarias. Para ellos debe diseñarse una arquitectura de comunicaciones que aborde básicamente dos problemas: la separación geográfica entre los filósofos y su separación idiomática (distinta forma de representar sus pensamientos). En la figura 2.3 puede verse el esquema de dicha arquitectura.

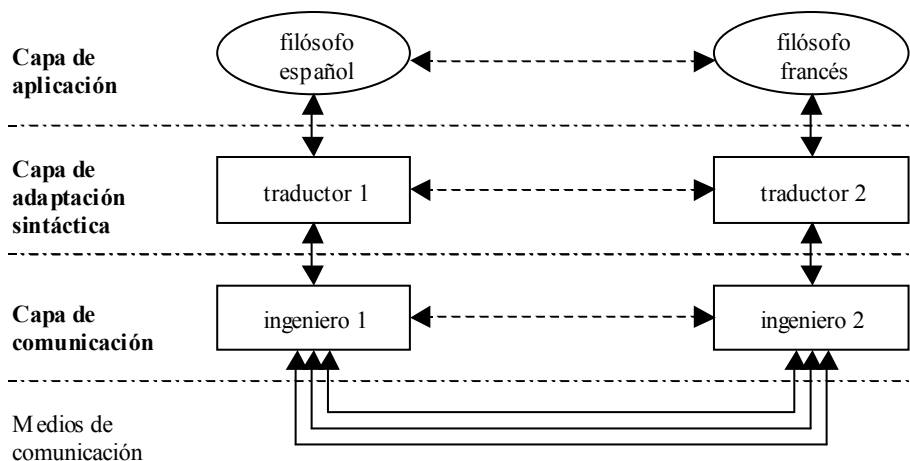


Fig. 2.3 Analogía de una arquitectura de comunicaciones

La capa de adaptación sintáctica aborda el problema de la distinta forma en que los filósofos (procesos de aplicación) representan sus pensamientos (información), es decir, ofrece un servicio de intercambio de información transparente (independiente) al idioma en que está expresada. Con este fin, dicha capa se construye con dos entidades, *traductor1* y *traductor2*, colaborando entre sí, capaces de traducir a/desde el idioma de cada filósofo a otros idiomas comunes a ambos traductores (al menos uno de ellos).

La capa de comunicación aborda el problema de enviar cualquier información entre los sitios geográficos distintos donde residen los filósofos, es decir, ofrece un servicio de transporte de información entre sitios distantes, óptimo (en cuanto a rapidez, fiabilidad o coste) y transparente (independiente) a los problemas (extravíos, retrasos, desordenamientos, etc.) que puedan surgir en los envíos. A tal fin, dicha capa se construye con dos entidades, *ingeniero1* e *ingeniero2*, colaborando entre sí, con un conjunto de medios de comunicación (teléfono, fax, correo postal, correo electrónico, etc.) a su disposición para ponerse en contacto.

Analicemos algunos aspectos que el uso de esta arquitectura de dos niveles o capas supone: el diálogo entre los filósofos es *virtual* (no es directo) y se produce a través de la arquitectura diseñada para ellos; en concreto, su diálogo sólo se produce a través de su interacción con la capa de adaptación sintáctica, desconociendo ellos cualquier detalle de cómo dicha capa funciona y, mucho menos, cualquier detalle de las capas inferiores a ésta.

Algo similar puede decirse de las entidades traductor; éstas, para llevar a cabo la función que tienen encomendada, se comunican virtualmente (no directamente) a través de su interacción con la capa de comunicación, desconociendo cualquier detalle de cómo su información es intercambiada y qué medios se usan para ello.

Veamos otro aspecto: las entidades de la capa de adaptación sintáctica no hacen valoraciones, ni mucho menos modifican, los pensamientos intercambiados por los filósofos; es decir, son de nuevo transparentes en ese aspecto; sólo se ocupan de que el significado (semántica) de los pensamientos de los filósofos se mantenga lo más intacto posible durante el proceso de traducción de idioma que llevan a cabo. Lo mismo aplica a las entidades de la capa de comunicación que, tomando literalmente (transparentemente) la información proporcionada por la capa superior, se encargan de que ésta llegue intacta y de la mejor forma posible (económica, rápida, etc.) al otro extremo.

Además, debe observarse que cuando uno de los filósofos, inquieto ante una idea nueva que acaba de surgir en su mente, decide intercambiar impresiones con su colega e invoca los servicios de la arquitectura de comunicaciones que tiene a su disposición, se desencadenan una serie de hechos en las capas de dicha arquitectura. El filósofo solicita el servicio de la capa de adaptación sintáctica aportando su pensamiento y quizás algún parámetro relativo a la urgencia requerida en su envío; el traductor se pone en contacto con su colega de capa invocando para ello los servicios de la capa de comunicación; una vez en contacto, ambos traductores se saludan, se comunican la lista de idiomas que dominan (de alguna forma preestablecida) y negocian cuál será el mejor idioma (o quizás el único posible) para llevar a cabo sus funciones como capa.

Algo similar sucede con los ingenieros de la capa de Comunicación: se saludan, usando un medio de comunicación preestablecido; se comunican la lista de medios de comunicación de que disponen cada uno de ellos y negocian cuál es el mejor medio a utilizar en aquella ocasión, basándose quizás en algún parámetro de urgencia proporcionado por la capa superior. En resumen, puede decirse que el servicio que ofrece cada capa requiere de una fase previa de establecimiento, con posible negociación, antes de llevar a cabo el servicio propiamente dicho; es lo que se llama servicio *orientado a conexión*.

Otros aspectos remarcables son la existencia de lo que podemos denominar un diálogo *horizontal* o *protocolo de capa*, entre las entidades de una misma capa para iniciar, llevar a cabo y finalizar las funciones de capa. Asimismo, también existe un diálogo *vertical*, entre entidades de capas adyacentes que residen en un mismo sitio, con el que se expresa la demanda / ofrecimiento de los servicios de capa.

Este diálogo vertical se conforma mediante las llamadas *primitivas de servicio*, que permiten tener acceso a los servicios ofrecidos por las capas inferiores y comunicar con las superiores. Por ejemplo, una alarma de cable desconectado se notifica desde el nivel inferior hacia las capas superiores mediante primitivas.

Finalmente, la concepción de arquitectura modular permite cambios parciales sin que haya que modificarla enteramente. Por ejemplo, las entidades *traductor* de la capa de adaptación sintáctica

pueden sustituirse (los traductores también merecen vacaciones...) por otras que quizás conozcan idiomas distintos.

La funcionalidad de la capa (sus servicios), sin embargo, no se verán alterados por eso. Lo mismo puede decirse de las entidades *ingeniero*, que pueden ser sustituidas, o de los medios de comunicación a su disposición, que pueden variar de un momento a otro.

## 2.4 El modelo de referencia OSI de la ISO

### 2.4.1 Introducción

Como se ha visto en la breve introducción histórica del apartado 2.1, a finales de la década de 1970 existían un número creciente de sistemas informáticos a conectar. Dichos sistemas eran heterogéneos, es decir:

- a) usaban distintos procedimientos de comunicación
- b) usaban distintos formatos para representar la información intercambiada
- c) se basaban en arquitecturas desarrolladas por sus fabricantes para ellos mismos (*propietarias*)
- d) la conexión entre ellos era imposible

Por ello, en 1977 la ISO (*International Standards Organization*) promovió el desarrollo de un modelo de referencia sobre el que desarrollar una arquitectura de comunicaciones *abierta* a adoptar por todo sistema informático en una red de comunicaciones.

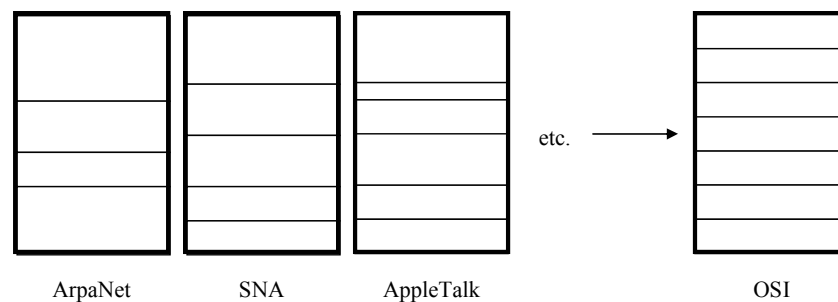


Fig. 2.4 Evolución hacia una arquitectura común (abierta) propuesta por la ISO

Los sistemas que adoptasen dicha arquitectura se llamarían *sistemas abiertos*, es decir, capaces de entenderse entre sí, independientemente de su fabricante, de su *hardware* y de su sistema operativo. El documento básico del modelo OSI (*Open Systems Interconnection*) vio la luz en 1983 (ISO 7498).

Algunas características de dicho modelo:

- a) Es *universal* (apto para cualquier tipo de interacción entre procesos de aplicación de cualquier tipo, en cualquier contexto, en el presente y en el futuro)
- b) Es *abstracto* (independiente de la tecnología empleada en su realización, del sistema operativo del sistema donde se implante y de la red)

El modelo OSI contempla una arquitectura de comunicaciones dividida en siete capas o niveles. Algunas de las razones que condujeron a la definición y número de las capas son las siguientes:

- a) Cada capa debe representar un nivel de abstracción claramente distinto en el conjunto de tareas encomendado a una arquitectura de comunicaciones
- b) Cada capa se define de manera que el interfaz entre ellas sea lo más claro y sencillo posible
- c) El número de capas debe ser suficientemente grande como para que funciones claramente distintas queden separadas
- d) El número de capas debe ser suficientemente reducido como para que el modelo no se complique con excesivos interfaces y protocolos distintos
- e) Debe aprovecharse el esfuerzo de investigación y la práctica que representaban las arquitecturas propietarias ya existentes

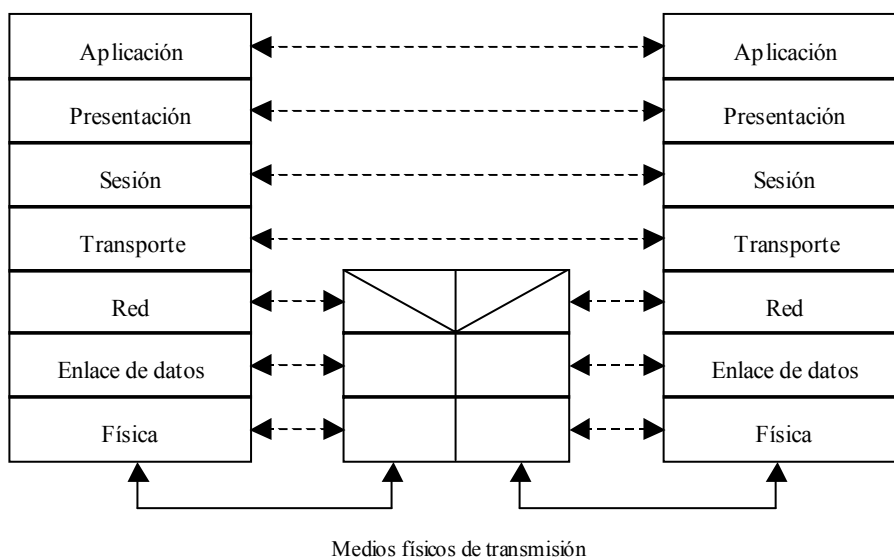


Fig. 2.5 Modelo arquitectónico de siete capas propuesto por la ISO

### 2.4.2 Terminología OSI

El estándar OSI de la ISO recoge en su especificación una terminología de uso común. Los principales términos se definen a continuación.

- a) *Capa o nivel N*: conjunto de funciones afines en una arquitectura de comunicaciones. Construida por la unión de las diversas entidades de capa N dialogando entre sí por medio del protocolo de capa.
- b) *Entidad de capa N*: realización de las funciones de la capa en un sistema (final o intermedio) concreto. Dicha realización no está especificada por el modelo; una entidad puede ser una pieza *hardware* (circuito integrado) o *software* (proceso en ejecución en un sistema operativo), etc. La interacción entre todas las entidades de una misma capa ofrece la funcionalidad (los servicios) de dicha capa. Dicha interacción entre entidades es de igual a igual; por eso las entidades de una misma capa se llaman entidades “parejas” (*peer entities*).
- c) *Protocolo de capa N*: el protocolo que siguen todas las entidades parejas dentro de una capa. Constituye un diálogo horizontal entre entidades que residen en sistemas distintos. Es objeto de detallada y estricta estandarización, por tanto.
- d) *Primitivas de servicio de capa N*: la definición (en forma de funciones y parámetros asociados) de los servicios que ofrece una capa. Las primitivas se intercambian en un diálogo vertical entre entidades de capas adyacentes (N-1 o N+1) que residen en un mismo sistema. Dado que ese diálogo es local o interno al sistema, su realización concreta no es objeto de tan estricta definición como en el caso anterior.
- e) *N-SAP (N-Service Access Point)*: puntos de acceso al servicio de nivel N. Puntos, en el interfaz entre capas adyacentes, donde se solicitan/prestan los servicios de capa N. Permiten distinguir diferentes entidades usuarias o servidoras. Dicho de otro modo, son las direcciones en sentido genérico (locales o externas) donde se ofrecen los servicios de capa N o donde están sus usuarios.
- f) *N-PDU (N-Protocol Data Unit)*: unidad de datos del protocolo de capa N. Es cualquier mensaje que una entidad de capa N intercambia con sus entidades parejas siguiendo el protocolo de capa N. Dicho mensaje puede contener datos de la capa superior o no.
- g) *N-PCI (N-Protocol Control Information)*: información de control del protocolo. Parte de la N-PDU que contiene información generada por y para las entidades de capa N. Las entidades de capa N dialogan a través de la información contenida en las N-PCI que añaden usualmente a la información asociada a los servicios invocados por la capa superior, las N-SDU, para constituir las N-PDU. Las N-PCI equivalen a lo que, en otros contextos, se suelen llamar *cabeceras (headers)* o *colas (trailers)*.
- h) *N-SDU (N-Service Data Unit)*: unidad de datos del servicio de capa N. Parte de una N-PDU que contiene los datos de usuario (provenientes de la capa superior y tratados de forma transparente) para los que debe realizar el servicio solicitado.

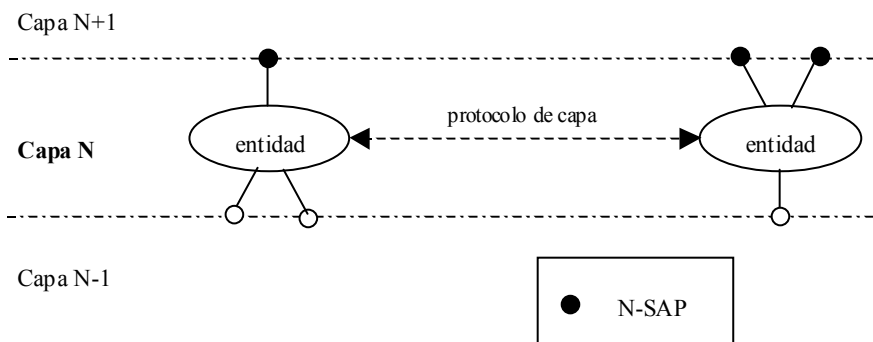


Fig. 2.6 Definiciones principales de la arquitectura OSI

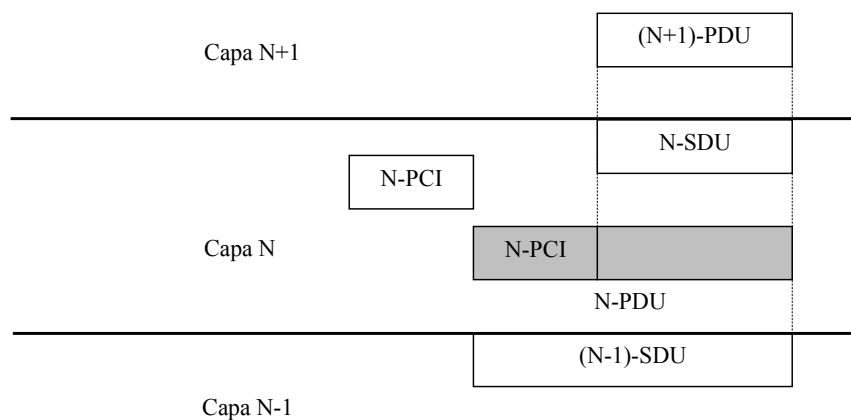


Fig. 2.7 Encapsulado de los datos en el modelo OSI

### 2.4.3 La capa física

Algunas características de la capa física son:

- Su contacto directo con los medios físicos de transmisión.
- Ofrece un servicio básico de transmisión de bits entre puntos adyacentes (conectados al mismo medio de transmisión). Dicho servicio no tiene por qué ser fiable (puede ser propenso a los errores).
- Define las características mecánicas, eléctricas y de procedimiento para el inicio, mantenimiento y finalización del servicio anteriormente descrito.

- d) La unidad de información en este nivel es el bit (a veces grupos de pocos bits: *dibits*, *nibbles*, *bytes*).
- e) En ella se definen aspectos como el acoplamiento al medio físico de transmisión (descripción mecánico-eléctrica de conectores, incluso antenas, lentes...); la técnica de modulación o codificación empleada; los niveles de tensión, corriente o intensidad luminosa asociados a los distintos símbolos que representan uno o más bits; la duración de los bits ( $T_b$ ) y por tanto la tasa de bit/s ( $R$ ); similarmente, la tasa (*baud rate*) de símbolos/s; la sincronización de bit; etc.

Son ejemplos de especificaciones de capa física: EIA RS-232C (para enlaces locales de datos entre computadores y periféricos), IEEE 802.3 (una parte de ella, para redes de área local *Ethernet*), CCITT I.430 (para el acceso básico a la RDSI), etc.

#### 2.4.4 La capa de enlace de datos

La capa de enlace de datos mejora el servicio de transmisión de bits -en bruto- de la capa física. Sus funciones principales son:

- a) Control de errores
- b) Control de flujo
- c) Control de acceso a medios físicos compartidos.

La unidad de información es un grupo de bits llamado *trama* (en terminología OSI, 2-PDU o L-PDU; donde la L significa *Link*).

#### Métodos de control de errores y flujo

El control de errores se realiza de forma más eficiente considerando los datos agrupados en conjuntos de bits u octetos. La idea básica consiste en añadir a dichos grupos una información adicional, llamada *redundancia*, que se obtiene a partir de determinadas operaciones realizadas sobre ellos. En función del tipo y cantidad de redundancia añadida, se tienen dos filosofías de control de los errores:

- a) *FEC (Forward Error Correction)*: corrección directa de errores; cuando la información de redundancia, en proporción elevada, es tal que permite a la recepción determinar exactamente aquellos bits que han sufrido errores y, por tanto, corregirlos directamente. Se emplea cuando no son adecuadas las retransmisiones, se requiere un bajo retardo o el enlace es unidireccional (*simplex*).
- b) *ARQ (Automatic Repeat reQuest)*: retransmisión automática; cuando la redundancia, en pequeña proporción, solamente permite a la recepción determinar si el conjunto de bits (trama), considerado como un todo, presenta algún error; la corrección se basa entonces en la retransmisión repetida de la trama hasta que se reciba sin error; requiere canales bidireccionales (*half* o *full duplex*) e introduce más retardo en caso de errores frecuentes.

Por otra parte, el *control de flujo* es la función encargada de armonizar las velocidades de generación de datos (tramas) en la fuente a las capacidades de procesado en el destino a fin de no perder información.

Se basa en el envío en sentido contrario al del flujo de datos de señales de control o regulación. El método más simple consistiría en señales de detención/reanudación del flujo de datos; otros métodos más sofisticados contemplarían el envío hacia la fuente de cantidades máximas (créditos) de bits o tramas permitidos en un momento dado. El control de flujo se encuentra usualmente combinado con el control de errores.

A continuación se describirán los principales mecanismos que combinan técnicas de control de errores con las de flujo para conseguir una transmisión exenta de errores en un enlace, esto es, una comunicación *fiable*. Estos mecanismos son el de *parada y espera* y los protocolos *ARQ continuos*.

### Protocolo Stop & Wait (parada y espera)

Es la forma más simple de control de errores ARQ y también de flujo. Dicho protocolo establece que el emisor, tras enviar una trama de datos, detiene sus envíos y permanece a la espera de una señal por parte del receptor que le indique el éxito o fracaso en el envío de aquella trama. Si la señal es favorable, el emisor procede a enviar la siguiente trama de datos; en caso contrario, el emisor vuelve a transmitir la misma trama.

La señal positiva, de éxito en la transmisión (usualmente llamada ACK: *Acknowledgment*) siempre debe existir, mientras que la negativa, de fracaso en la transmisión, puede ser explícita (NAK: *Negative Acknowledgment*) o implícita (ausencia de señal tras cierto tiempo).

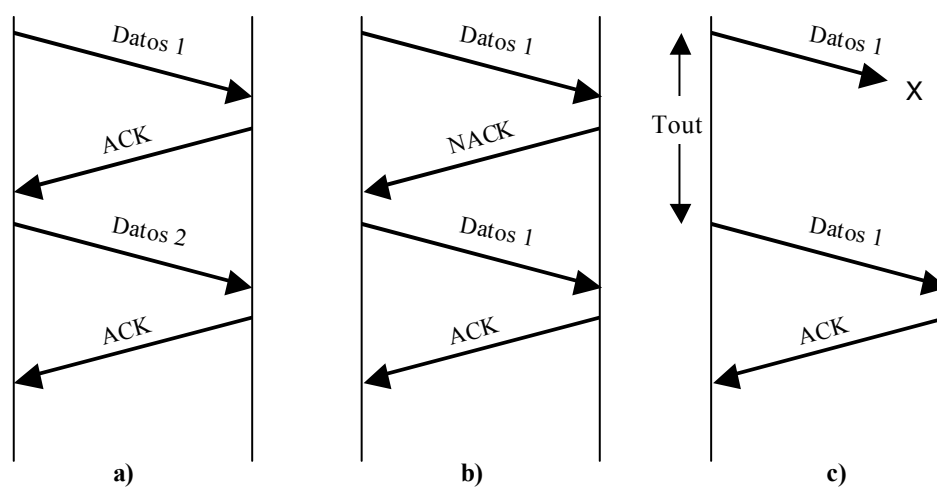


Fig. 2.8 Protocolo parada-espera: a) confirmación positiva; b) confirmación negativa explícita; c) confirmación negativa implícita (por falta de respuesta tras un cierto tiempo)



Defínase el rendimiento, eficiencia o utilización ( $U$ ) de un protocolo como la relación entre la tasa de datos de usuario (en bit/s) que éste puede servir y la tasa bruta ( $R$ ) (en bit/s) del canal sobre el que opera.

Se calculará la eficiencia en el protocolo parada-espera en el caso en que no haya errores de transmisión.

En relación a la figura 2.10 tenemos:

$T_t$ : el tiempo de transmisión de una trama de datos (que se consideraran todas de igual longitud)

$T_p$ : el tiempo de propagación a través del enlace

$T_{proc}$ : el tiempo de procesado (que incluye, en transmisión, los tiempos necesarios para tomar el siguiente bloque de datos de usuario de la memoria, calcular sus bits de redundancia, etc. y, en recepción, el tiempo necesario para determinar si el bloque recibido es erróneo o no a partir de los bits de redundancia, etc.)

$T_{ack}$ : el tiempo de transmisión del mensaje de confirmación ACK

$T$ : el tiempo total realmente empleado para el envío de un bloque de datos

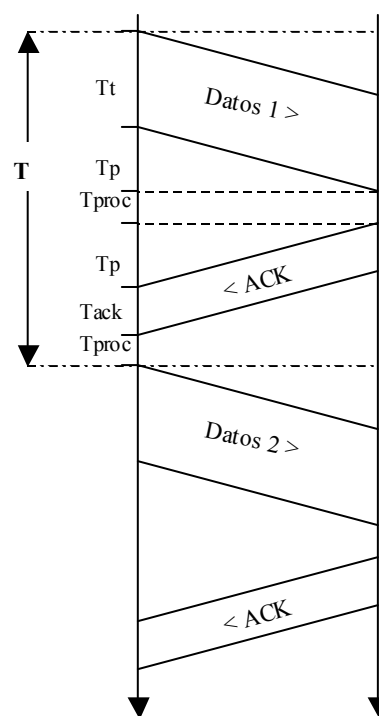


Fig. 2.10 Protocolo parada-espera. Tiempos a considerar en el análisis de su rendimiento

Puede verse fácilmente que  $T = T_t + 2T_p + 2T_{proc} + T_{ack}$ .

Considerando que el tiempo de procesado del protocolo es despreciable (es decir, que la velocidad de ejecución del protocolo es prácticamente infinita; lo cual es bastante cierto para los controladores en circuito integrado actuales) y que el tiempo de transmisión del mensaje de ACK (unos pocos bits) es muy pequeño en comparación con el del bloque de datos, tenemos que  $T \approx T_t + 2T_p$ .

La definición de eficiencia dada anteriormente puede expresarse de forma equivalente como un cociente de tiempos en vez de tasas; así,

$$U = \frac{T_t}{T} = \frac{T_t}{T_t + 2T_p} = \frac{1}{1 + 2\frac{T_p}{T_t}} = \frac{1}{1 + 2a} \quad (2.1)$$

En donde el parámetro  $a$  se ha definido como el tiempo de propagación en el enlace con respecto al tiempo de transmisión de las tramas de datos:

$$a = \frac{T_p}{T_t} \quad (2.2)$$

Dicho parámetro “resume” las características del enlace y también puede expresarse como

$$a = \frac{T_p}{T_t} = \frac{dR}{V_p L_t} \quad (2.3)$$

donde:

$d$ : longitud del enlace de datos (en m)

$R$ : tasa de transmisión (en bit/s)

$V_p$ : velocidad de propagación en el enlace (en m/s)

$L_t$ : longitud de la trama de datos (en bit).

La expresión de la eficiencia obtenida anteriormente puede completarse con el fin de introducir la ulterior reducción de ésta que supone la *sobrecarga* de los bits de control (números de secuencia, bits de redundancia, etc.). Así,

$$U' = \frac{L_d}{L_t} U = \frac{L_d}{L_t} \frac{1}{1 + 2a} = \frac{L_d}{L_d + L_h} \frac{1}{1 + 2a} \quad (2.4)$$

Donde  $L_d$  es la longitud de los datos de usuario y  $L_h$  es la longitud de los datos de control del protocolo (la PCI).

A modo de ejemplo, se calculará la eficiencia de un protocolo parada-espera en tres contextos distintos:

a) *Enlace entre 2 estaciones terrestres a través de un satélite en órbita geoestacionaria:*

$$d=2\cdot 38.000 \text{ km}, R=2\text{Mbit/s.}, L_i=4000 \text{ bit}, V_p=3\cdot 10^8 \text{ m/s}$$

$$T_p=270 \text{ ms}, T_i=1,953 \text{ ms}$$

$$a=138,24, U=0,0036$$

b) *Red de área local clásica:*

$$d=1 \text{ km}, R=4\text{Mbit/s.}, L_i=500 \text{ bit}, V_p\approx 2\cdot 10^8 \text{ m/s}$$

$$T_p=5 \mu\text{s}, T_i=125 \mu\text{s}$$

$$a=0,04, U=0,93$$

c) *Transmisión de datos sobre red telefónica (módem):*

se contemplan dos escenarios (comunicación local e intercontinental)

$$d_1=1 \text{ km}, d_2=5000 \text{ km.}, R=9600 \text{ bit/s.}, L_i=500 \text{ bit}, V_p=2\cdot 10^8 \text{ m/s}$$

$$T_{p1}=5 \mu\text{s}, T_{p2}=25 \mu\text{s}, T_i=52,1 \text{ ms}$$

$$a_1=0,000096, U_1\approx 1$$

$$a_2=0,48, U_2=0,69$$

### Protocolos ARQ continuos

Cuando el valor del parámetro  $a$  no es pequeño (digamos  $a>0,05$ ) el protocolo parada-espera no presenta una buena eficiencia. Para mejorarla debe pensarse en un protocolo que “inunde” el canal de comunicación, es decir, que transmita continuamente tramas, aun cuando se desconozca la suerte de la transmisión de tramas anteriores. El número de tramas ( $N$ ) que el transmisor puede enviar sin tener que detenerse a esperar confirmaciones es el llamado *crédito* o *ventana* de transmisión. La figura 2.11 ilustra el funcionamiento de ese tipo de protocolos.

Como puede verse de la figura 2.11, existen, a su vez, dos variantes de protocolos ARQ, en función del procedimiento seguido cuando hay errores en las tramas recibidas:

1. *ARQ Go-Back N* (vuelta atrás): el receptor, cuando detecta una trama errónea, descarta las tramas siguientes a esa y se queda a la espera de recibir de nuevo la trama en cuestión; el emisor, cuando se percibe del error en una trama, vuelve hacia atrás y procede a retransmitir la trama en error y todas las tramas que ya hubiera transmitido.

Este método, que puede ser ineficiente porque retransmite tramas que pueden haber sido recibidas correctamente, simplifica la realización del protocolo, dado que no requiere capacidad de almacenamiento y reordenación de tramas en el receptor.

2. *ARQ Selective Reject* (rechazo selectivo): el receptor, cuando detecta una trama errónea, envía la señal para su retransmisión pero sigue aceptando las tramas siguientes; el emisor, cuando se apercibe del error en una trama, sólo retransmite la trama errónea a la primera oportunidad. Este método es más eficiente, pero supone que el receptor es capaz de almacenar varias tramas para, cuando reciba correctamente la trama en error, entregarlas en orden al usuario del nivel superior.

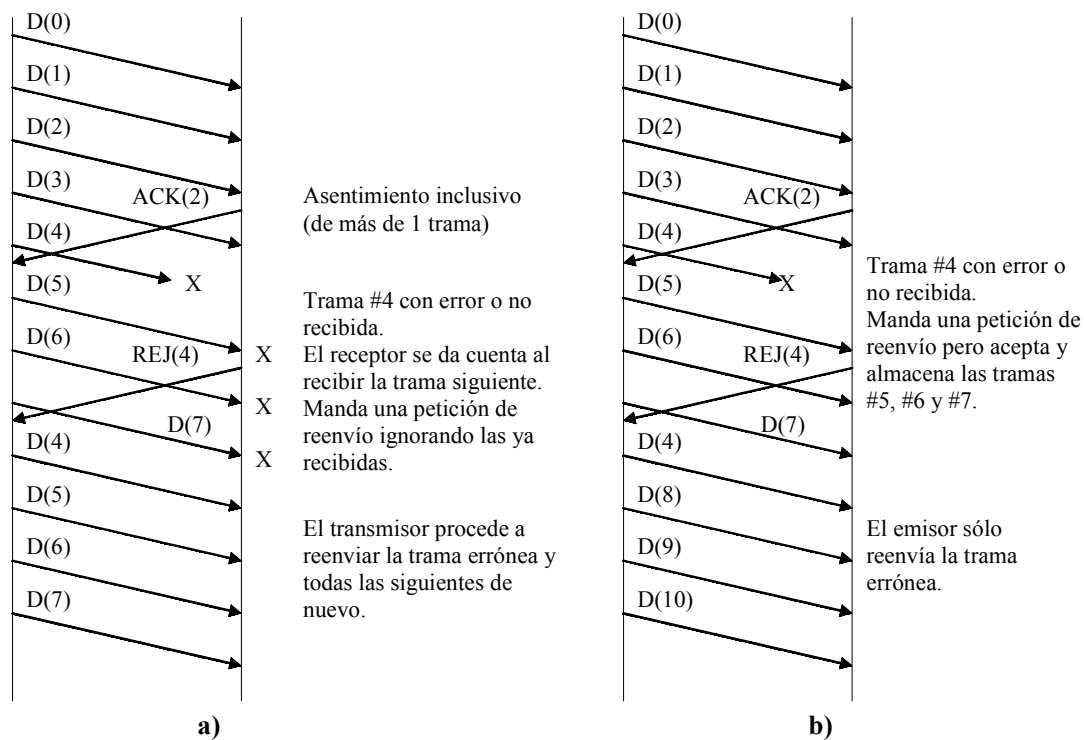


Fig. 2.11 Protocolos ARQ continuos: a) go-back (vuelta atrás); b) selective reject (rechazo selectivo)

Los protocolos ARQ continuos presentan una eficiencia de 1 (en el caso sin errores y sin considerar la sobrecarga de la cabeceras de control) cuando la ventana de transmisión ( $N$ ) es “suficientemente” grande, es decir, cuando el transmisor nunca debe detenerse a la espera de confirmaciones. Tras cierta reflexión, puede verse que esto se cumple cuando  $N > 2a + 1$ .

Cuando se considera el efecto de los errores en las tramas, es decir, cuando existe una probabilidad no nula de error en cualquier trama transmitida ( $P_t$ ), la eficiencia se reduce consecuentemente. El análisis de la eficiencia en esta situación resulta algo más complejo.

La implementación real de un protocolo ARQ debe contemplar muchos detalles no comentados aquí (el establecimiento/iniciación del enlace de datos, el control de flujo, el soporte del modo *full-duplex*, etc.). Esto conlleva a una definición de nuevos campos en las cabeceras (L-PCI) de las tramas (L-PDUs) y a una especificación compleja del protocolo (según los estados y transiciones entre ellos, variables de estado, temporizadores, etc.). Un buen ejemplo de implementación de un protocolo de enlace de datos clásico lo constituye la familia de protocolos HDLC (*High level Data Link Control*) estandarizada por la ISO.

*Algunos ejemplos de protocolos pertenecientes a la capa de enlace de datos:*

- a) BSC (*Binary Synchronous Control*): uno de los primeros protocolos para el control de errores y flujo desarrollado por IBM en las década de 1960. Era *orientado a carácter*, es decir, los campos de control y datos estaban estructurados en octetos.
- b) SDLC (*Synchronous Data Link Control*): mejora del protocolo anterior también debida a IBM. Es *orientado a bit*, es decir, los campos de control y datos contienen un número arbitrario de bits.
- c) HDLC (*High level Data Link Control*): Familia de protocolos de enlace de datos estandarizada por la ISO e inspirada en SDLC.
- d) LAP-B (*Link Access Procedure – Balanced*): Variante de HDLC que define la capa de enlace de datos en el estándar de acceso a redes públicas de conmutación de paquetes definido en la norma X.25 del ITU-T.
- e) LAP-D (*Link Access Procedure on D channel*): Variante de HDLC que define la capa de enlace de datos en el canal D de señalización en el estándar de acceso a la RDSI (Red Digital de Servicios Integrados) de banda estrecha.
- f) LLC (*Logical Link Control*): Variante de HDLC que define la sub-capa superior de la capa de enlace de datos para redes de área local según la norma IEEE 802.2.

### **Control de acceso a medios físicos de transmisión compartidos**

Cuando el medio físico de transmisión es compartido por más de dos sistemas (lo que a veces también se denomina *enlace multi-punto*) surge el problema adicional de arbitrar su uso, es decir, determinar *cuándo* y *cómo* cada uno de los sistemas puede usarlo, evitando los conflictos. Dicha función, dentro de la capa de enlace de datos, tiene tal relevancia que, arquitectónicamente hablando, se la ha asociado a una subcapa denominada MAC (*Medium Access Control*). Así pues, cuando convenga (típicamente en entornos de redes de área local), la capa de enlace de datos se estructura según la figura 2.12.

Existen multitud de métodos (protocolos) para realizar la funcionalidad de la subcapa MAC, dependiendo de si el control del acceso reside en una sola entidad (el controlador maestro) o está

distribuido entre todas las entidades que comparten el medio, o en función del grado de ordenamiento o determinismo de los accesos por parte de las diferentes entidades (protocolos *aleatorios*, por *demanda* o por *turnos*). Todo ello se verá más en detalle en el capítulo 4.

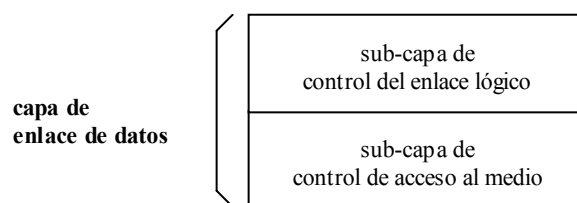


Fig. 2.12 Subdivisión de la capa de enlace de datos en el caso de medios de transmisión compartidos por múltiples sistemas

### 2.4.5 La capa de red

Esta capa, situada por encima de la de enlace, aprovecha los servicios brindados por esta y añade algunas características, como son:

- a) Reenvío (*relaying*) de la información a través de los distintos enlaces y sistemas intermedios que constituyen una red de comunicaciones.
- b) Encaminamiento (*routing*) de la información, es decir, elección del camino a seguir a través de la red en función de la optimización de algún criterio (coste, rapidez, seguridad, fiabilidad, equilibrio –justicia-, etc.). El otro término relacionado es el de conmutación que se aplica más bien al proceso que sufre la información en los sistemas intermedios cuando ésta es encaminada.
- c) Control de congestión, es decir, control de todas las comunicaciones que se producen a través de la red con el fin de que los recursos de ésta se utilicen de la mejor forma posible.
- d) Tarificación, es decir, cómputo de los costes incurridos en el envío de la información a través de la red en función de volumen, del tiempo empleado, de la distancia, o del grado de servicio (calidad) ofrecido.
- e) Interconexión (adaptación) entre redes.
- f) La unidad de información en esta capa es el paquete o datagrama (para las redes de conmutación de paquetes).

Puede decirse que la capa de red articula el conjunto de enlaces físicos, mejorados por la capa de enlace de datos, para constituir lo que propiamente se entiende por red de comunicaciones. Aunque la capa de Red está obviamente implementada en los sistemas finales, es en los sistemas intermedios donde ésta se encuentra realizada de forma completa.

### Servicios de Red Orientados y No Orientados a conexión

En la sección 1.2.1 del capítulo de introducción se definieron los conceptos de servicio *orientado* y *no-orientado* a conexión. En ésta, se planteará su aplicación en el nivel de red de la arquitectura OSI.

El *servicio de red orientado a conexión (OC)* se caracteriza por el establecimiento de un camino y la reserva de ciertos recursos, lo que usualmente se llama un *circuito virtual*, en la red.

Supone tres fases: establecimiento de la conexión, intercambio de datos y la liberación de la conexión (esto es, de los recursos que se han ocupado durante la fase de intercambio de la información). El símil para este tipo de servicio lo constituiría la realización de una llamada telefónica.

Esta técnica ofrece las siguientes ventajas:

- a) Requiere poco esfuerzo de encaminamiento durante la fase de transferencia, al estar fijado el camino
- b) Ofrece una gran facilidad para garantizar diversas calidades de servicio, al poder reservar recursos de antemano
- c) La tarificación es fácil de llevar a cabo
- d) Garantiza gran fiabilidad, al poder secuenciar los paquetes como pertenecientes a una conexión y poder detectar la pérdida de alguno o evitar su desordenamiento
- e) Menor complejidad en los sistemas finales o terminales en cuanto al control de errores

Aunque presenta también algunos inconvenientes:

- a) El establecimiento de la conexión es un proceso lento y costoso en recursos de red y, por tanto, mal adaptado a aplicaciones que envían pequeñas ráfagas de datos a múltiples destinos
- b) Al estar fijado el camino, la caída de un nodo de conmutación o enlace origina la pérdida de las conexiones que pasaban a través de ellos, aun cuando topológicamente existan caminos alternativos
- c) Requiere una elevada capacidad de memoria en los nodos de conmutación para almacenar el *estado* de todos los circuitos virtuales

Por otro lado, el *servicio de red no orientado a conexión (NOC)* se caracteriza por que cada bloque de datos (datagrama) es auto-contenido (lleva información de su destino) y es tratado de forma individual e independiente por la red. No existe, por tanto, camino establecido a priori ni fases asociadas al servicio.

El símil para este tipo de servicio lo constituiría el envío de cartas del servicio postal. Las ventajas de este procedimiento son las siguientes:

- a) No presenta el retardo previo que se requiere para el establecimiento de una conexión

- b) Al no existir un camino fijado *a priori*, sino determinado para cada datagrama, se tiene mayor robustez frente a fallos de nodos de conmutación o enlaces (siempre que topológicamente existan caminos alternativos)
- c) Se caracteriza por un uso más eficiente de los recursos de red al no existir ninguna reserva previa
- d) No se requiere memoria en los nodos de conmutación para almacenar información relativa a las conexiones pues éstas no existen

Aunque tiene los siguientes inconvenientes:

- a) Al tener que evaluar el trayecto que debe seguir cada unidad de datos, se requiere mayor capacidad y velocidad de cálculo en los nodos de conmutación
- b) Muestra cierta ineficiencia dado que cada datagrama debe llevar información completa (direcciones, etc.) para su tránsito por la red
- c) Tarificación más difícil, especialmente cuando existe más de un proveedor del servicio de red, al tener que identificar los usuarios fuente y destinatarios de cada datagrama
- d) Control de la congestión más difícil, al no existir ninguna planificación/reserva de recursos de antemano
- e) Poca fiabilidad, al no poder secuenciar los datagramas como pertenecientes a ninguna conexión y no poder detectar su pérdida o desordenamiento

Puede resumirse diciendo que las diferencias entre el servicio de red OC y el NOC residen en su mayor o menor fiabilidad y en donde se sitúa la complejidad (en los sistemas intermedios para el servicio OC y en los sistemas finales -capa de transporte más sofisticada- para el servicio NOC).

Durante la redacción del modelo arquitectónico OSI, se propuso que el servicio de red debía ser OC (más en la línea de pensamiento en Europa), aunque los defensores del servicio NOC (más en la línea de pensamiento en EEUU) también hicieron prevalecer sus argumentos. Finalmente se incluyeron ambos tipos de servicio de red.

## Direccionamiento

El *direccionamiento* es la identificación única de cada equipo terminal o sistema final conectado a una red o grupo de redes interconectadas.

Para ello se suelen utilizar números. Dos aspectos importantes son:

- a) el número de dígitos o bits usados, que determinan el tamaño del “espacio” de direcciones, es decir, el número máximo de terminales distintos que podrán conectarse las redes

- b) la estructuración de dichos números y los criterios para su asignación a los equipos terminales en función de su situación geográfica, que deben ser de tal forma que faciliten la creación de tablas y algoritmos de encaminamiento viables para los nodos de conmutación o sistemas intermedios (los esquemas de direccionamiento en la red telefónica o en la Internet, por ejemplo, siguen ese criterio).

Estos números pueden estar asociados o no a una palabra que facilite recordar el nombre del equipo. Un ejemplo donde existe esta asociación es en la red Internet. Tras cualquier dirección de página web existe siempre una identificación de 32 bits que indica el equipo donde se encuentran los contenidos solicitados.

Un ejemplo donde no existe asociación es en los números de telefonía. Pero sería francamente cómodo poder efectuar una llamada simplemente indicando al operador el nombre del interlocutor con el que se desea conversar.

### Algoritmos de encaminamiento (routing)

Los algoritmos de encaminamiento permiten a los nodos intermedios de la red tomar decisiones sobre el siguiente nodo al que deben dirigir la unidad de datos que acaban de recibir. Los criterios considerados en la toma de decisión constituyen el algoritmo de encaminamiento.

A continuación se presenta una clasificación de los algoritmos de encaminamiento:

- a) *No adaptativos*: en estos algoritmos, las decisiones de encaminamiento se basan en tablas fijas, elaboradas *a priori* y cargadas en los sistemas intermedios. Dichas tablas, que no varían con el tiempo, se calculan según la topología de la red y a las expectativas de tráfico que va a soportar.
- b) *Adaptativos*: en este caso, las decisiones de encaminamiento pueden variar en el tiempo en función de los cambios de la topología de la red (por la adición/supresión de nodos o enlaces) y del tráfico. A su vez, pueden clasificarse en:
  1. *globales o centrales*, cuando recogen información de toda la red
  2. *locales*, cuando sólo utilizan información disponible en un nodo de conmutación
  3. *distribuidos*, a medio camino entre los dos anteriores, es decir, utilizan información de zonas de la red

### Control de congestión

El control de congestión refiere a los mecanismos necesarios para que los recursos de la red se utilicen de forma armoniosa entre todas las comunicaciones. Debe evitar los efectos de realimentación positiva, que resulta negativa o perjudicial para el equilibrio de tráfico en el sistema.

Por ejemplo, en presencia de tráfico intenso, el número de paquetes perdidos por la red puede aumentar al llegar al límite la capacidad de almacenamiento en los sistemas intermedios. En esta situación, el número de retransmisiones de paquetes aumenta, lo que hace agravar aún más el problema. Existen diversos métodos para llevar a cabo el control de la congestión. Se pueden citar los siguientes, entre otros existentes:

- a) reserva de memoria (*buffers*) en los nodos de conmutación
- b) reserva de ancho de banda en los enlaces para redes con servicio orientado a conexión (OC)
- c) mecanismo de permisos (*tokens*) para transmitir paquetes con el fin de mantener más o menos constante el número de paquetes en la red en cualquier momento
- d) uso del control de flujo a nivel de cada circuito virtual para redes con servicio OC o de mensajes de limitación de flujo a nivel de equipo terminal para redes con servicio no OC

### Interconexión de redes

La propuesta inicial del modelo OSI asignaba a la capa de red la misión de resolver las diferencias para permitir la interconexión de redes con tipos de servicio y calidades distintos (aunque en el mundo real la interconexión de redes puede resolverse no sólo en dicho nivel). A efectos de interconexión, la capa de red puede dividirse hasta en tres subcapas, como se refleja en la figura 2.13.

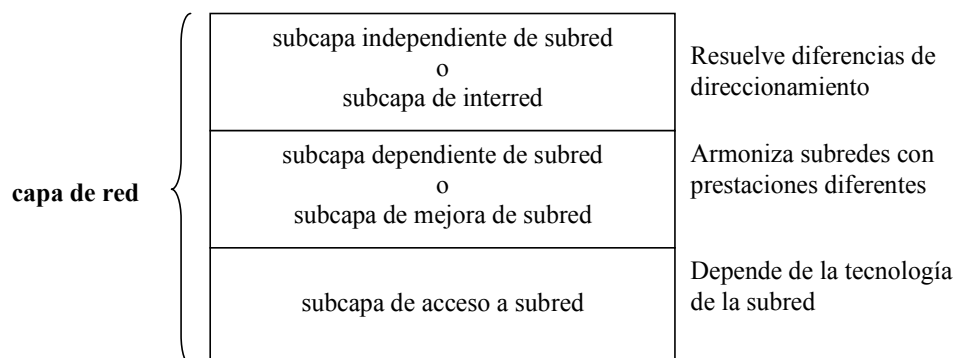


Fig. 2.13 Subdivisión de la capa de red según OSI a efectos de interconexión entre redes

Con el fin de ilustrar las ideas subyacentes en la interconexión de redes, en la figura 2.14 se presenta un ejemplo relativamente abstracto y elemental de interconexión entre una red con servicio no OC y una red con servicio OC.

La primitiva N-UNIT-DATA (la única existente) sirve para el envío de bloques de datos (datagramas) en la red con servicio no OC. Las primitivas N-CONNECT, N-DATA y N-DISC sirven para el establecimiento de conexión, el envío de datos y la liberación de la conexión, respectivamente, en la red con servicio OC. Obsérvese que en el ejemplo se establece una conexión por la que se envía el

datagrama procedente de la red con servicio no OC hacia su destino en la red con servicio OC, liberándose a continuación dicha conexión.

En muchos casos es razonable pensar que cuando un sistema final envía un bloque de datos a otro sistema final, es que se ha iniciado algún tipo de interacción entre procesos residentes en ellos y que, por tanto, en un futuro inmediato se observarán nuevos bloques de datos intercambiados entre ellos. Si esto fuera así, sería más eficiente que el sistema intermedio mantuviera la conexión abierta, es decir, no invocara la primitiva N-DISC.req tan pronto hubiera enviado el datagrama procedente de la red con servicio no OC, sino que esperara cierto tiempo antes de hacerlo. Con ello se reduciría el retardo y se disminuiría el número de establecimientos y liberaciones de conexión.

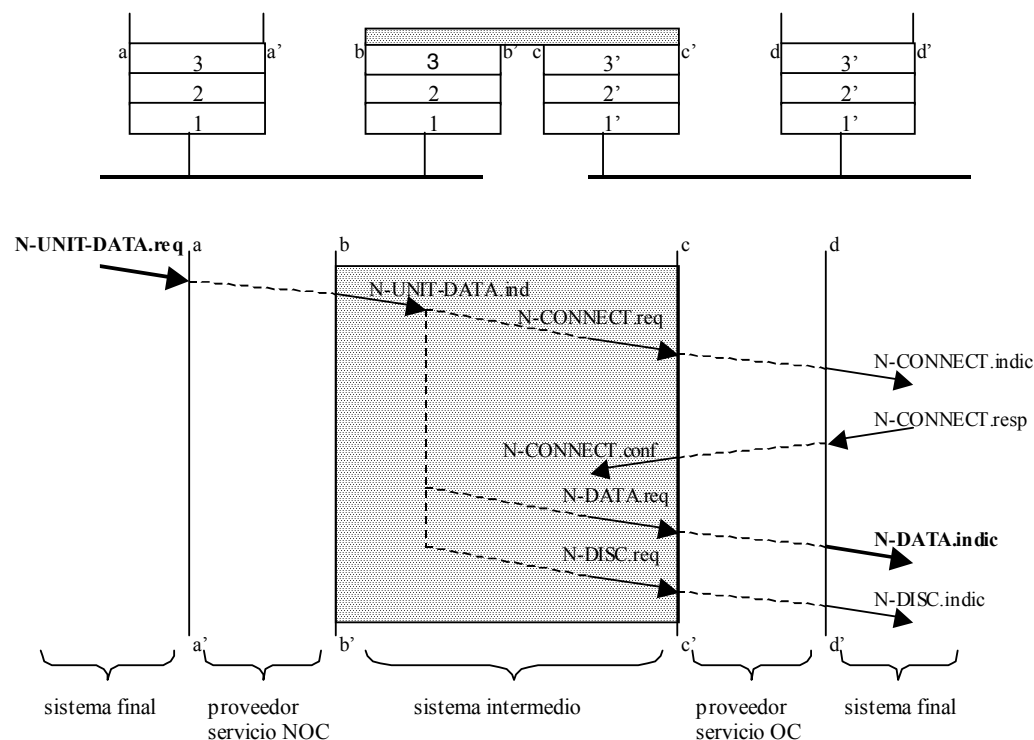


Fig. 2.14 Ejemplo de realización de un sistema intermedio para la interconexión entre una red con servicio no OC y otra red con servicio OC

## 2.4.6 La capa de transporte

La capa de transporte solamente está realizada en los sistemas finales (aquellos que alojan a los procesos de aplicación que se comunican); no existen entidades de transporte en los sistemas intermedios. Se dice, entonces, que la capa de transporte opera *extremo a extremo*, al igual que las capas por encima de ésta.

Su misión principal es “esconder” las imperfecciones de las redes subyacentes a los procesos de aplicación, ofreciendo un servicio de comunicación de datos extremo a extremo perfectamente fiable y de una calidad pactada. Esto debe hacerlo, además, de forma eficiente, es decir, optimizando el uso de los recursos que tiene a su disposición.

La anteriormente citada *calidad de servicio* (en adelante, *QoS: Quality of Service*) que la capa de transporte se compromete a ofrecer viene especificada, según OSI, hasta por 12 parámetros que pueden, a su vez, contener subparámetros. La realización completa de la capa de transporte OSI en un caso real, en cuanto a sus exigencias de compromiso con tan sofisticada especificación de QoS, podría llegar a ser extremadamente compleja. Sin embargo, actualmente el término QoS ha resurgido con fuerza ante el fenómeno de la integración de un número cada vez mayor de servicios (especialmente los de audio y vídeo en tiempo real) en redes IP (léase internet). Para ilustrar el concepto de QoS se enumeran solamente algunos de los parámetros que la definen:

- a) *Caudal*, es decir, flujo medio en bits/s entre los dos T-SAP que definen una conexión de transporte; el uso de subparámetros permitiría especificar dicho caudal de forma más completa, por ejemplo: caudal mínimo garantizado, caudal de pico, duración máxima del pico de caudal, etc.
- b) *Retardo de tránsito*, es decir, el tiempo medio desde que una T-SDU atraviesa el T-SAP origen hasta que aparece en el T-SAP destino; el uso de subparámetros permitiría especificar, por ejemplo: fluctuación (desviación) media o máxima de dicho retardo, retardo máximo, etc.
- c) *Probabilidad de error residual*, es decir, de errores no detectados y, por tanto, no corregidos producidos en la conexión de transporte
- d) *Probabilidad de desconexión espontánea*
- e) *Tiempo máximo de establecimiento / liberación* de la conexión de transporte

Con el fin de determinar la complejidad necesaria en la capa de transporte, OSI clasifica las redes en tres tipos:

- a) Tipo A: redes prácticamente perfectas, sin ningún tipo de errores
- b) Tipo B: redes con una tasa de errores *señalizados* inaceptable
- c) Tipo C: redes con una tasa de errores *residuales* (y quizás también señalizados) inaceptable

Los *errores señalizados* son aquellos que, aunque no corregidos, son detectados por la capa de red y notificados (señalizados) a su capa usuaria. Son típicamente los reinicios y liberaciones espontáneos en los circuitos virtuales de las redes orientadas a conexión.

Los *errores residuales*, como su nombre indica, son aquellos no detectados (que pasan desapercibidos) por la capa de red. Son, típicamente, la corrupción de algunos bits en los datagramas, la pérdida, desordenación, e incluso duplicación, de datagramas en las redes no orientadas a conexión.

En función del tipo de redes con las que tratar, la capa de transporte OSI define cinco niveles de realización de complejidad creciente; son las llamadas *clases de transporte*:

- a) Clase 0 (*Simple Class*): la más sencilla; no recupera errores; para redes tipo A
- b) Clase 1 (*Error Detection Class*): recupera errores señalizados; para redes tipo B
- c) Clase 2 (*Multiplexing Class*): como la clase 0 pero incorporando mecanismos de multiplexado
- d) Clase 3 (*Error Detection and Multiplexing Class*): como la clase 1 pero incorporando mecanismos de multiplexado
- e) Clase 4 (*Error Detection and Recovery Class*): recupera todo tipo de errores; de uso obligatorio para redes de tipo C

En la arquitectura TCP/IP, el protocolo TCP posee funciones equivalentes a la clase 4 de transporte OSI, aunque con algunas diferencias. Asimismo, en dicha arquitectura, los números de *puerto* hacen las funciones equivalentes de los T-SAP de OSI.

Para ofrecer el refinado servicio de transporte definido, dicha capa incorpora un elevado número de mecanismos (en los que no se abundará aquí). Algunos de ellos son muy parecidos a los de la capa de enlace de datos: adición de bits de redundancia (*checksum*) para detectar errores de bit; uso de números de secuencia, mecanismos de asentimiento y retransmisión para detectar pérdidas, desordenamientos y duplicaciones en las T-PDU; uso de créditos o ventanas de transmisión para mejorar la eficiencia y controlar el flujo de datos, etc.

Otros mecanismos son exclusivos de la capa de transporte. Por ejemplo: *segmentación/reensamblado* (para poder enviar T-PDU de longitudes superiores al tamaño máximo de las N-PDU -paquetes, datagramas- en las redes subyacentes); *concatenación/separación* (para enviar más de una T-PDU dentro de una única N-PDU aumentando la eficiencia en el uso de las redes); etc. Se comentarán, a modo de ejemplo, los mecanismos de multiplexado definidos en la capa de transporte:

- a) *Varias conexiones de transporte en una conexión de red* (optimiza el uso de las redes cuando varias conexiones de transporte tienen el mismo origen y destino).

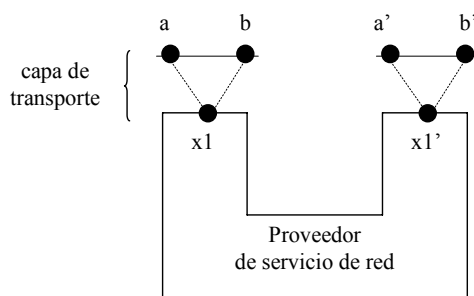


Fig. 2.15

- b) *Una conexión de transporte en varias conexiones de red* (permite alcanzar los niveles de QoS acordados; en efecto, el uso de varias conexiones de red “en paralelo” permite conseguir un caudal mayor o tener mayor redundancia para disminuir el retardo o la probabilidad de desconexión de la conexión de transporte).

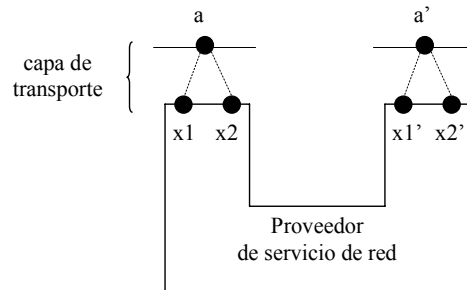


Fig. 2.16

#### 2.4.7 La capa de sesión

La capa de sesión fue una aportación relativamente nueva del modelo OSI de la ISO a las arquitecturas de comunicaciones. En efecto, podría pensarse que, dado el servicio perfectamente fiable que ofrece la capa de transporte, los procesos de aplicación no necesitarían nada más y podrían usar dicho servicio directamente.

De hecho, y siguiendo este razonamiento, la arquitectura TCP/IP carece de las capas de sesión y presentación explícitamente. Superados los errores de comunicación en la capa de transporte, la capa de sesión puede verse como un conjunto de herramientas, a disposición de los programadores, que permiten estructurar y enriquecer el diálogo entre los procesos de aplicación.

Algunos de los servicios que dicha capa ofrece son:

- a) Establecimiento, mantenimiento y finalización de las sesiones
- b) Gestión del diálogo y las actividades
- c) Sincronización y recuperación
- d) Gestión de los permisos (*tokens*) para realizar ciertas acciones
- e) Cierre ordenado de las conexiones

Una conexión de sesión (sesión) se soporta sobre una conexión de transporte, pero va más allá. Así, por ejemplo, una sesión puede prolongarse más allá de una conexión de transporte o viceversa, como se ilustra en la figura 2.17.

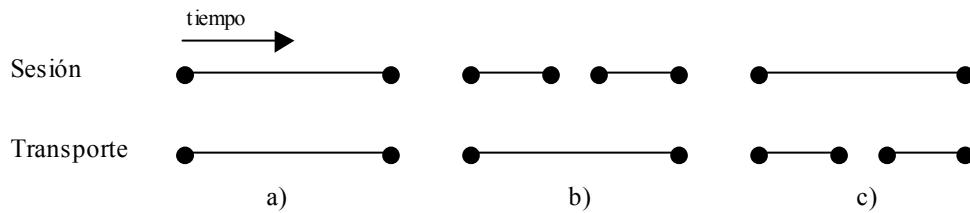


Fig. 2.17 Una sesión es algo más refinado que una conexión de transporte: a) una sesión en una conexión de transporte; b) varias sesiones en una misma conexión de transporte; c) una sesión en varias conexiones de transporte

Aunque las conexiones de transporte se definen como *full-duplex*, una sesión puede estructurarse como *half-duplex*, es decir, que existan turnos en las transmisiones o que, en un momento dado, sólo uno de los extremos de la sesión esté facultado para enviar datos. Para ello, la capa de sesión ofrece la posibilidad de usar el testigo (*token*) de transmisión que se posee, se pide y se cede por parte de los extremos como se ilustra en la figura 2.18.

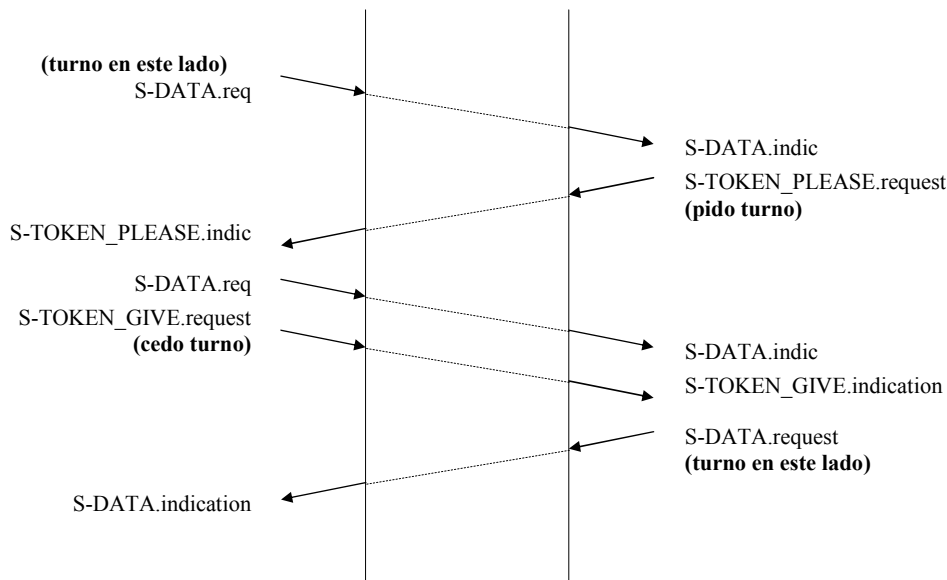


Fig 2.18 Ejemplo de gestión del modo half-duplex mediante el uso del testigo de transmisión

La capa de sesión también ofrece mecanismos para sincronizar o “marcar” el diálogo entre sus usuarios a través de la inserción de puntos de sincronización de los que existen dos tipos: mayores (*major sync points*) y menores (*minor sync points*).

Mediante la inserción de esos puntos la sesión se estructura en *diálogos* (aquella porción comprendida entre punto de sincronización mayor consecutivos) y *actividades* (que comprenden varios puntos de sincronización mayor). Las actividades pueden ser iniciadas, acabadas e incluso suspendidas (para continuarlas posteriormente). La utilización o no de estas facilidades, así como su significado, no son competencia de la capa de sesión, sino de los procesos de aplicación en dicha capa. La figura 2.19 ilustra todo esto.

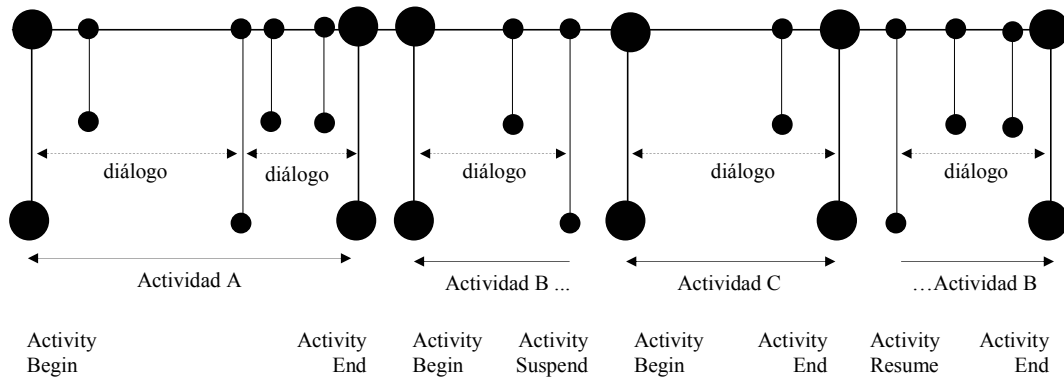


Fig. 2.19 Inserción de puntos de sincronización en una sesión

Por *resincronización* de la sesión se entiende la conducción de ésta (su *status* completo) a un punto o estado anterior identificado por algún punto de sincronización. Como mucho se puede retroceder no más allá del punto de sincronización mayor anterior. El uso y utilidad de la *resincronización*, como siempre, depende del proceso de aplicación. Podría servir para recuperar una situación de error originada por la aplicación (no un error de comunicación ya que éste sería solventado por la capa de transporte).

A modo de ejemplo, imagínese la impresión de un libro en un periférico remoto (impresora de alta calidad, no disponible localmente); para ello la sesión "impresión del libro" se estructura con puntos de sincronización menor para cada página y puntos de sincronización mayor para cada sección o capítulo; imaginemos que la impresora se queda sin tinta y que el operador del sistema donde se encuentra la impresora se da cuenta de ello, procediendo a sustituir el cartucho de tinta agotado por uno nuevo cuando ya han salido varias páginas defectuosas o en blanco.

En esta situación, la capa de transporte no puede solventar ese problema originado por el proceso de aplicación impresora remota; dicha capa ha cumplido perfectamente su misión de enviar datos fiablemente de un extremo a otro y no debe ni puede determinar que dichos datos no se han procesado adecuadamente en la aplicación destino impresora remota; las funciones de *resincronización* que ofrece la capa de sesión, en cambio, permitirían reanudar el proceso de impresión a partir del punto donde se inició el fallo elegantemente.

La capa de transporte OSI define un cierre de la conexión *abrupto*, es decir, cualquiera de los usuarios de la conexión puede iniciar una desconexión sin tener en cuenta el otro extremo ni los datos que pueda haber en tránsito.

La capa de sesión ofrece mecanismos para el cierre ordenado de las conexiones (*graceful close*), es decir, sin ambigüedades o pérdida de datos. En la arquitectura TCP/IP, en cambio, dicho cierre ordenado ya está disponible en la capa de transporte (en el protocolo TCP). La figura 2.20 ilustra el concepto de cierre ordenado en donde implícitamente se definen sesiones abiertas, sesiones medio cerradas (es decir, que el extremo cerrado no envía más datos pero sí puede recibirlos) y sesiones cerradas.

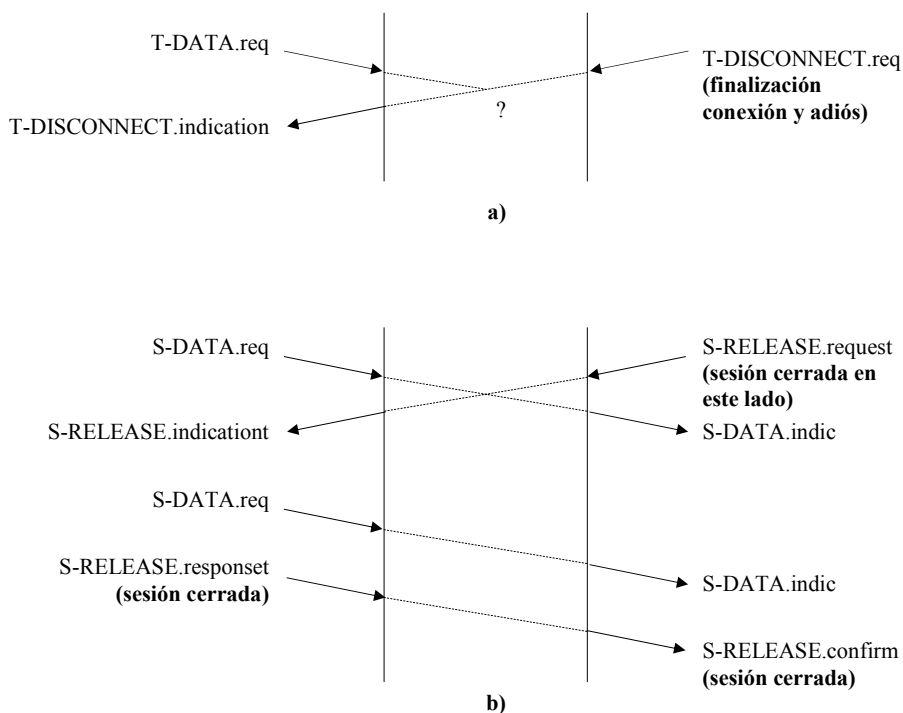


Fig. 2.20 a) Cierre abrupto de una conexión de transporte; b) cierre ordenado de una conexión de sesión

El nivel de sesión OSI ofrece hasta 4 tipos de *flujos de datos* distintos:

- Normal data*: los datos normales intercambiados principalmente en una sesión
- Typed data*: pequeñas cantidades de datos que pueden transferirse sin la posesión del testigo de transmisión
- Capability data*: pequeñas cantidades de datos que se pueden transferir fuera de las actividades para funciones de control
- Expedited data*: datos con prioridad respecto a los normales; basado en los servicios de datos urgentes de las capas inferiores de transporte y red (si están disponibles)

### 2.4.8 La capa de presentación

La capa de presentación está relacionada con el significado (semántica) y formato (sintaxis) de los datos intercambiados en una sesión entre procesos de aplicación. Si la capa de transporte ofrece un intercambio de *datos* fiable independiente de las redes y la capa de sesión estructura/enriquece ese intercambio de *datos*, la capa de presentación ofrece un servicio de intercambio de *información*, es decir, a nivel de dicha capa no se intercambian meramente grupos de bytes, sino algo con significado (por ejemplo: textos escritos, vectores o matrices de números enteros o reales, imágenes, sonidos, etc.).

Se suele decir que el nombre más adecuado para la capa de Presentación hubiera sido el de capa de representación. Si el modelo OSI perseguía la interconexión de sistemas que fueran realmente abiertos, debía resolverse el problema de los diferentes formatos con que cada sistema representaba localmente su información (dependiente del fabricante del sistema, de su *hardware* y de su sistema operativo). Es ahí donde interviene la capa de presentación.

El modelo OSI asigna a la capa de presentación tres funciones:

- a) Representación común de la información (formatos, conversiones)
- b) Seguridad en el intercambio de información (privacidad, protección, autenticación)
- c) Compresión de la información

Para ilustrar mejor la necesidad de la capa de presentación véanse algunos ejemplos: los sistemas de la marca IBM han venido usando el código EBCDIC para representar la información textual, mientras que la mayor parte de los demás sistemas usan el código ASCII. En esta situación es evidente que, si un sistema IBM intercambia un texto con otro sistema, se pierde completamente su significado. Algunos sistemas almacenan los números enteros en su memoria o disco con el bit más significativo en primer lugar, mientras que otros lo hacen con el bit menos significativo en primer lugar; no digamos ya, la manera en que cada sistema representa a los números reales (¿cuántos bits para la mantisa?, ¿cuántos para el exponente?, ¿en qué formato?) o cómo lo hace para estructuras de datos más complejas (vectores o matrices de números, registros de una base de datos, etc).

Al abordar el problema de conseguir la deseada *transparencia* en el intercambio de información entre sistemas por parte de la capa de presentación cabe plantearse tres alternativas:

- a) Conversión del formato de origen al formato de destino en el sistema destino (una vez recibida la información)
- b) Conversión del formato de origen al de destino en el sistema origen (antes de mandar la información)
- c) Conversión a un formato común (antes de mandar la información) y conversión al formato de destino (una vez recibida la información).

Las dos primeras alternativas suponen que cada sistema debe tener algoritmos de traducción a/de cada uno de todos los posibles formatos que pueda usar cualquier sistema de cualquier fabricante. Si  $N$  es el número de formatos distintos existentes, el número de algoritmos necesarios crece con  $N^2$  y, además, la introducción de sistemas que usen un nuevo formato de representación de su información obliga a una actualización en todos los sistemas en comunicación.

La tercera alternativa es la adoptada en la capa de presentación OSI, pues requiere sólo  $2N$  algoritmos de conversión y la adición de sistemas con nuevos formatos sólo requiere actualizaciones locales en dichos nuevos sistemas. Las figuras 2.21 y 2.22 ilustran esquemáticamente esas ideas.

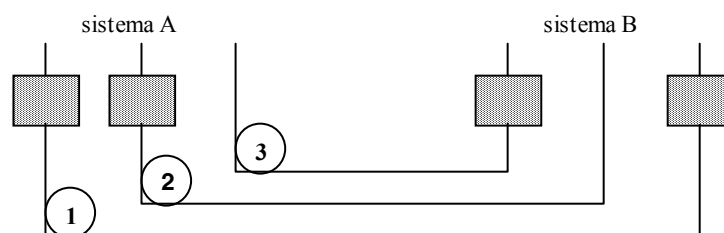


Fig. 2.21 Alternativas para lograr la transparencia en la capa de presentación: 1) conversión a/de un formato común; 2) conversión en origen al formato de destino; 3) conversión en destino del formato origen

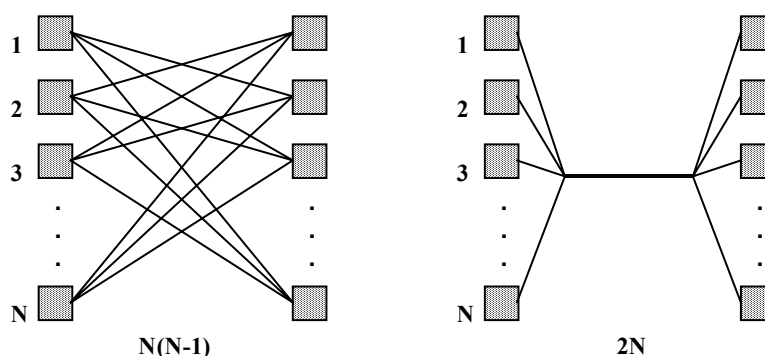


Fig. 2.22 Conversiones de formato necesarias, dados  $N$  formatos distintos, para las distintas alternativas

Asociada a la capa de presentación, existen los conceptos de *sintaxis abstracta* y *sintaxis de transferencia*.

La *sintaxis abstracta* se utiliza para definir de forma simbólica cualquier estructura de datos susceptible de ser intercambiada entre procesos de aplicación de sistemas distintos. Para ello se usa el lenguaje ASN.1 (*Abstract Syntax Notation 1*). Dicho lenguaje permite definir cualquier estructura de datos a partir de la definición de unos tipos de datos primitivos o básicos (INTEGER, BOOLEAN, BIT STRING, OCTET STRING, ANY, NULL; OBJECT IDENTIFIER) y unos constructores (SEQUENCE, SEQUENCE OF, SET, SET OF, CHOICE), siendo similar a lo que ofrecen lenguajes de programación como PASCAL o C.

La *sintaxis de transferencia*, por otra parte, describe cómo se codifica y transfiere cualquier estructura descrita mediante ASN.1 (estructura de campos, orden y significado de cada bit en ellos, etc). Pueden existir diversas sintaxis de transferencia, dependiendo, por ejemplo, de si se utiliza reducción de redundancia (compresión), cifrado, etc. La figura 2.23 ilustra las anteriores ideas.

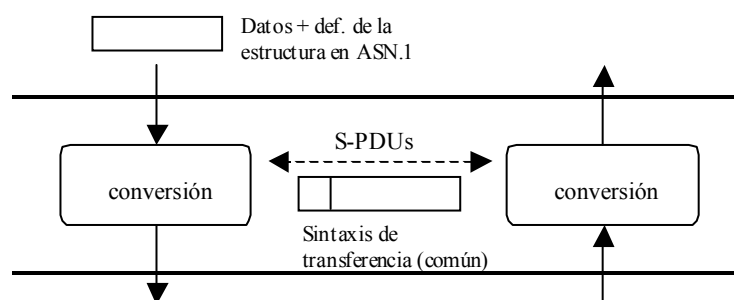


Fig. 2.23 Operación de la capa de presentación para conseguir la transparencia deseada

Otro concepto clave en la capa de presentación OSI es el de *contexto*. Por contexto se entiende la descripción de todas las estructuras de datos, mediante ASN.1, que se utilizarán en una conexión entre procesos de aplicación, así como la sintaxis de transferencia que se aplicará para ellas. Los contextos se negocian durante el establecimiento de una conexión de presentación y pueden también alterarse durante ella. Así, por ejemplo, una aplicación de cálculo numérico distribuida utilizará un contexto claramente distinto al de una aplicación de acceso a una base de datos.

Los servicios que ofrece la capa de presentación, aparte de las funciones de conversión de sintaxis para mantener la semántica de la información y de las funciones de compresión o cifrado, son los mismos que ya ofrecía la capa de sesión. Los servicios que una capa transfiere sin modificación alguna, desde su capa inferior a su superior, suelen denominarse servicios “reflejados” (*mirroring services*) o servicios de “paso directo” (*pass through services*), como se representa simbólicamente en la figura 2.24.

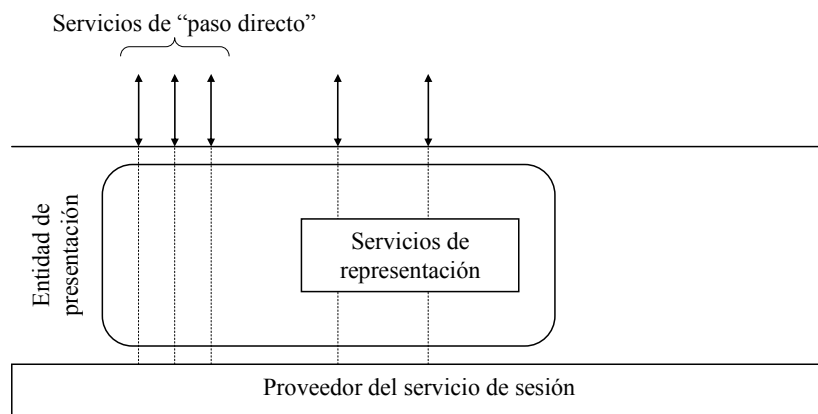


Fig. 2.24 La capa de presentación "refleja" los servicios de la capa de sesión, excepto aquellos que impliquen aspectos de representación de la información

Si bien el modelo OSI establece que las funciones de compresión y cifrado deben realizarse en la capa de presentación, ello no debe necesariamente ser siempre así en una arquitectura de comunicaciones. Así, por ejemplo, podría concebirse una arquitectura en donde la capa de enlace de datos cifrara y/o comprimiera las L-PDU (tramas). Las implicaciones de una u otra alternativa pueden ser importantes: con cifrado/compresión en la capa de enlace de datos, las PCI (cabeceras) de la capa de red y superiores (transporte y sesión) resultan afectadas, mientras que con dichas funciones realizadas en la capa de Presentación, no. Asimismo, la primera alternativa implica dotar a los sistemas intermedios de funciones de compresión/cifrado, mientras que la alternativa de la capa de presentación sólo implica a los sistemas finales.

Un ejemplo típico de funcionalidad aplicada a la capa de presentación es el de la criptografía. Veamos a continuación algunas pinceladas sobre esta ciencia.

### ***Nociones básicas de criptografía***

Los objetivos básicos de las técnicas de criptografía son:

- a) Ocultar la información (evitar que pueda ser leída por entidades ajenas o no autorizadas)
- b) Proteger la información (evitar que agentes externos puedan modificar, borrar o insertar partes)
- c) Verificar el emisor de un mensaje (autenticación, no repudio).

En la figura 2.25 se representa el escenario general donde se aplican las técnicas criptográficas.

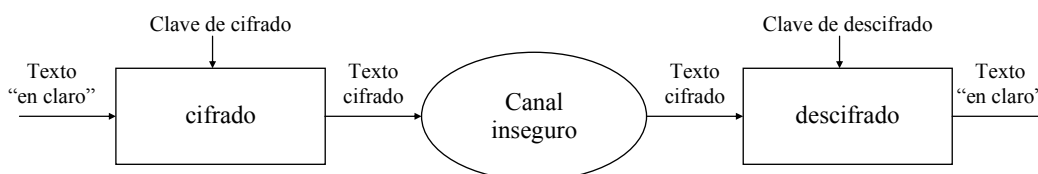


Fig. 2.25 Escenario general donde se aplican las técnicas de cifrado

En sus inicios la fortaleza de un sistema criptográfico se basaba en el secreto del algoritmo.

Actualmente, los algoritmos criptográficos son estándares públicos, es decir, conocidos por todos, estando confiada su potencia al secreto y tamaño de la clave y al hecho de que los algoritmos sean robustos, en el sentido de que no existan “atajos” para descifrar el texto y sólo quepa probar todas las claves posibles (método de la *fuerza bruta*).

Las técnicas clásicas se basan en el uso de *substituciones* o de *transposiciones*.

Por *substitución* se entiende el reemplazo de cada carácter por otro o por un grupo de ellos. Entre esas técnicas pueden citarse, por ejemplo:

- a) Aplicación de una *máscara XOR* a cada carácter. El algoritmo lo constituye la función XOR (O-exclusivo), la clave la constituye la máscara. Así, por ejemplo, si la clave (máscara) es “11001110” (“CE”<sub>hex</sub>) y el texto a cifrar es “CASA” (“43 41 53 41”<sub>hex</sub>), el texto cifrado sería “8D 8F 9D 8F”<sub>hex</sub>. El algoritmo de descifrado para recuperar el texto original es el mismo, es decir, aplicar de nuevo la función XOR con la clave a cada octeto. En este ejemplo, conocido el algoritmo, el número de claves posibles es de sólo  $2^8$ , lo que da una idea de su poca seguridad. Sin embargo, dicho algoritmo se utilizó hace años para codificar los documentos de texto por parte de un conocido fabricante de *software* que confiaba en el secreto del algoritmo.
- b) Uso de una *tabla de traducción* (variante del método utilizado por el emperador romano César Augusto para intercambiar partes de guerra). Así, por ejemplo, dada la tabla de traducción siguiente (Q, M, U, G, X, J, P, R, A, H, W, ...) que indica que la primera letra del abecedario debe sustituirse por la Q, la segunda por la M y así sucesivamente, el texto “BECA” resultaría “MXVQ”; puede verse fácilmente que el número posible de tablas distintas, es decir, de claves, es el factorial del número de letras del abecedario (27!). Dicho algoritmo es algo más robusto, aunque con los ordenadores actuales puede vulnerarse fácilmente.

Por *transposición* se entiende el cambio en el orden de los octetos del mensaje a cifrar según indique una clave. Así, por ejemplo, dada la clave (3, 1, 4, 2) que indica las transposiciones a realizar en el texto a cifrar, tomado en bloques de cuatro octetos, de manera que el tercer octeto de cada bloque se coloca en primer lugar, el primer octeto en segundo lugar, etc., el texto “MI\_ORDENADOR” resultaría “\_MOIERNDOARD”.

### El método de cifrado simétrico DES (Data Encryption Standard)

El método de cifrado DES propuesto en USA divide el texto a cifrar en bloques de 64 bits sobre los que se aplican elaboradas técnicas de sustitución y transposición repetidamente a partir de la información que contiene una clave de 56 bits. El método ha demostrado ser robusto y existen mejoras basadas en el uso de claves de mayor longitud, el cifrado encadenado de bloques, etc. Existen circuitos integrados para llevar a cabo los algoritmos DES de forma rápida y eficiente. Este método de cifrado se llama *simétrico* porque utiliza la misma clave para cifrar y descifrar. Su inconveniente es el problema de la distribución de claves; en efecto, el emisor debe mandar al destinatario la clave usada en el cifrado, ¡pero ello no puede hacerse por la misma red que se considera insegura y para la cual se utiliza el cifrado!; las claves deben enviarse, por tanto, por otro medio que pueda considerarse seguro (envío personal, en mano, por ejemplo). En muchos casos, tampoco se desea que el receptor de una información que se manda cifrada conozca la clave que el emisor utiliza para todos sus envíos, pues podría ser suplantado por aquel. En estos casos es más conveniente usar métodos de cifrado asimétricos o de clave pública, como se describe a continuación.

### Criptografía de clave pública o cifrado asimétrico

Uno de los problemas más importantes de las técnicas de criptografía es el de la distribución de las claves. Si un emisor sospecha que algún intruso conoce su clave y desea cambiarla, éste no puede utilizar la propia red insegura para mandar su nueva clave al destinatario. Tampoco es deseable, en muchas ocasiones, que el destinatario conozca la clave que el emisor utiliza para todos sus envíos.

Los algoritmos de clave pública se basan en que cada usuario que desea intervenir en un entorno de intercambio seguro de información posee una clave de dos partes: su clave pública  $K_p$  y su clave privada o secreta  $K_s$ . Un usuario puede fácilmente generar (o solicitar a un organismo administrador de claves de confianza) pares de claves ( $K_p$ ,  $K_s$ ). La clave pública  $K_p$ , como indica su nombre, es depositada en una base de datos al alcance de cualquier otro usuario, mientras que la clave privada  $K_s$  es mantenida en secreto por su usuario y nunca viajará por la red. Aunque, obviamente, existe una relación entre  $K_p$  y  $K_s$ , es extremadamente difícil deducir  $K_s$  a partir de  $K_p$ , es decir

$$K_s \neq f(K_p) \quad (2.5)$$

Además se cumple que los algoritmos de cifrado empleados (en adelante AE) son asimétricos, es decir, no reversibles fácilmente. La reversibilidad estará condicionada a la utilización de la otra parte del par que constituye una clave. Supongamos que un usuario  $A$  desea mandar un mensaje  $M$  (en claro) a otro usuario  $B$  de manera que nadie, excepto el destinatario pueda entender su contenido. Para ello el usuario  $A$  cifra el mensaje  $M$  usando la clave pública  $K_{pB}$  del destinatario, obteniendo el mensaje cifrado  $X$  que viajará por la red, es decir,

$$X = AE(K_{pB}, M) \quad (2.6)$$

Nadie podrá descifrar dicho mensaje (el proceso de cifrado con  $K_{pB}$  no es reversible), es decir,

$$M \neq AE'(K_{pB}, X) \quad (2.7)$$

excepto quien posea la otra parte de esa clave, es decir  $K_{sB}$ , es decir, el destinatario

$$AE(K_{sB}, X) = M \quad (2.8)$$

Estos algoritmos se llaman *asimétricos*, dado que la clave para cifrar y descifrar son distintas. Cualquier usuario distinto de B no podrá entender, descifrar, el mensaje X. Además, si el usuario B desea cambiar de claves (porque sospecha que alguien haya obtenido su clave privada o simplemente por rutina de seguridad) no tiene más que generar un nuevo par de claves ( $K'_{pB}$ ,  $K'_{sB}$ ) y distribuir por la red sin restricciones  $K'_{pB}$ .

En la figura 2.26 se esquematiza el proceso anteriormente explicado.

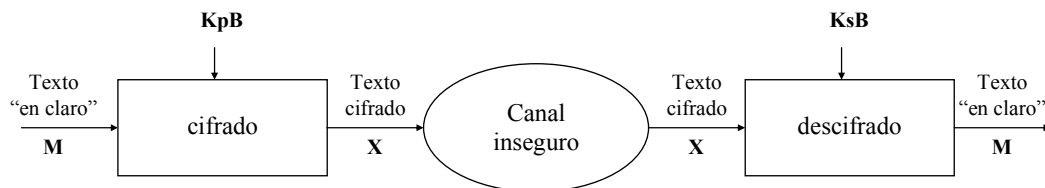


Fig. 2.26 Escenario del envío de un mensaje con confidencialidad

En el último tercio del siglo pasado se descubrieron algoritmos con las peculiares características antes citadas. A grandes rasgos, éstos se basan en la generación de pares de números primos muy grandes y en la aplicación de la función exponencial. Su coste computacional es relativamente elevado y sólo se usan donde son imprescindibles o combinados con los algoritmos simétricos de tipo DES, por ejemplo.

Pero dichos algoritmos aún permiten más; por ejemplo, implementar mecanismos de autenticación o de *firma digital*. Para ello, basta que además de

$$AE(K_s, AE(K_p, M)) = M \quad (2.9)$$

cumplan también, como así lo hacen, que

$$AE(K_p, AE(K_s, M)) = M \quad (2.10)$$

En estas condiciones un usuario A puede enviar un mensaje *autenticado* o *firmado digitalmente*, es decir, que solamente puede provenir de éste y además éste no puede retractarse de su generación (principio de *no repudio*), cifrándolo con su clave secreta  $K_{sA}$ . El mensaje ahora no es secreto, pues puede descifrarlo cualquier usuario usando la clave pública de su emisor  $K_{sA}$ ; ahora bien, el mensaje sólo se descifra correctamente si fue cifrado con  $K_{sA}$ , es decir, si es auténtico.

Si además, el usuario A desea mandar un mensaje firmado F a otro usuario B, de manera que sólo el destinatario pueda leerlo, procede a cifrarlo de nuevo usando la clave pública del destinatario, como ya se vio anteriormente, es decir:

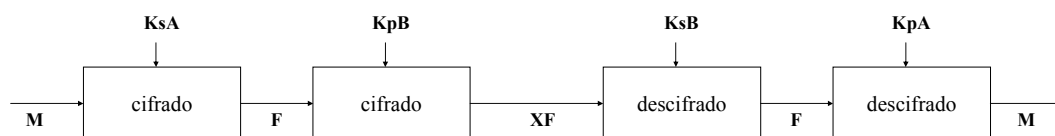


Fig. 2.27 Escenario del envío de un mensaje firmado y confidencial

Los sistemas de cifrado en las redes cada vez tienen más importancia en la llamada sociedad de la información. Algunos ejemplos de herramientas de seguridad, que combinan algunas de las técnicas descritas con otras nuevas, son: *Kerberos*, *SSL (Secure Socket Layer)*, *PGP (Pretty Good Privacy)*, etc.

### Noción de compresión de la información

El objetivo de usar técnicas de compresión es el de reducir el número de octetos con el que se representa la información. Sus ventajas son: reducción de los costes de uso de las redes, aumento del rendimiento (mayor velocidad de transferencia en redes lentas, mejor aprovechamiento del ancho de banda disponible) y reducción de los errores (al disminuir el número de octetos a transmitir).

Pueden distinguirse varios tipos de compresión:

- a) *Basada en los contenidos* (semántica) de las fuentes de datos: tienen en cuenta el tipo de información que representan los datos con lo que pueden aplicarse las técnicas de reducción más adecuadas en cada caso.
- b) *Independiente* de los contenidos de las fuentes de datos o *genérica*: analizan el conjunto de octetos, sin tener en cuenta qué representan, e intentan representarlos unívocamente con un menor número de ellos.

En íntima relación con los dos tipos anteriores, también se habla de:

- a) *Compresión con pérdidas*: es decir, la información que se recupera tras someterse al proceso de compresión no es idéntica a la original, pero es suficiente para los propósitos establecidos. Típicamente se usa en la compresión de sonidos e imágenes fijas o en movimiento; para este tipo de información, cuyo destinatario es el oído u ojo humano, se admiten ciertas degradaciones en la calidad que toman ventaja de las peculiares características (o imperfecciones) de los sentidos humanos. La compresión con pérdidas está, en general, claramente asociada con la compresión basada en los contenidos y permite índices de compresión elevados (del orden de 10:1 hasta 50:1). Algunos ejemplos para este tipo son: los codificadores GSM o MP3 para audio y los algoritmos GIF, JPEG, MPEG para imágenes y video.
- b) *Compresión sin pérdidas*: es decir, la información recuperada tras el proceso de compresión es exactamente igual a la original. Se usa normalmente en la compresión de textos, de código

ejecutable, etc. Dicha compresión está asociada a algoritmos genéricos o independientes de las fuentes y se obtienen índices de compresión generalmente menores. Algunos ejemplos para este tipo son: los códigos de Huffman, de Lempel-Ziv, los conocidos ficheros .ZIP, etc.

## 2.4.9 La capa de Aplicación

La capa de aplicación es la más superior del modelo arquitectónico y ofrece los llamados *servicios de aplicación* e incluso las *aplicaciones* más comunes y *estandarizadas* de forma completa. Al principio del capítulo se comentó que el objetivo de toda arquitectura de comunicaciones era poner en contacto a los procesos de aplicación.

Un *proceso de aplicación* es aquella parte de un sistema final que procesa información en un entorno de red OSI y que es invocado por un usuario o proceso local o remoto; suele dividirse conceptualmente en dos partes: el *agente de aplicación* (aquella parte del proceso de aplicación fuera del entorno OSI, dependiente del sistema local y que puede actuar como interfaz con el usuario u otros procesos internos) y la *entidad de aplicación* (aquella parte que pertenece a la capa de aplicación OSI).

La capa de aplicación puede llegar a ser muy compleja y en el modelo OSI se considera formada por los llamados elementos de servicio de aplicación (en adelante ASE: *Application Service Element*). De estos se distinguen: los CASE (*Common ASE*), módulos que ofrecen funcionalidades genéricas a un programador de aplicaciones en red, por ejemplo, y los SASE (*Specific ASE*), módulos destinados a construir aplicaciones concretas estandarizadas en la capa de aplicación.

Algunos CASE son:

- a) ACSE (*Association Control Service Element*): el más básico, pues aporta la funcionalidad básica para las conexiones a nivel de aplicación, llamadas *asociaciones*. Nótese que la palabra sugiere una interacción al más alto nivel que hereda todos los refinamientos aportados por cada una de las capas del modelo arquitectónico; es decir, una asociación es una conexión altamente refinada que permite el intercambio transparente de información (presentación) de forma segura y autenticada (presentación), eficiente (transporte, presentación -en su faceta de compresión-), fiable (transporte, enlace de datos), estructurada (sesión) entre cualquier sistema final (red), etc.
- b) RTSE (*Reliable Transfer Service Element*): obsoleto, diseñado para la transferencia de grandes volúmenes de información y antes de que las capas de sesión y presentación estuvieran completamente desarrolladas.
- c) ROSE (*Remote Operations Service Element*): para operaciones de tipo cliente-servidor.
- d) CCR (*Commitment, Concurrency and Recovery*): para el soporte de las llamadas *acciones atómicas* (es decir, indivisibles). Una acción atómica es aquella que implica a dos o más entidades, que debe desarrollarse sin interferencia de entidades ajenas y, lo más importante, que debe realizarse en su totalidad o no realizarse en absoluto. En una acción atómica existe un maestro (que supervisa la realización de toda la acción) y uno o más esclavos y se establecen varias fases. En la primera fase, el maestro solicita a los esclavos las acciones a realizar y estos responden si están en condiciones de realizarlas (compromiso); si, y sólo si, todos los esclavos

responden afirmativamente, el maestro ordena la segunda fase, la de ejecución concurrente de las acciones por parte de los esclavos, que deberán responder cuando su acción se haya llevado a término; la tercera fase supone el fin exitoso de la acción atómica o la recuperación, es decir, el deshacer la acciones realizadas por los esclavos en caso de que alguno de ellos haya fallado en su consecución (para ello, durante la fase de compromiso, los esclavos habrán almacenado en lugar seguro el *status* de la situación de partida para poder regresar a ella en caso necesario y habrán bloqueado el acceso de otras entidades ajenas a la acción atómica). Es fácil imaginar situaciones en donde el concepto de acción atómica es útil; por ejemplo, en las transacciones bancarias u operaciones en cajeros electrónicos.

Los SASE están orientados a ofrecer funciones más específicas relacionadas con una aplicación completa usual y estandarizada. Algunas de las aplicaciones estandarizadas por OSI en la capa de aplicación son:

- a) FTAM (*File Transfer, Access and Management*)
- b) E-MAIL (*Electronic Mail*)
- c) VT (*Virtual Terminal*)
- d) DS (*Directory Services*)
- e) JTM (*Job Transfer and Management*).

A continuación se presentaran sucintamente las aplicaciones de red más comunes haciendo referencias al enfoque que propone la arquitectura OSI como al de otras arquitecturas como la TCP/IP.

### ***Aplicaciones de transferencia y compartición de archivos***

En la arquitectura TCP/IP, FTP (*File Transfer Protocol*) es una aplicación para el acceso a un sistema de ficheros en un sistema remoto y la transferencia de ficheros entre sistemas. Se trata de una aplicación relativamente antigua, sencilla e íntimamente ligada a los sistemas operativos de tipo UNIX. Permite la ejecución de los comandos habituales de acceso y gestión a sistemas de ficheros (listar, cambiar de directorio, borrar, etc.), así como la transferencia de archivos o grupos de archivos.

La aplicación no ofrece apenas nivel de abstracción en el sentido de que es bastante dependiente de los sistemas de ficheros (tipo UNIX, aunque también se haya adaptado a los de tipo Windows); tampoco garantiza el mantenimiento del significado del contenido de los ficheros, al pasar éstos de un sistema a otro, pues dichos ficheros son tratados de forma transparente por la aplicación. La aplicación consta de las entidades cliente ftp y servidor ftp, que se comunican mediante el protocolo FTP, y usa dos conexiones de transporte TCP, una para el control de la sesión (transferencia de comandos y respuestas) y la otra para la transferencia de archivos, habitualmente en los puertos 20 y 21.

También asociada a la arquitectura TCP/IP, NFS (*Network Filing Systems*) es una potente y robusta (a pesar de ser enteramente no orientada a conexión) aplicación para la compartición de ficheros entre múltiples sistemas en un entorno generalmente local. Su filosofía de operación se basa en que: un sistema puede *exportar* una parte o la totalidad de su sistema de ficheros (es decir, hacerlo visible y

accesible, a través de la red, a otros sistemas); un sistema puede *montar* localmente sistemas de ficheros exportados por otros (es decir, incorporarlos a su sistema de ficheros y verlos como propios).

La transparencia de este proceso es casi total, en el sentido de que se pueden realizar operaciones de abertura y acceso para lectura y/o escritura (*open, read, write*) de ficheros remotos como si fueran locales; aquí radica la gran diferencia respecto a la aplicación FTP comentada anteriormente. Por otra parte, la transparencia en cuanto a la semántica de los ficheros no es considerada, es decir, todos los sistemas deben estar de acuerdo en la estructura interna y significado de los datos de los ficheros que comparten.

La propuesta de la arquitectura OSI es FTAM (*File Transfer, Access and Management*), una aplicación completa que pretende cubrir todos los aspectos relativos a sistemas de ficheros de sistemas finales en un entorno de red abierto. Su filosofía gira en torno al concepto de sistema de ficheros virtual (en adelante VFS: *Virtual File System*), es decir todas las operaciones relacionadas con ficheros entre sistemas abiertos se realizan sobre ese VFS o, en otras palabras, los sistemas de ficheros, o partes de éstos, que un sistema desea compartir en red se ven como un VFS, independientemente de sus particularidades locales.

La definición del VFS es rica y compleja y describe la estructuración de un conjunto de ficheros (estructura en árbol, directorios, enlaces entre partes del árbol, atributos de directorios, etc.), así como los ficheros (atributos, estructura interna, etc.). En la figura 2.28 se puede ver la estructura genérica de un fichero según FTAM; dicha estructura permite representar la mayoría de ficheros reales definidos por cualquier sistema operativo.

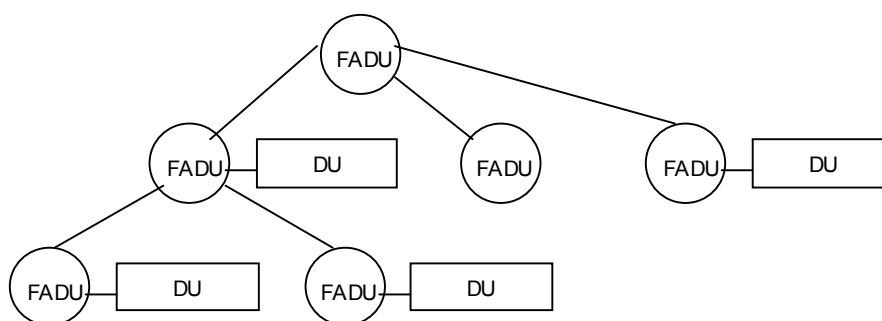


Fig. 2.28 Los diversos puntos de acceso al fichero, cuya estructura genérica es de árbol, se describen por los FADU (File Access Data Unit). Asociados a cada uno de estos puntos pueden existir o no datos, los DU (Data Unit)

FTAM define asimismo las operaciones que pueden realizarse sobre el VFS. Los distintos grupos de operaciones definen diversos *regímenes* de operación posibles (o grados de interacción) para sistemas que se relacionan mediante FTAM. Muy sucintamente, la figura 2.29 representa los diferentes regímenes y algunas primitivas de servicio asociadas a ellos.

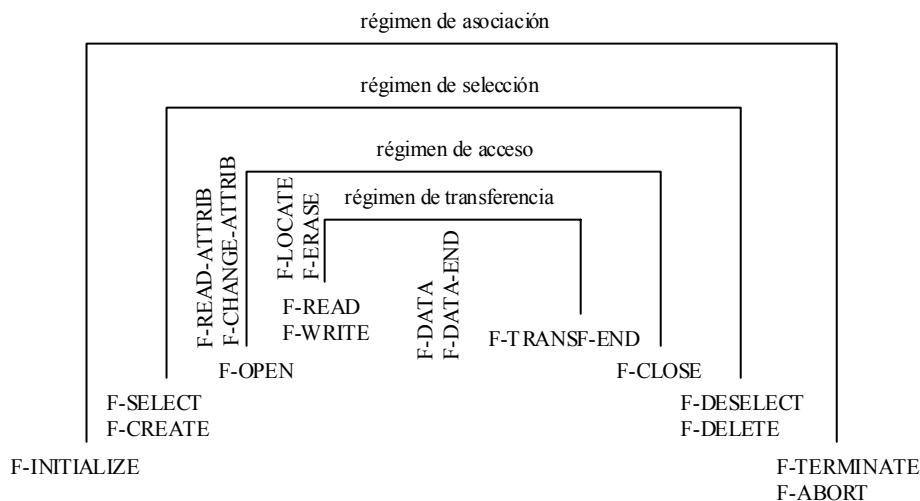


Fig 2.29 Los regímenes de operación en FTAM y algunas primitivas de servicio asociadas

Como ya se ha dicho, el VFS y sus operaciones describen como se ve el sistema de archivos de un sistema abierto en un entorno OSI. Es evidente que ello implica que existan mecanismos de traducción del sistema de ficheros real y local al sistema VFS más abstracto y remoto, como ilustra la figura 2.30.

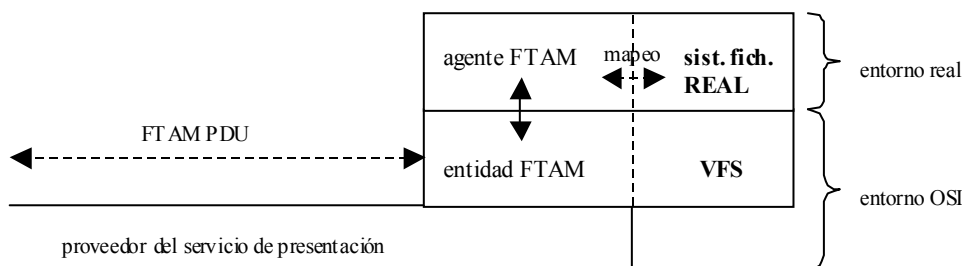


Fig. 2.30 Las operaciones sobre el sistema de ficheros virtual (VFS) se "mapean" o traducen en el correspondiente sistema de ficheros local

### Aplicaciones de mensajería electrónica

La norma X.400 del CCITT de 1984 define los sistemas MHS (*Message Handling Systems*). En 1988 se produce su armonización con la propuesta de la ISO llamada MOTIS (*Message Oriented Interchange System*).

En el entorno de la arquitectura TCP/IP de internet el nombre que más suena es SMTP (*Simple Mail Transfer Protocol*).

Los sistemas de mensajería, conocidos popularmente como de correo electrónico (*E-mail*), tuvieron una gran aceptación desde un principio, puesto que no requerían la presencia simultánea del emisor del mensaje y de su receptor (un sondeo mostró que sólo el 28% de las llamadas telefónicas realizadas en el entorno profesional tenían éxito debido a que se producían errores en el número de teléfono, la línea se encontraba ocupada, el destinatario no estaba disponible, o estaba ausente, o en una reunión, o bien, no conocía la respuesta a la consulta realizada en aquel momento...).

La estructura de los mensajes manejados por un sistema de correo electrónico es la siguiente:

- Sobre (envelope)*: contiene las direcciones del destinatario (nombre, localización) y del remitente, las opciones especiales tales como la urgencia del mensaje, si está cifrado y/o firmado, si se requiere confirmación de entrega (correo certificado), etc.
- Cabecera del mensaje (header)*: contiene información del remitente (nombre, empresa), la fecha de emisión del mensaje, el tema o resumen del mensaje (*subject*), las referencias a otros mensajes y la estructura del mensaje (de cuantas partes consta, de que tipo de contenido se trata, etc.).
- Cuerpo del mensaje (body)*: el/los contenido/s del mensaje en sí; los contenidos pueden ser textos, ficheros binarios, gráficos, imágenes en movimiento, sonidos, etc.

La figura 2.31 muestra el modelo arquitectónico básico de un sistema de correo electrónico.

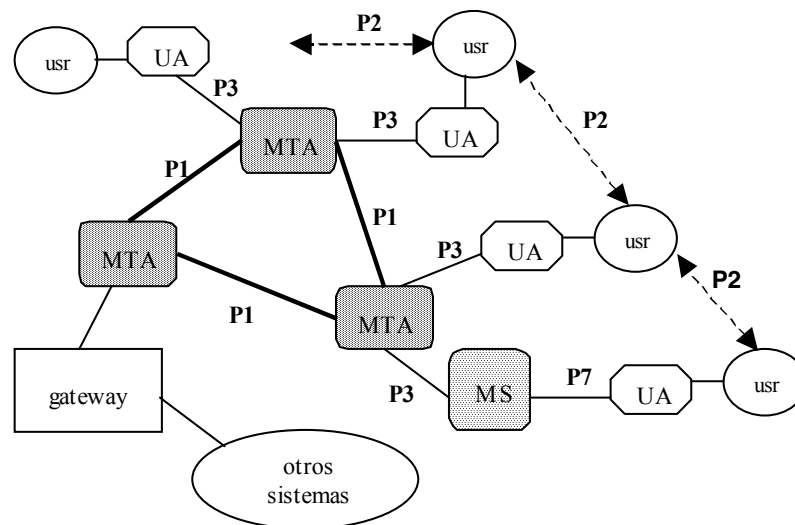


Fig. 2.31 Arquitectura de un sistema de correo electrónico: entidades funcionales y protocolos

Las funciones de las diversas entidades funcionales son:

- a) UA (*User Agent*): Encargadas del envío y recepción de los mensajes, de su presentación al usuario, así como de permitir la edición de nuevos mensajes, su almacenaje, su respuesta, etc. Interpretan y tratan la cabecera de los mensajes. Son equivalentes a las bandejas de correo entrante y saliente y al escritorio del hombre de oficina.
- b) MTA (*Message Transfer Agent*): Encargadas del reenvío o progresión de los mensajes hacia su destino; son, por así decirlo, los nodos de conmutación de mensajes a nivel de dicha aplicación. Interpretan el *sobre* de los mensajes al igual que las *oficinas de correos* del sistema postal tradicional.
- c) MS (*Message Store*): Almacenes de mensajes recibidos; utilizados cuando las UA de destino no están permanentemente activas. Son equivalentes a los *buzones* del sistema postal tradicional.
- d) *Gateways*: Convertidores de formatos y protocolos para unir servicios de mensajería distintos.

Los protocolos establecidos entre las diferentes entidades funcionales de un sistema de mensajería electrónica son:

- a) P1: protocolo entre entidades de transferencia de mensajes residentes en distintos sistemas; es un protocolo interno al sistema de mensajería pero que todas las entidades MTA deben seguir. En la arquitectura TCP/IP dicho protocolo puede ser SMTP (*Simple Mail Transfer Protocol*).
- b) P2: protocolo que se establece en UA y que define la estructura e interpretación de los mensajes intercambiados entre usuarios. Un ejemplo podría ser MIME (*Multi-purpose Internet Mail Exchange*).
- c) P3: protocolo que se establece entre los UA y los MTA. En la arquitectura TCP/IP, el protocolo con dicha función es también SMTP (comentado anteriormente).
- d) P7: protocolo que se establece entre los UA y las entidades MS. Ejemplos de dicho protocolo, asociados a la arquitectura TCP/IP, serían POP3 (*Post Office Protocol, versión 3*) e IMAP (*Internet Message Access Protocol*).

### ***Aplicaciones de acceso remoto de terminal***

Una de las formas más habituales de interacción entre el usuario humano y el sistema informático es a través del concepto de *terminal*. Independientemente de otras formas más modernas y refinadas de interacción (escritorios, ventanas, ratón), el terminal, formado por una unidad de entrada de datos (normalmente un teclado) y una unidad de salida de datos (normalmente una pantalla de visualización de caracteres alfanuméricos), sigue siendo una forma básica de acceso a los sistemas. Muchos procesos de aplicación hacen uso del concepto de terminal para la interacción con el usuario humano; ahora bien, a lo largo de la historia se han definido muchos tipos de terminales (básico o teletipo, ANSI, VT-52, VT-100, IBM 3270, etc.).

La aproximación de OSI a dicho tema se basa en el concepto de terminal virtual (en adelante VT: *Virtual Terminal*). El VT es una representación abstracta de una unidad de entrada/salida, de sus

capacidades de representación y de las operaciones en relación a ésta. Así, el VT en OSI es una definición parametrizable de un terminal que define la estructura de una pantalla y teclado virtuales, sus características y las operaciones que pueden realizarse sobre ésta. El VT puede ser alfanumérico, con un determinado número de filas y columnas de caracteres y con sus atributos (texto normal, parpadeante, con sobre-intensidad, con colores, etc.), o gráfico, con determinado número de puntos (pixels), etc. El teclado del VT puede estar constituido por teclas alfanuméricas y también especiales (por ejemplo, teclas de ESCape, de función, de movimiento de cursor, etc.).

Las operaciones sobre el VT pueden ser de escritura de caracteres en determinada posición (x, y), de borrado de caracteres, de borrado de línea, de cambios de atributos (color, aspecto), de borrado de pantalla, de trazado de líneas o dibujo de bloques, en terminales gráficos, etc. Las aplicaciones OSI (editores de textos, bases de datos, etc.) interactúan con un terminal virtual, es decir, sólo deben escribirse una vez en relación a ese terminal virtual, independientemente del terminal real que se use; en el otro extremo existe un terminal real y las correspondientes funciones de “mapeo” asociadas, como se ilustra en la figura 2.32.

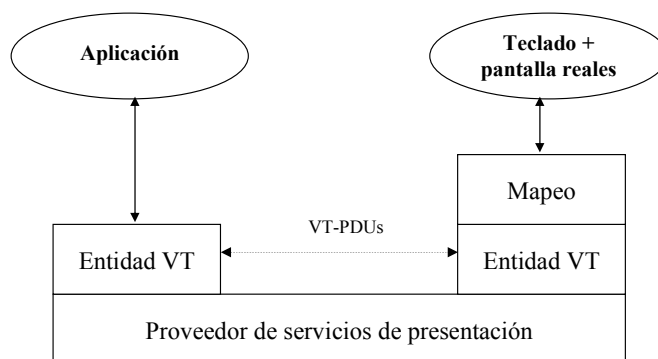


Fig. 2.32 Esquema básico de funcionamiento de la aplicación terminal (virtual) remoto

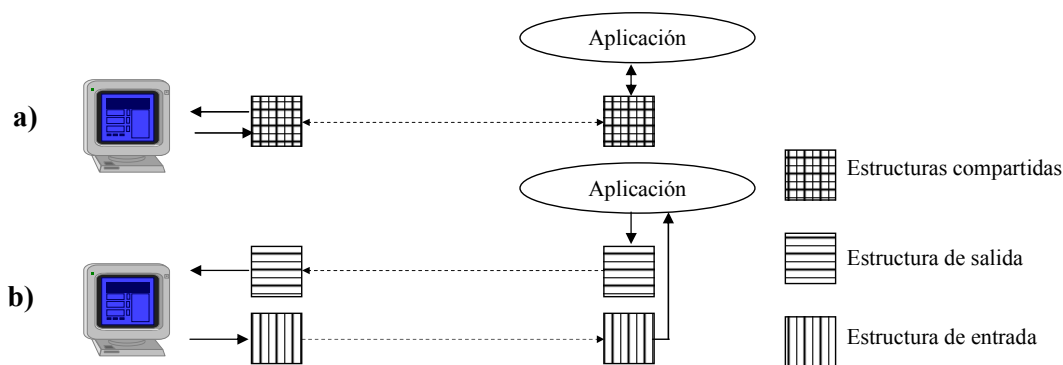


Fig. 2.33 Modos de funcionamiento de un terminal: a) síncrono (una estructura de datos representa al VT en cada extremo y el intercambio de VT-PDU mantiene su coherencia; b) asíncrono (la entrada y salida de terminal están desvinculadas)

El concepto de VT permite modelar la mayoría de modos de funcionamiento de los terminales habituales: modo *scroll* o *teletipo* (donde la pantalla es similar al papel de una impresora convencional), modo *página* o *formulario* (donde se consideran pantallas enteras con campos de información y campos “rellenables”; idea propuesta en las décadas de los 1970 y 1980 del siglo pasado y puesta de nuevo en auge por algunas aplicaciones tipo *web*). Asimismo, cabe considerar los modos asíncrono y síncrono, como se representa en la figura 2.33.

El terminal tradicional en sistemas tipo UNIX opera en el modo asíncrono, es decir, la entrada y salida están desligadas y sujetas al proceso de aplicación.

En la arquitectura TCP/IP, los procesos *telnet* cliente y *telnet* servidor y el protocolo TELNET establecido entre ellos representan la alternativa histórica, y algo arcaica ya, al acceso de terminal remoto. *Telnet* también define un terminal virtual (NVT: *Network Virtual Terminal*) sobre el que se realizan las operaciones de entrada/salida de muchas aplicaciones. Dicho terminal es de tipo alfanumérico únicamente y posee un modo “transparente” que permite a muchas aplicaciones desarrolladas para terminales reales típicos (por ejemplo, VT-100 de Tektronic) funcionar adecuadamente. Ligado también a la arquitectura TCP/IP y a los sistemas UNIX, *X-Windows* y el protocolo *X-protocol*, representan una alternativa más moderna y versátil a la interacción remota entre usuarios y procesos de aplicación.

### ***Aplicaciones de directorio***

El objetivo de las aplicaciones de servicios de directorio es el de mantener y facilitar información sobre objetos o entidades (personas, servidores, etc.). Su filosofía es parecida, por ejemplo, a la de las guías telefónicas, tanto en su variante de *páginas blancas* como de *páginas amarillas*. El estándar X.500 define estos servicios de forma general y completa. Asociado a la arquitectura TCP/IP, el protocolo LDAP (*Lightweight Directory Access Protocol*), define algo parecido. También asociado a esta arquitectura, el DNS (*Domain Names Service*) define y suministra información de entidades que ofrecen algún servicio (básicamente sitios web, aunque también servidores de ficheros, bases de datos con acceso en modo terminal, etc.).

### ***Aplicaciones para ejecución remota de tareas***

La propuesta OSI al proceso distribuido, es decir, al aprovechamiento de las capacidades de proceso o cálculo entre sistemas finales conectados en red, fue JTM (*Job Transfer & Manipulation*). JTM está orientado a la sumisión (orden de ejecución) de tareas en sistemas remotos y al control de los documentos asociados. Sus principales funciones son:

- a) Sumisión de tareas en sistemas remotos
- b) Control de las dependencias entre partes de tareas complejas
- c) Disposición de los resultados (retornar al usuario invocador, nueva entrada para otro proceso en otro sistema, es decir, encadenamiento de trabajos, etc.)

- d) Interrogación sobre el progreso de actividades JTM
- e) Modificación de trabajos, de las dependencias entre ellos o controles de seguridad

En la arquitectura TCP/IP, un ejemplo de una funcionalidad similar se tiene en RPC (*Remote Procedure Call*). RPC permite la ejecución de subrutinas en sistemas remotos y el paso de parámetros de entrada y de salida. Asociado a RPC se encuentra XDR (*eXternal Data Representation*), que permite cierta transparencia entre sistemas en cuanto a los parámetros de entrada/salida asociados a la ejecución de procedimientos remotos; XDR, por tanto, tendría funciones identificables con la capa de presentación OSI.

### 3. Introducción al análisis mediante teoría de colas

#### 3.1 Introducción

Los sistemas de transmisión pueden ser muy frecuentemente modelados según un esquema como el de la siguiente figura:



Fig. 3.1 Modelo simple de un sistema de transmisión de datos

Este esquema aparece de forma natural al estudiar las redes de transmisión. Muestra una fuente de datos, una cola de espera o almacenamiento temporal a la espera de que las unidades que en ella se acumulen sean atendidas por un servidor.

De este modelo, deben destacarse dos aspectos fundamentales: La disciplina con que se generan los mensajes ( $\lambda$ ) y la disciplina con que se sirven ( $\mu$ ). El término disciplina hace referencia a la estadística de las unidades de información.

En el caso de la disciplina de generación (también podemos referirnos a ella como disciplina de llegada de unidades), se trata de la estadística (momentos) de los tiempos de llegada de las unidades. Es muy importante notar que  $\lambda$  y  $\mu$  corresponden a la media de estas estadísticas, pero no aportan más información sobre la forma en que se genera la información (ráfagas, uniforme, etc.).

Una línea de transmisión puede ilustrar un ejemplo: la cola modela el retardo de transmisión (con posibles variaciones, asociadas al tamaño de la cola), y las tasas  $\lambda$  y  $\mu$  corresponden a la velocidad de entrada y salida de la información de dicha línea, que podría tener pérdidas de información.

En un sistema como el de este ejemplo, debe considerarse que todas las unidades de datos tienen las mismas características y requieren el mismo esfuerzo desde el punto de vista de su generación y servicio. En ciertos casos, ello puede implicar que sean de tamaño fijo.

Supóngase una tasa de llegadas de información al sistema de  $\lambda = 5$  paquetes/seg. (es decir, en media se recibe un paquete cada  $1/\lambda = 0,2$  segundos. Pasemos a estudiar las situaciones que se pueden distinguir en función del valor de la tasa de servicio  $\mu$ , por medio de simples razonamientos intuitivos:

Para  $\mu < \lambda$ , el sistema sirve las unidades en cola a un ritmo inferior al que llegan a ella.

Por lo tanto, no es capaz de servir las unidades que se reciben a razón de 5 por segundo en media, por lo cual el tamaño de la cola en cada instante dependerá de la estadística de las llegadas y en régimen permanente tenderá hacia infinito.

Para  $\mu > \lambda$ , el sistema es capaz de servir más de 5 paquetes por segundo, por lo cual la cola tendrá un tamaño finito. Ahora bien, el tamaño en general no será nulo, puesto que aunque las llegadas tengan una media  $\lambda$ , podrían producirse en ráfagas.

El último caso a considerar es la situación límite  $\mu = \lambda$ . En este caso, el sistema se encuentra al límite de estabilidad.

En resumen,

$$\mu \begin{cases} < \lambda \Rightarrow \text{Cola} \rightarrow \infty \\ = \lambda \Rightarrow \text{Límite estabilidad} \\ > \lambda \Rightarrow \text{Estable} \end{cases} \quad (3.1)$$

De este razonamiento aparentemente se deriva la necesidad de evitar sistemas cuyas colas de espera queden muy ocupadas, para lo cual emplear valores de  $\mu$  suficientemente mayores a  $\lambda$ . Aunque no es incorrecto, no siempre es oportuno considerar este criterio.

Una condición muy empleada en el diseño de sistemas, en general razonable, es la previsión de una cierta congestión de modo que se puedan rentabilizar económicamente los recursos invertidos.

Es evidente que el diseño de sistemas con servidores exageradamente dimensionados favorece la calidad de servicio entregada al cliente, aunque perjudica la amortización de equipos.

Un dimensionado adecuado debe efectuarse ajustando el diseño a niveles de congestión moderados donde se garanticen unas cotas mínimas de calidad (retardo y pérdidas de paquetes).

De estos conceptos, se define el parámetro *utilización* o *intensidad de tráfico en el enlace* como la relación entre la tasa de llegadas y la de servicio. Esto es,

$$\rho = \lambda / \mu \quad (3.2)$$

De donde, rehaciendo la relación 3.1, se puede escribir que:

$$\rho \begin{cases} > 1 \Rightarrow \text{Cola} \rightarrow \infty (\text{Inestable}) \\ = 1 \Rightarrow \text{Límite estabilidad} \\ < 1 \Rightarrow \text{Estable} \end{cases} \quad (3.3)$$

Obsérvese que en cualquier sistema que se desee analizar deben considerarse los siguientes aspectos:

- a) Cómo es la estadística de las llegadas de unidades al sistema.
- b) Cómo es la estadística del servicio de unidades de la cola.
- c) Cuántos servidores trabajan en paralelo (es decir, cuantas unidades pertenecientes a la misma cola de espera pueden servirse simultáneamente)
- d) Cuántos clientes generan unidades hacia la cola.
- e) Con qué disciplina operan las colas, desde el punto de vista de almacenar unidades y entregarlas a los servidores para que sean atendidas. Algunas alternativas comunes son las denominadas FIFO (*first in first out* - primera en entrar, primera en salir -) y LIFO (*last in first out* - última en entrar, primera en salir -).

## 3.2 Procesos de Poisson

En el apartado anterior se ha mencionado la importancia de la forma (estadística) en que las unidades se reciben en el sistema y en la que son atendidas. Es lo que se denomina disciplina de llegadas y de servicio.

En esta sección vamos a comentar una de las más empleadas por su simplicidad, propiedades y características generales: los procesos de Poisson. Sin embargo, estas prometedoras características le restringen a marcos de análisis más bien simples y no válidos en general para casos en los cuales participen fuentes de datos más o menos complejos, como puedan ser los multimedia (audio, video, etc.), aunque no es menos cierto que se ajusta bien a fuentes de datos en general.

### 3.2.1 Definición de proceso de Poisson

Considérense las siguientes hipótesis:

- a) Un proceso en el cual cada llegada sea independiente de cuando se produjo la anterior. Denominémosle *sin memoria*
- b) Población infinita. Es decir, que el número de fuentes sea tan grande que se pueda considerar que la tasa media de llegada de unidades no depende de la ventana temporal y es por tanto una constante, cuyo valor es  $\lambda$
- c) Que la probabilidad de que se produzca una llegada sea proporcional al tiempo  $\Delta t$ , es decir, que sea  $\lambda \cdot \Delta t + O(\Delta t)$ , donde  $O(\Delta t)$  es una  $O$  de Landau y hace referencia a los términos de orden superior a  $\Delta t$  (tienden hacia 0 más rápido que  $\Delta t$ , conforme  $\Delta t$  tiende a 0)

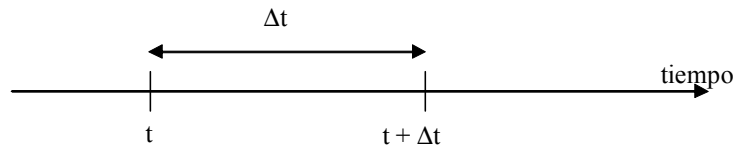


Fig. 3.2 Llegadas consecutivas

Bajo estas hipótesis, se demostrará más adelante que la probabilidad de que se produzcan  $n$  llegadas de unidades en un tiempo  $T$  (o  $\Delta t$ ) es:

$$P_n(T) = \frac{(\lambda T)^n}{n!} e^{-\lambda T} \quad (3.4)$$

Se puede demostrar fácilmente que está normalizado. Esto es,

$$\sum_{n=0}^{\infty} P_n(t) = 1 \quad (3.5)$$

De esta expresión se puede obtener la probabilidad de que no se produzca ninguna llegada en un tiempo  $t$  mediante  $n=0$ ,  $T=t$ , es decir,

$$P_0(t) = \frac{(\lambda t)^0}{0!} e^{-\lambda t} = e^{-\lambda t} \quad (3.6)$$

La probabilidad de tener alguna llegada en tiempo  $t$  se puede obtener de dos formas:

- a) Bien como la suma de tener 1 llegada, 2 llegadas, 3 llegadas, etc. hasta infinito
- b) Bien, de forma más simple (recordando que las probabilidades están normalizadas como indica la expresión 3.5) como 1 menos la probabilidad de no tener ninguna

Por ambas vías, se obtiene que:

$$P_{n \neq 0}(t) = 1 - e^{-\lambda t} \quad (3.7)$$

### 3.2.2 Propiedades

El número medio de unidades en el sistema en un intervalo de tiempo  $t$  se puede evaluar según la expresión

$$E[n] = \sum_{n=0}^{\infty} n \cdot P_n(t) \quad (3.8)$$

Por lo que,

$$E[n] = \sum_{n=0}^{\infty} n \cdot \frac{(\lambda t)^n}{n!} e^{-\lambda t} = e^{-\lambda t} \sum_{n=1}^{\infty} \frac{(\lambda t)^n}{(n-1)!} = e^{-\lambda t} \cdot \lambda t \cdot e^{\lambda t} = \lambda t \quad (3.9)$$

Para realizar el cálculo del sumatorio, resulta útil recordar que

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!} \quad (3.10)$$

A la vista de que  $E[n] = \lambda \cdot t$  se puede deducir - tal como puede esperarse - que  $\lambda$  es la velocidad de las llegadas por unidad de tiempo, ya que  $\lambda = E[n] / t$ .

De un modo parecido puede evaluarse la varianza de las llegadas de un proceso de Poisson:

$$\sigma^2 = E[n^2] - [E[n]]^2 \quad (3.11)$$

Este cálculo puede efectuarse de fácilmente teniendo en cuenta que

$$E[n^2] - [E[n]]^2 = E[n(n-1)] + E[n] - [E[n]]^2 \quad (3.12)$$

Finalmente, se obtiene que

$$\sigma^2 = \lambda \cdot t \quad (3.13)$$

En resumen, los procesos de Poisson cumplen que:

$$E[n] = \lambda t \quad \text{y} \quad \sigma^2 = \lambda \cdot t \quad (3.14)$$

Según esta propiedad, se definen los siguientes tipos de tráfico, en función de la relación entre la varianza y la media:

$$\text{Si } \frac{\sigma^2}{E[n]} \begin{cases} > 1 \Rightarrow \text{Tráfico de pico} \\ = 1 \Rightarrow \text{Tráfico de Poisson} \\ < 1 \Rightarrow \text{Tráfico suavizado} \end{cases} \quad (3.15)$$

### 3.2.3 Distribución de las llegadas en un proceso de Poisson

Hasta el momento nos hemos ocupado de estudiar cuántas llegadas se producen en un determinado intervalo de tiempo en una estadística de un proceso de Poisson. En esta sección se estudiará cuánto tiempo transcurre entre llegadas consecutivas mediante la evaluación analítica y obtendremos la función densidad de las llegadas para un proceso de Poisson.

Considérese para ello un intervalo de tiempo con un origen de tiempo arbitrario, al final del cual se produce la llegada de la siguiente unidad.

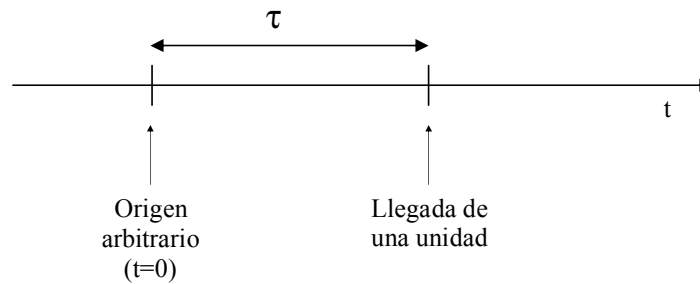


Fig. 3.3 Llegada de una unidad en  $t=\tau$

En la situación de la figura 3.3, se tiene que no se recibe ninguna unidad en el intervalo de tiempo comprendido en  $(0,\tau)$ .

Por lo tanto, la probabilidad de no tener ninguna llegada en el intervalo  $(0,t)$  es exactamente la de que  $\tau$  sea mayor a  $t$ . Es decir,

$$P(\tau > t) = P_0(t) = e^{-\lambda t} \quad (3.16)$$

Y por tanto,  $P(\tau \leq t) = 1 - e^{-\lambda t}$ . Nótese que  $P(\tau \leq t)$  denota la función distribución  $F_\tau(t)$ , por lo que por simple derivación puede obtenerse la función densidad:

$$f_\tau(t) = \frac{dF_\tau(t)}{dt} = \lambda \cdot e^{-\lambda t} \quad (3.17)$$

En resumen, en un proceso de Poisson, las llegadas siguen una función densidad *exponencial*.

A partir de esta función densidad, puede calcularse el tiempo medio entre llegadas:

$$E[t] = \int_0^{\infty} t \cdot f_\tau(t) \cdot dt = \int_0^{\infty} t \cdot \lambda \cdot e^{-\lambda t} \cdot dt = \frac{1}{\lambda} \quad (3.18)$$

En consecuencia, el tiempo medio entre llegadas es de  $1/\lambda$ , lo cual es un resultado completamente esperado a la vista de las hipótesis de partida.

Otros momentos estadísticos son:

$$E[t^2] = \frac{2}{\lambda^2} \quad (3.19)$$

$$\sigma^2 = \frac{1}{\lambda^2} \quad (3.20)$$

Este último resulta especialmente interesante, puesto que resulta que, en un proceso de Poisson, la media entre dos llegadas consecutivas coincide con la desviación. En otras palabras, se recibe una unidad en media cada  $1/\lambda$ , y puede estar desviada en media esa misma cantidad.

### 3.2.4 Propiedad de superposición

Supóngase dos procesos de Poisson independientes. Se puede enunciar la propiedad de la superposición del siguiente modo:

La suma de procesos de Poisson es otro proceso de Poisson cuya tasa es la suma de las tasas tributarias.

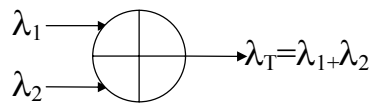


Fig. 3.4 Suma de procesos de Poisson

La demostración se basa en justificar el caso de dos fuentes de Poisson, ya que para el caso de tres, basta con asociar primeramente dos de ellos para demostrar que también se cumple. Del mismo modo se puede proceder para demostrar cualquier otro número de fuentes.

### 3.2.5 Propiedad de descomposición

Supóngase un caudal  $\lambda_T$  de Poisson. Al aplicar una función que descomponga este caudal en caudales más pequeños de forma aleatoria de acuerdo a una probabilidad  $P_i$ , los caudales resultantes tendrán una tasa  $\lambda_i = \lambda_T \cdot P_i$ , y serán también de Poisson.

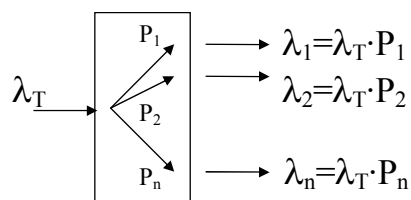


Fig. 3.5 Descomposición de procesos de Poisson

La demostración puede efectuarse para el caso de separar 2 de ellos, y el caso general será la descomposición de éstos hasta conseguir los  $n$  flujos deseados.

### 3.3 Cadenas de Markov

A continuación se estudiará una potente herramienta de análisis basada en la definición de ciertos estados en los que se puede encontrar un sistema con el fin de hallar las probabilidades de hallarlo en uno de estos estados. A partir de estas probabilidades, se pueden calcular parámetros de bondad que permitan caracterizarlo.

Considérese un sistema con diversos estados. Denominemos  $E_i$  al estado  $i$ . Por ejemplo, el estado  $E_i$  puede simbolizar el estado asociado a que  $i$  usuarios estén en un instante dado efectuando una llamada telefónica. Si hubiera  $n$  circuitos en total para cursar las llamadas, habría que definir desde un estado  $E_0$  hasta un estado  $E_n$ .

En esta situación, se denota la probabilidad de estar en el estado  $E_m$  en el instante  $t_i$  como  $P_r[E_m(t=t_i)]$ , que de forma abreviada puede escribirse como  $P_m(t_i)$ .

Se define el *Vector de Estado* del sistema como:

$$P(t_i) = [P_0(t_i), P_1(t_i), P_2(t_i), \dots] \quad (3.21)$$

Nótese que la notación del vector se puede distinguir porque en ella no aparece el subíndice.

Se trata de un *vector estocástico*, puesto que por definición se verifica que:

$$\begin{aligned} 0 \leq P_m(t_i) \leq 1 \\ \sum_{\forall m} P_m(t_i) = 1 \end{aligned} \quad (3.22)$$

Diremos que tenemos una *cadena* por disponer de un conjunto de estados que pueden representarse gráficamente enlazados entre ellos mediante flechas de transición entre unos estados y otros.

En general, la evolución de un sistema puede depender de todos los estados pasados, es decir, que  $\forall i$ , la  $P_r[E_m(t=t_{i+1})]$  puede depender de los estados anteriores  $E_n(t=t_i)$ ,  $E_p(t=t_{i-1})$ ,  $E_q(t=t_{i-2})$ , etc.

cumpléndose que  $t_{i+1} > t_i > t_{i-1} > t_{i-2} > \dots$ . En el caso de que únicamente dependa del estado presente,  $E_n(t=t_i)$ , se puede escribir que:

$$P_r[E_m(t=t_{i+1}) | E_n(t=t_i), E_p(t=t_{i-1}), E_q(t=t_{i-2}), \dots] = P_r[E_m(t=t_{i+1}) | E_n(t=t_i)] \quad (3.23)$$

En este caso, diremos que estamos ante un proceso sin memoria, un proceso de Markov.

Según las posibles transiciones entre los estados, queda definida la cadena de Markov, tal como muestra la figura 3.6. Nótese que no es necesario que las flechas alcancen todos los posibles estados. Cada flecha va asociada a una probabilidad de transición entre estados que debe ser definida.

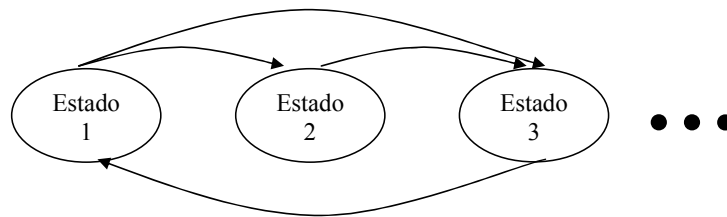


Fig. 3.6 Ejemplo simple de cadena de Markov

### 3.3.1 Sistemas de tiempo discreto y sistemas de tiempo continuo

La clasificación de los sistemas según si consideran transiciones de estado en instantes de tiempo determinados o indefinidos conduce a definir los sistemas como de *tiempo discreto* o *continuo* respectivamente.

Las señales de reloj de los sistemas digitales son un ejemplo de sistema discreto. La llegada del público a la entrada de un cine es un ejemplo de sistema continuo, puesto que se efectúan sin ningún instante de tiempo predeterminado.

De ahora en adelante nos vamos a interesar únicamente por los sistemas de tiempo continuo.

### 3.3.2 Cadenas de Markov de tiempo continuo

Dado que nos interesan los sistemas cuyas llegadas y salidas se produzcan en tiempo continuo, de ahora en adelante nos ocuparemos únicamente de cadenas de este tipo.

En este caso, la notación que anteriormente habíamos expresado en general como  $P_r[E_m(t=t_{i+1})]$  ahora se puede expresar simplemente como  $P_r[E_m(t)]$ , o más abreviadamente  $P_m(t)$ .

El vector de estado quedará escrito para tiempo continuo como

$$P(t) = [P_0(t), P_1(t), P_2(t), \dots] \quad (3.24)$$

cumpléndose que

$$\begin{aligned} 0 &\leq P_m(t) \leq 1 \\ \sum_{\forall m} P_m(t) &= 1 \end{aligned} \quad (3.25)$$

En este caso, que sea una cadena de Markov conduce a que la notación sea

$$P_r[E_n(t) | E_m(u), E_p(v), E_q(w), \dots] = P_r[E_n(t) | E_m(u)], \quad \text{donde } t > u > v > w > \dots \quad (3.26)$$

que abreviadamente expresaremos como

$$P_r[E_n(t) | E_m(u)] = P_{mn}(u, t) \quad (3.27)$$

Esta expresión se interpreta como la probabilidad de pasar del estado  $m$  al  $n$  desde el instante  $u$  al  $t$ .

Dado que la probabilidad de estar en el estado  $n$  en el instante  $t$  puede descomponerse según todos los caminos procedentes de cada uno de los estados hasta  $n$ , podemos escribir que:

$$P_n(t) = \sum_m P_m(u) \cdot P_{mn}(u, t) \quad (3.28)$$

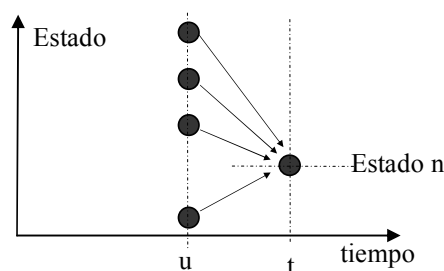


Fig. 3.7 Transiciones desde todos los estados hasta uno determinado

Disponiendo todos los valores  $P_{mn}(u, t)$  en forma matricial se define  $P(u, t)$ .

Con esta nueva notación, puede expresarse que

$$P(t) = P(u) \cdot P_{mn}(u, t) \quad (3.29)$$

Esta expresión sintetiza todos los aspectos relacionados con las cadenas de Markov de tiempo continuo estudiados en esta sección.

### 3.3.3 Ecuación de futuro

De acuerdo a la figura 3.8, que muestra la evolución del sistema desde un estado  $m$  en el instante  $u$  hasta un estado  $n$  en  $t+\Delta t$ , definimos la ecuación de futuro como:

$$P_{mn}(u, t + \Delta t) = \sum_p P_{mp}(u, t) \cdot P_{pn}(t, t + \Delta t) \quad (3.30)$$

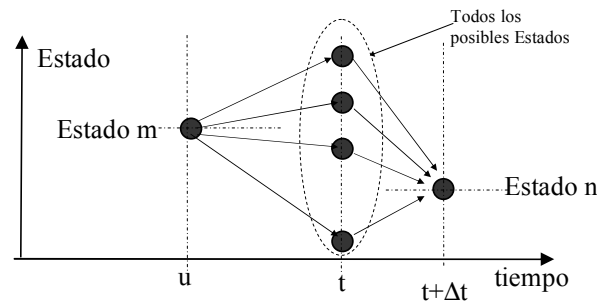


Fig. 3.8 Evolución temporal genérica de los estados

Extrayendo el término  $p=n$  del sumatorio y restando  $P_{mn}(u, t)$  a la expresión 3.30:

$$P_{mn}(u, t + \Delta t) - P_{mn}(u, t) = \left[ \sum_{p \neq n} P_{mp}(u, t) \cdot P_{pn}(t, t + \Delta t) \right] + P_{mn}(u, t) \cdot P_{nn}(t, t + \Delta t) - P_{mn}(u, t) \quad (3.31)$$

Dividiendo por  $\Delta t$  y tendiendo al límite  $\Delta t \rightarrow 0$ , podemos reconocer la expresión de la derivada, con lo cual resulta la siguiente ecuación de futuro:

$$\frac{\partial P_{mn}(u, t)}{\partial t} = \left[ \sum_{p \neq n} P_{mp}(u, t) \cdot q_{pn}(t) \right] + P_{mn}(u, t) \cdot q_{nn}(t) \quad (3.32)$$

En esta expresión se ha definido la velocidad de transición  $q_{pn}(t)$  y de permanencia  $q_{nn}(t)$  de la siguiente manera:

$$q_{pn}(t) = \lim_{\Delta t \rightarrow 0} \frac{P_{pn}(t, t + \Delta t) - \delta_{pn}}{\Delta t} \quad (3.33)$$

$$q_{nn}(t) = \lim_{\Delta t \rightarrow 0} \frac{P_{pn}(t, t + \Delta t) - 1}{\Delta t} \quad (3.34)$$

Se puede aplicar la siguiente condición inicial: en instante de tiempo 0 se parte del estado 0, por lo cual  $P_{0n}(0, t) = P_n(t)$ . Entonces la ecuación de futuro puede ser reescrita de la siguiente manera:

$$\frac{dP_n(t)}{dt} = \left[ \sum_{p \neq n} P_p(t) \cdot q_{pn}(t) \right] + P_n(t) \cdot q_{nn}(t) \quad (3.35)$$

Además, puede demostrarse fácilmente que

$$\sum_n q_{pn}(t) = 0 \quad (3.36)$$

Si se define la matriz  $Q(t)$  como la formada por  $[q_{pn}]$ , se puede escribir la siguiente ecuación que resume lo contado hasta el momento:

$$\frac{dP(t)}{dt} = P(t) \cdot Q(t) \quad (3.37)$$

### 3.3.4 Procesos de nacimiento y muerte

Hasta el momento, se han estudiado las cadenas de Markov en las cuales desde cualquier estado se puede ir a cualquier otro en el siguiente instante de tiempo.

Vamos a continuación a restringir esta situación y tomar como hipótesis que únicamente se puede pasar en el siguiente instante de tiempo a un estado inmediatamente vecino, esto es, desde el estado  $E_n$  se puede pasar al  $E_{n+1}$ , al  $E_{n-1}$  o permanecer en  $E_n$ . Este escenario define los procesos de nacimiento (cuando se pasa a un estado superior) y muerte (a un estado inferior).

En este caso, únicamente no serán nulas las probabilidades  $P_{m,m+1}$ ,  $P_{m,m}$  y  $P_{m,m-1}$ .

Como ejemplo, puede tomarse las llegadas a la cola de un cine. Cuando llega alguien, se pone a la cola. Incluso, aunque llegue un grupo, se puede considerar que cada persona se pone en cola con un diferencial de tiempo entre cada una de ellas.

En este escenario, la ecuación de futuro queda reducida a la siguiente expresión:

$$\frac{dP_n(t)}{dt} = P_{n-1}(t) \cdot q_{n-1,n}(t) + P_{n+1}(t) \cdot q_{n+1,n}(t) + P_n(t) \cdot q_{nn}(t) \quad (3.38)$$

Se define  $q_{n-1,n}(t) = \lambda_{n-1}(t)$  y se le conoce como *velocidad de nacimiento*.

Se define  $q_{n+1,n}(t) = \mu_{n+1}(t)$  y se le conoce como *velocidad de muerte*.

En este caso, considerando 3.36, se cumplirá que  $q_{nn}(t) = - (q_{n,n+1}(t) + q_{n,n-1}(t))$ . Usando la nueva notación, resulta que  $q_{nn}(t) = - (\lambda_n(t) + \mu_n(t))$ , donde hay que prestar un especial cuidado a los subíndices.

Con ello, la ecuación de futuro va tomando la forma buscada:

$$\begin{cases} \frac{dP_n(t)}{dt} = \mu_{n+1}(t) \cdot P_{n+1}(t) + \lambda_{n-1}(t) \cdot P_{n-1}(t) - (\lambda_n(t) + \mu_n(t)) \cdot P_n(t) & ; n > 0 \\ \frac{dP_0(t)}{dt} = \mu_1(t) \cdot P_1(t) - \lambda_0(t) \cdot P_0(t) & ; n = 0 \end{cases} \quad (3.39)$$

Constituye un sistema de ecuaciones diferenciales. Para su resolución, por motivos de linealidad del sistema, siempre deberá desprejiciarse una y tomar otra ecuación que sea linealmente independiente. Una que resulta adecuada para este fin es:

$$\sum_{n=0} P_n(t) = 1 \quad (3.40)$$

Y puede tomarse la siguiente representación gráfica:

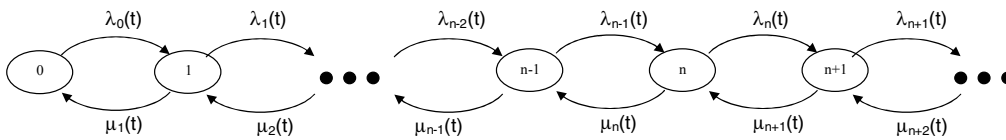


Fig. 3.9 Cadena de Markov de tiempo continuo para procesos de nacimiento y muerte

### 3.3.5 Ejemplo

Considérese las siguientes hipótesis:

- a) Proceso homogéneo. En este caso, las tasas de nacimiento y muerte no dependerán del estado:  $\lambda_n(t) = \lambda(t), \forall n$
- b) Población infinita. Un conjunto de fuentes tan “alta” permite garantizar que la velocidad del sistema será constante. En este caso,  $\lambda(t) = \lambda (= \text{constante}), \forall t$
- c) Asumamos también que sea un proceso de nacimiento puro, es decir, no hay muertes ( $\mu_n(t) = 0$ )
- d) Finalmente, consideremos que sea de Markov, y por tanto no tenga memoria

Una primera reflexión nos conduce a pensar que éstas son precisamente las características de los procesos de Poisson. Veamos, pues, con las herramientas de que disponemos, qué podemos obtener.

La cadena de Markov asociada a estas hipótesis es la mostrada en la figura 3.10.

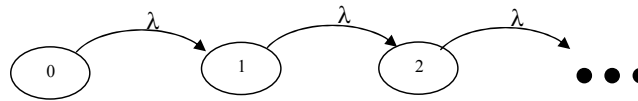


Fig. 3.10 Cadena de Markov para un proceso de Poisson

Substituyendo éstos en las hipótesis del ejemplo en la ecuación de futuro:

$$\begin{cases} \frac{dP_n(t)}{dt} = 0 + \lambda \cdot P_{n-1}(t) - (\lambda + 0) \cdot P_n(t) & ; n > 0 \\ \frac{dP_0(t)}{dt} = 0 - \lambda \cdot P_0(t) & ; n = 0 \end{cases} \quad (3.41)$$

es decir,

$$\begin{cases} \frac{dP_n(t)}{dt} = \lambda \cdot P_{n-1}(t) - \lambda \cdot P_n(t) & ; n > 0 \\ \frac{dP_0(t)}{dt} = -\lambda \cdot P_0(t) & ; n = 0 \end{cases} \quad (3.42)$$

con la condición inicial

$$P_k(0) = \begin{cases} 1 & ; k = 0 \\ 0 & ; k \neq 0 \end{cases} \quad (3.43)$$

donde la condición inicial tiene en cuenta que en el instante inicial el sistema se encuentra en el estado 0 (sin ninguna unidad).

Este sistema es fácilmente resoluble:

$$\underline{P_0(t)}: \quad \frac{dP_0(t)}{dt} = -\lambda \cdot P_0(t) \rightarrow P_0(t) = A \cdot e^{-\lambda \cdot t}$$

Mediante la condición inicial, se pueden obtener las constante de integración que irán apareciendo en el sistema. Para el caso de  $A$  de forma inmediata se obtendrá su valor 1.

Por tanto,  $P_0(t) = e^{-\lambda \cdot t}$

$$\underline{P}_1(t): \quad \frac{dP_1(t)}{dt} = \lambda \cdot P_0(t) - \lambda \cdot P_1(t) \rightarrow \frac{dP_1(t)}{dt} = \lambda \cdot e^{-\lambda t} - \lambda \cdot P_1(t) \rightarrow P_1(t) = \lambda \cdot t \cdot e^{-\lambda t}$$

Siguiendo la resolución del sistema para  $P_2(t)$ ,  $P_3(t)$ , etc. se puede obtener una expresión general:

$$P_n(t) = \frac{(\lambda \cdot t)^n}{n!} \cdot e^{-\lambda t} \quad (3.44)$$

Se reconoce la expresión que al principio de este capítulo se usó para definir los procesos de Poisson. Por lo tanto, este ejemplo permite justificarla.

### 3.3.6 Procesos de nacimiento y muerte en régimen permanente

Cuando se desea estudiar el régimen permanente de un sistema, éste ha alcanzado una situación en la cual ya no hay dependencias temporales. Las derivadas respecto al tiempo pasan a ser nulas.

En el caso de los procesos de nacimiento y muerte, el régimen permanente supone que  $dP_n(t)/dt = 0$  y el sistema de ecuaciones diferenciales 3.39 queda reducido a:

$$\begin{cases} 0 = \mu_{n+1} \cdot P_{n+1} + \lambda_{n-1} \cdot P_{n-1} - (\lambda_n + \mu_n) \cdot P_n & ; n > 0 \\ 0 = \mu_1 \cdot P_1 - \lambda_0 \cdot P_0 & ; n = 0 \end{cases} \quad (3.45)$$

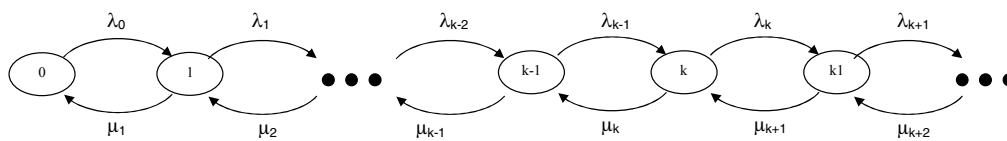


Fig. 3.11 Cadena de Markov para un proceso de nacimiento y muerte

### 3.3.7 Estudio mediante flujos

De estas ecuaciones y cadena representada se puede concluir que estos procesos pueden estudiarse en régimen permanente como flujos entrantes y salientes.

Los valores que pueden encontrarse son los siguientes, asociados a los estados iniciales, finales y el flujo entre ellos:

Estado inicial	Estado final	Flujo
0	1	$P_0 \cdot \lambda_0$
1	2	$P_1 \cdot \lambda_1$
k	k+1	$P_k \cdot \lambda_k$
1	0	$P_1 \cdot \mu_1$
2	1	$P_2 \cdot \mu_2$
k	k-1	$P_k \cdot \mu_k$

Dado que el sistema no acumula unidades, podemos afirmar que la suma de flujos entrantes debe coincidir con el de salientes,

$$\sum \text{Flujos Entrantes} = \sum \text{Flujos Salientes} \quad (3.46)$$

A partir de este simple razonamiento, se obtienen también el sistema de ecuaciones para los procesos de nacimiento y muerte, de forma simple y sin tener que recordarlas.

### 3.3.8 Cálculo de las probabilidades de estado de los procesos de nacimiento y muerte

Directamente podemos resolver las ecuaciones, empezando desde el estado 0 y llegando hasta el estado  $k$ , para obtener una expresión general:

$$\underline{n=0}: \quad \text{Directamente, } P_1 = \frac{\lambda_0}{\mu_1} P_0$$

$$\underline{n=1}: \quad \mu_2 \cdot P_2 = P_1 \cdot (\lambda_1 + \mu_1) - \lambda_0 \cdot P_0 = \frac{\lambda_0}{\mu_1} P_0 \cdot (\lambda_1 + \mu_1) - \lambda_0 \cdot P_0 = \frac{\lambda_0 \cdot \lambda_1}{\mu_1} P_0 \Rightarrow P_2 = \frac{\lambda_0 \cdot \lambda_1}{\mu_1 \cdot \mu_2} P_0$$

siguiendo la iteración y llegando a  $n=k$ , fácilmente se puede deducir que:

$$P_k = P_0 \frac{\lambda_0 \cdot \lambda_1 \cdot \dots \cdot \lambda_{k-1}}{\mu_1 \cdot \mu_2 \cdot \dots \cdot \mu_k} = P_0 \cdot \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \quad (3.47)$$

Para obtener el valor de  $P_0$ , se puede despejar de la expresión  $\sum_{k=0}^{\infty} P_k = 1$ :

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}} \quad (3.48)$$

La expresión obtenida para  $P_k$  y de  $P_0$  son especialmente útiles para el estudio de los sistemas que abordaremos en las siguientes secciones.

Nótese que  $P_0$  es la probabilidad de que no haya ninguna unidad en el sistema. Por consiguiente, el valor  $1 - P_0$  es la probabilidad de tener alguna unidad en el sistema.

### 3.4 Fórmula de Little

Sea un sistema modelable como el de la figura 3.12, donde  $\lambda$  es la tasa de llegadas al sistema,  $E[n]$  el número medio de unidades en el interior de este sistema y  $E[t]$  el tiempo medio de permanencia en él.

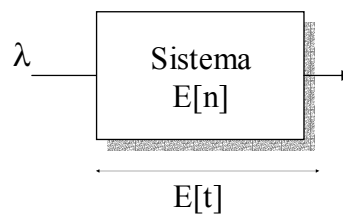


Fig. 3.12 Sistema genérico donde la fórmula de Little puede aplicarse

En estas condiciones, se puede demostrar que

$$E[n] = \lambda \cdot E[t] \quad (3.49)$$

Esta expresión es muy general y aplicable a todos los sistemas de colas que aparecerán en este libro, que son los más usuales.

### 3.5 La notación de Kendall y los modelos de colas.

La forma más usual de denotar de forma abreviada el tipo de cola empleado es mediante la notación de Kendall, que sigue el siguiente formato:

$$A / B / X / Y / Z$$

Estas variables caracterizan las colas de espera por los siguientes elementos:

- Tipo o distribución de las llegadas de las unidades. Se denota mediante  $A$ . Hace referencia al parámetro  $\lambda$ . Ejemplo son las llegadas a tasa constante o Poisson.

- Tipo o estadística del servicio ofrecido. Se denota mediante  $B$ . Hace referencia al parámetro  $\lambda$ .
- Número de servidores dispuestos en paralelo que atienden a la misma cola. Este valor corresponde a  $X$  de acuerdo a la notación anterior.
- Tamaño de la cola  $Y$  (o lo que es equivalente, número de unidades en el sistema. En este caso, el tamaño de la cola es el número de unidades en el sistema menos el número de servidores). En caso de omitirse este parámetro, se asume que el tamaño es infinito.
- Disciplina de la cola.  $Z$  puede tomar el valor de FIFO (*first in first out*) o LIFO (*last in first out*) según corresponda a un disciplina de cola o de pila, respectivamente. Por defecto, se asume una disciplina FIFO.

Por ejemplo, una cola  $M/M/1/\infty/FIFO$  modela un sistema con llegadas markovianas (es decir, estadística de Poisson), servicio también markoviano, con un único servidor, capacidad  $\infty$  (por lo cual es tamaño de la cola también ilimitado), con disciplina FIFO. De forma abreviada, resulta equivalente escribir  $M/M/1$ .

Veamos otro ejemplo. Sea una cola  $M/D/1$ . En este sistema, las llegadas siguen una estadística de Poisson, el servicio es *determinista*, con un único servidor. Al igual que en el ejemplo anterior, la cola es de longitud  $\infty$  y la disciplina FIFO. Los servicios deterministas se caracterizan por una entrega de unidades de datos de forma equidistante. Un tipo de redes que operan de esta forma son las basadas en tecnología ATM, tal como se estudiará más adelante.

A continuación se estudiarán los principales modelos para obtener sus propiedades y los valores estadísticos correspondientes a su modo de operación.

### 3.6 La cola M/M/1

#### 3.6.1 Modelo de cola

Considérese una tasa de llegadas  $\lambda$  y una tasa de servicio  $\mu$ . Cada estado representa el número de unidades en la cola de espera.

El primer paso en el estudio de un sistema es la representación de la cadena de Markov asociada. A partir de ella, y mediante un estudio a través de flujos o usando directamente las expresiones que ya hemos obtenido para las probabilidades de estado, se abordará el resto del estudio estadístico.

Dado que las llegadas se producen a tasa  $\lambda$ , independientemente del número de unidades en el sistema ésa será la tasa de nacimientos, y así se pone de manifiesto en la cadena representada en la figura 3.13.

En cuanto a las tasas de muerte, el servidor único de este sistema siempre sirve a tasa  $\mu$ , independientemente del estado de la cola. Así queda representado en la cadena.

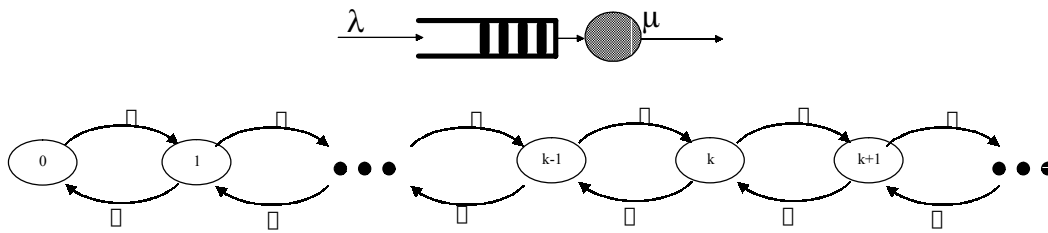


Fig. 3.13 Representación de una cola M/M/1

### 3.6.2 Probabilidades de estado

Partiendo de las expresiones 3.47 y 3.48 y considerando que  $\lambda_i = \lambda, \forall i$ , y  $\mu_i = \mu, \forall i$ , donde  $\lambda$  y  $\mu$  son valores constantes, se obtiene:

$$P_k = P_0 \cdot \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} = P_0 \left( \frac{\lambda}{\mu} \right)^k \tag{3.50}$$

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}} = \frac{1}{1 + \sum_{k=1}^{\infty} \left( \frac{\lambda}{\mu} \right)^k} = \frac{1}{1 + \frac{\lambda/\mu}{1 - \lambda/\mu}} = 1 - \lambda/\mu \tag{3.51}$$

Para el cálculo de  $P_0$  se ha tenido en cuenta que  $\lambda/\mu < 1$ , es decir, que la tasa de llegadas es menor a la de servicio. Desde un punto de vista matemático, ha permitido efectuar el sumatorio, donde se ha usado el conocido resultado del siguiente sumatorio:

Para  $x < 1$ : 
$$\sum_{k=1}^{\infty} x^k = \frac{x}{1-x}$$

Definiendo la *utilización* como  $\rho = \lambda / \mu$ , quedan las siguientes expresiones:

$$P_0 = 1 - \rho \tag{3.52}$$

$$P_k = (1 - \rho) \rho^k \tag{3.53}$$

Una vez se conocen las probabilidades de estado, podemos decir que el sistema ha quedado caracterizado. Vamos a evaluar a continuación los parámetros fundamentales.

### 3.6.3 Número medio de unidades en el sistema

Este valor puede obtenerse mediante simple cálculo estadístico, teniendo en cuenta que cada estado refleja el número de unidades en el sistema. Esto es, en el estado 0 hay cero unidades, en el estado 1 hay una unidad, etc.

$$E[n] = \sum_{n=0}^{\infty} n \cdot P_n = \sum_{n=0}^{\infty} n \cdot (1-\rho) \cdot \rho^n = \frac{\rho}{1-\rho} \quad (3.54)$$

### 3.6.4 Tiempo medio de permanencia de una unidad en el sistema

Para el cálculo del tiempo de permanencia, basta con usar la expresión de Little:

$$E[t] = \bar{T} = \frac{E[n]}{\lambda} = \frac{\frac{\rho}{1-\rho}}{\lambda} = \frac{1/\mu}{1-\rho} \quad (3.55)$$

Nótese que este resultado contiene tanto el tiempo de espera en cola como el tiempo de servicio:

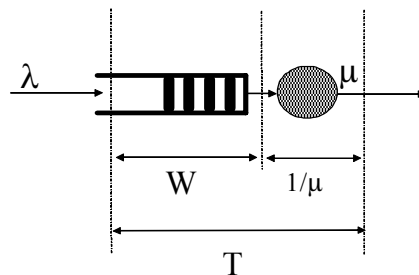


Fig. 3.14 Tiempo de espera en cola ( $W$ ) y tiempo medio de permanencia en el sistema ( $T$ )

$$\bar{T} = \bar{W} + \frac{1}{\mu} \quad (3.56)$$

Por tanto, el tiempo medio de espera en la cola será:

$$\bar{W} = \bar{T} - \frac{1}{\mu} = \frac{1/\mu}{1-\rho} - \frac{1}{\mu} = \frac{\rho}{(1-\rho)\mu} \quad (3.57)$$

Y aplicando la fórmula de Little, el número medio de unidades en la cola es de:

$$\overline{n}_q = \overline{W} \cdot \lambda = \frac{\rho^2}{1-\rho} \quad (3.58)$$

### 3.6.5 Ejemplo numérico

Supongamos las siguientes tasas de llegada y de servicio ante un sistema modelado mediante M/M/1 en el cual la unidad mínima de llegada son paquetes de información de tamaño fijo, pongamos de 100 bytes:  $\lambda = 8$  paquetes /seg (llegan a razón de 1 cada 1/8 de segundo),  $\mu = 10$  paquetes / seg (en media se sirve uno cada 0,1 segundos).

La utilización de este sistema es de  $\rho = 8/10 = 0,8$ .

La probabilidad de que esté desocupado es de  $P_0 = 1 - \lambda/\mu = 1 - \rho = 0,2$ .

Y la probabilidad de que esté ocupado es de  $1 - P_0 = 0,8$ .

La probabilidad de tener  $k$  unidades en el sistema es  $P_k = (1 - 0,8) \cdot 0,8^k = 0,2 \cdot 0,8^k$ .

El número medio de unidades en él es de  $E[n] = 0,8 / 0,2 = 4$  paquetes.

El tiempo medio de permanencia en el sistema de un paquete, desde que entra hasta que sale, es de  $E[t] = (1/10) / 0,2 = 0,5$  segundos.

El tiempo que este paquete habrá pasado en media en la cola, es de  $W = 0,8 / (1 - 0,8) / 10 = 0,4$  segundos y el tamaño medio de la cola es de  $n_q = 0,8^2 / (1 - 0,8) = 3,2$  paquetes, o lo que es lo mismo, el tamaño medio de la cola expresado en bytes es de  $3,2 \cdot 100 = 320$  bytes.

Lógicamente, estos valores contribuyen a poder dimensionar suficientemente el sistema, dadas las prestaciones obtenidas por él.

También son de interés el cálculo de momentos estadísticos de orden superior, con el fin de ajustar adecuadamente los valores.

Una vez dimensionado con una cola de tamaño finito (una memoria), conocer la probabilidad de que el sistema desborde consiste simplemente en sumar las probabilidades de los estados que quedan no cubiertos por el dimensionado de la cola (o bien, calcularlo como el complementario, es decir, 1 menos la suma de las probabilidades de todos los estados que sí se cubren con la memoria empleada).

### 3.7 La cola M/M/∞

#### 3.7.1 Modelo de cola

Al igual que para M/M/1, considérese una tasa de llegadas  $\lambda$  y una tasa de servicio  $\mu$  en un servidor. Cada estado representa el número de unidades en la cola de espera.

El primer paso en el estudio de un sistema es la representación de la cadena de Markov asociada. Podemos observar como primera intuición que este sistema dispone de infinitos servidores. En consecuencia, jamás habrá ninguna unidad en la cola de espera, puesto que siempre habrá un servidor dispuesto a atender a la unidad que llegue, de donde se deriva que en realidad este sistema no requiere ninguna cola y, en consecuencia, se trata simplemente de un conjunto de servidores atendiendo a todas las unidades recibidas.

Las llegadas se producen a tasa  $\lambda$ , independientemente del número de unidades en el sistema ésa será la tasa de nacimientos, y así se pone de manifiesto en la cadena representada en la figura 3.15.

En cuanto a las tasas de muerte, el hecho de que siempre haya un servidor esperando a una unidad que acabe de llegar hace que la tasa de servicio sea igual al número de unidades atendidas simultáneamente por el valor de un único servidor, que es  $\mu$ . Es decir, si hay 1 unidad en el sistema, se sirve a tasa  $\mu$ . Si hubiera 2, a tasa  $2\cdot\mu$ . Si hubiera  $k$ , a tasa  $k\cdot\mu$ . Se desprende que  $\mu_k = \mu\cdot k$ . Así queda representado en la cadena de la figura 3.15.

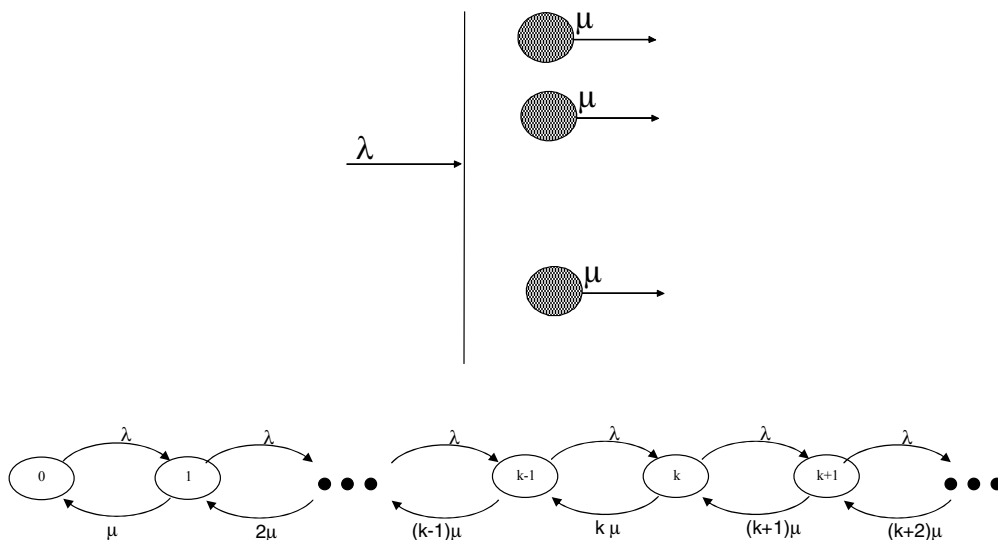


Fig. 3.15 Representación de una cola M/M/∞

### 3.7.2 Probabilidades de estado

Procediendo de forma análoga a como se ha efectuado para el sistema M/M/1, considerando que en este caso  $\lambda_i = \lambda$ ,  $\forall i$ , y  $\mu_i = i \cdot \mu$ ,  $\forall i$ , donde  $\lambda$  y  $\mu$  son valores constantes, se obtiene:

$$P_k = P_0 \cdot \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} = P_0 \cdot \left(\frac{\lambda}{\mu}\right)^k \prod_{i=0}^{k-1} \frac{1}{i+1} = P_0 \cdot \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!} \quad (3.59)$$

Llevando a cabo el cálculo para  $P_0$ , se obtiene que

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}} = e^{-\lambda/\mu} \quad (3.60)$$

por lo que

$$P_k = \frac{\left(\frac{\lambda}{\mu}\right)^k}{k!} e^{-\lambda/\mu} \quad (3.61)$$

En este caso, es perfectamente admisible (tanto matemática como semánticamente) que  $\lambda/\mu > 1$ , dado que siempre hay servidores disponibles.

Con estos valores, el lector puede efectuar, si lo desea, el cálculo de los valores medios tanto de unidades en el sistema como del tiempo de transmisión.

## 3.8 La cola M/M/m. Erlang C

### 3.8.1 Modelo de cola

Al igual que para los sistemas anteriores, considérese una tasa de llegadas  $\lambda$  y una tasa de servicio  $\mu$ , en un servidor. Cada estado representa el número de unidades en la cola de espera.

Como se desprende de la notación, el sistema que ahora se va a estudiar dispone de únicamente  $m$  servidores, con cola infinita.

En consecuencia, puede haber hasta  $m$  unidades atendidas simultáneamente y un número ilimitado esperando en cola. Nótese que la primera unidad que formará cola es la que ocupe el ordinal  $m+1$ . En este caso, el modelo de cola coincide con el de M/M/ $\infty$  hasta el estado  $m$ -ésimo, dado que hasta este momento, la situación es la misma.

Sin embargo, a partir de la llegada de la unidad  $m+1$ , ésta no será atendida y pasará a esperar su turno en cola. En consecuencia, cuando el estado del sistema sea superior a  $m$ , todos los  $m$  servidores estarán

activos, y por tanto, la tasa de servicio para los estados superiores a  $m$  será  $m \cdot \mu$ . Por otro lado, las llegadas se producen a tasa  $\lambda$ , independientemente del número de unidades en el sistema ésa será la tasa de nacimientos, y así se pone de manifiesto en la cadena representada en la figura 3.16.

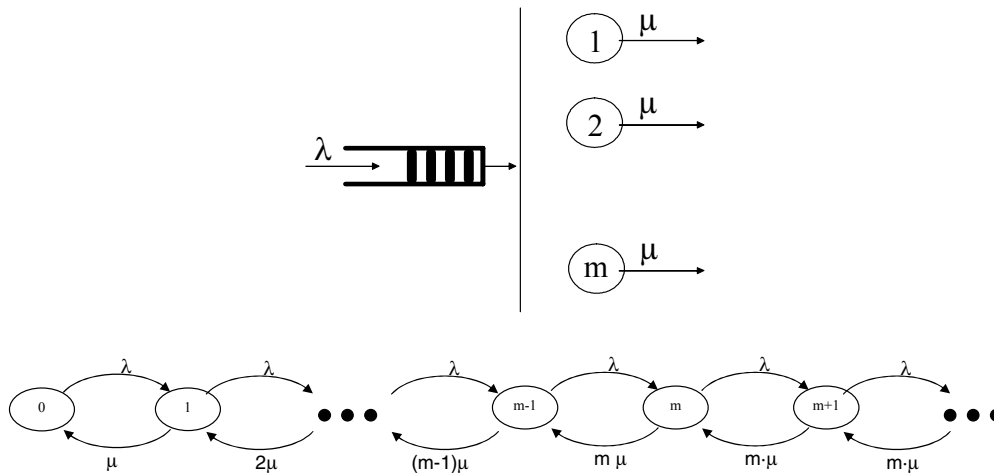


Fig. 3.16 Representación de una cola M/M/m

### 3.8.2 Probabilidades de estado

Procediendo de forma análoga a como se ha efectuado para el sistema M/M/1 y M/M/ $\infty$ , considerando en este caso que

$$\lambda_i = \lambda, \forall i,$$

$$\mu_i = i \cdot \mu \text{ para } i \in [1..m]$$

$$\mu_i = m \cdot \mu \text{ para } i \in [m+1..\infty)$$

donde  $\lambda$  y  $\mu$  son valores constantes. Para este caso con  $m$  servidores, se definen los siguientes parámetros:

$$\rho = \frac{\lambda}{m\mu} \text{ "utilización"} \quad (3.62)$$

que deberá ser inferior a la unidad para que el sistema sea estable.

$$a = \frac{\lambda}{\mu} \text{ "intensidad de tráfico"} \quad (3.63)$$

medido en Erlangs, inferior a  $m$  si el sistema es estable.

Se puede obtener que:

$$P_k = P_0 \cdot \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} = \begin{cases} \frac{m^m \rho^k}{m!} P_0 & ; k \geq m \\ \frac{(m \cdot \rho)^k}{k!} P_0 & ; k \leq m \end{cases} \quad (3.64)$$

Llevando a cabo el cálculo para  $P_0$ , el resultado es el siguiente

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}} = \frac{1}{\frac{a^m}{m!(1-\rho)} + \sum_{k=0}^{m-1} \frac{a^k}{k!}} \quad (3.65)$$

Más adelante se estudiarán algunas características importantes de este sistema de cola. Estudiemos primero otra disciplina de espera peculiar, denominada M/M/m/m, de la cual se derivarán consecuencias que también atañen a M/M/m.

Nótese que el parámetro  $a$  refleja el tráfico que desea cursarse sobre el sistema completo, mientras que  $\rho$  determina el rendimiento del sistema, en tanto por 1, considerando como potencial máximo de servicio  $m \cdot \mu$ , y  $\lambda$  como tasa de entrada.

### 3.9 La cola M/M/m/m. Erlang B

#### 3.9.1 Modelo de cola

Considerando como es habitual una tasa de llegadas  $\lambda$  y una tasa de servicio  $\mu$  en cada servidor, y cada estado representando el número de unidades en la cola de espera, este sistema presenta una capacidad  $m$  con  $m$  servidores.

Por lo tanto, no dispone de cola de espera para almacenar las unidades que a su llegada no puedan ser atendidas. En consecuencia, estas unidades se perderán inmediatamente. Un ejemplo típico de este tipo de sistemas se encuentra en las centralitas de telefonía, donde un usuario que no consiga un circuito de voz debe volver a llamar.

Debe notarse que el sistema sólo definirá un número finito de estados,  $m$  estados. No tiene sentido un estado  $m+1$  porque la  $m+1$  -ésima unidad se perdería al no haber espera posible.

La figura 3.17 muestra la figura asociada a esta disciplina, y la cadena de Markov correspondiente.

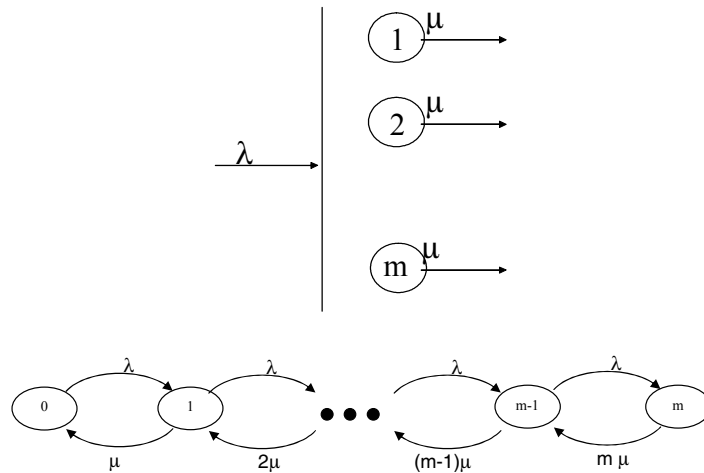


Fig. 3.17 Representación de una cola M/M/m/m

### 3.9.2 Probabilidades de estado

Se procede como siempre para el cálculo de las probabilidades de estado en régimen permanente, teniendo en cuenta que

$$\lambda_i = \lambda, i \in [0, m-1] (\lambda_i = 0 \text{ para el resto})$$

$$\mu_i = i \cdot \mu \text{ para } i \in [1..m] (\mu_i = 0 \text{ para el resto})$$

donde  $\lambda$  y  $\mu$  son valores constantes. Para este caso con  $m$  servidores, se usa la misma notación de  $\rho$  y  $a$  tal y como ha sido definida en M/M/m.

Se puede fácilmente obtener que:

$$P_k = P_0 \cdot \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} = \begin{cases} 0 & ; k > m \\ \frac{a^k}{k!} P_0 & ; k \leq m \end{cases} \tag{3.66}$$

donde

$$P_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}} = \frac{1}{\sum_{k=0}^m \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!}} \tag{3.67}$$

### 3.9.3 Situación de bloqueo. Función de Erlang B

Se define la situación de bloqueo como aquella en la cual todos los servidores están ocupados. En estas circunstancias, cuando llegue una nueva unidad, no podrá ser atendida y, al no disponer de cola de espera, se perderá.

De acuerdo con esta definición, se puede calcular la probabilidad de estar en esta situación de bloqueo por identificarla con estar con todos los servidores ocupados, puesto que en este caso la llegada de una nueva unidad provoca irremediamente su pérdida. En suma, la probabilidad de bloqueo coincide con la probabilidad del estado  $m$ , y es lo que se define como *función de Erlang B* usando la siguiente expresión de probabilidad:

$$P_m = \left(\frac{\lambda}{\mu}\right)^m \frac{1}{m!} \cdot \frac{1}{\sum_{k=0}^m \left(\frac{\lambda}{\mu}\right)^k \frac{1}{k!}} = \frac{a^m}{m!} \frac{1}{\sum_{k=0}^m \frac{a^k}{k!}} = Er_1(m, a) \quad (3.68)$$

Así pues, la función de Erlang B no es más que la probabilidad de bloqueo de un sistema M/M/m/m, es decir, de un sistema formado únicamente por servidores, sin cola de espera.

Este tipo de cola modela perfectamente el acceso a los circuitos de una central telefónica. Por ello, Erlang B es un modelo típicamente empleado para el dimensionado de sistemas basados en conmutación de circuitos.

Si  $a$  es el tráfico ofrecido (TO) al sistema, queda claro que  $a \cdot P_m (= a \cdot Er_1(m, a))$  debe ser el tráfico perdido (TP). Dado que el tráfico cursado (TC) es la diferencia entre el ofrecido y el perdido ( $TC = TO - TP$ ), se tiene que  $TC = a (1 - Er_1(m, a))$ , tal como muestra la figura 3.18.

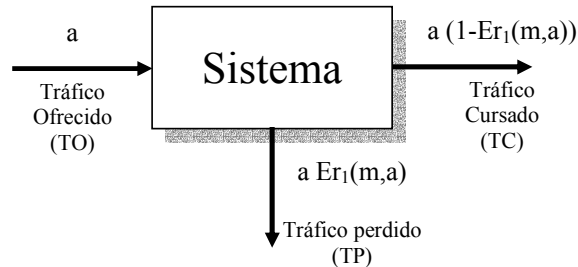


Fig. 3.18 Tráfico ofrecido, cursado y perdido en un sistema sujeto a pérdidas

También se puede obtener fácilmente que el número medio de unidades en el sistema es

$$E[n] = \sum_{k=0}^{\infty} k \cdot P_k = \sum_{k=0}^m k \cdot P_k = a \cdot [1 - Er_1(m, a)] \quad (3.69)$$

que es precisamente el tráfico cursado en el sistema.

### 3.9.4 Aplicación típica: Dimensionado de un sistema

Supóngase que se desea dimensionar un sistema que pueda cursar 1,2 Erlang, con una probabilidad de bloqueo del 5%, y desea conocerse el número de servidores necesario.

El problema puede plantearse en base a la fórmula de Erlang B, buscando cuál es el valor de  $m$  que verifica que  $Er_1(m, 1) \leq 0,02$ . El problema es obtener el valor de  $m$ , lo cual, a la vista de la expresión analítica de  $Er_1(m, a)$  se traduce en un problema de muy difícil resolución. La solución habitual pasa por emplear la técnica del tanteo, mediante la prueba de valores de  $m$  hasta aproximar lo mejor posible la probabilidad buscada.

La forma habitual de llevar a cabo esta labor es mediante las denominadas *tablas de Erlang*. Las tablas de Erlang son unos listados de valores de modo que se encuentran dispuestos en filas el TO ( $a$ ) y en columnas el número de circuitos ( $m$ ) de modo que la tabla presenta la probabilidad de bloqueo para cada  $a$  y  $m$ . De este modo, resulta mucho más cómodo localizar el número de circuitos  $m$  necesario para tener una probabilidad de bloqueo inferior al valor deseado, dado un determinado tráfico  $a$ .

La siguiente tabla muestra un fragmento de una tabla de Erlang B. La tabla puede diseñarse para ofrecer mayor precisión si la distancia entre valores consecutivos de  $a$  fuera menor.

Tabla 3.1 Vista parcial de una tabla de Erlang

$a \setminus m$	1	2	3	4
0,8	0,444444	0,150943	0,038694	0,007679
1,0	0,500000	0,200000	0,062500	0,015385
1,2	0,545455	0,246575	0,089776	0,026226
1,4	0,583333	0,289941	0,119180	0,040043

A la vista de esta tabla, se observa que escoger 3 servidores para un tráfico de 1,2 Erlang supone tener una probabilidad de bloqueo del 8,98%. Escoger 4 circuitos ofrece sólo un 2,62%. Dado que el requisito en el ejemplo era de un 5%, la elección de 4 servidores satisface las condiciones.

### 3.9.5 Relación de recurrencia

Se puede demostrar empleando 3.68 que se verifica la siguiente relación de recurrencia:

$$Er_1(m, a) = \frac{a \cdot Er_1(m-1, a)}{a \cdot Er_1(m-1, a) + m} \quad (3.70)$$

Esta expresión permite calcular  $Er_1(m, a)$  según  $Er_1(m-1, a)$ , es decir, en base a un servidor menos. Esta situación presenta un gran interés, ya que se puede conocer el valor de la función de Erlang B para

ningún servidor,  $Er_1(0,a)$ . Basta con atender al concepto semántico de  $Er_1(m,a)$ , que es la probabilidad de bloqueo. Resulta evidente que el sistema está completamente bloqueado si no se dispone de ningún servidor, por lo que

$$Er_1(0,a) = 1 \quad (3.71)$$

### 3.9.6 Ejemplo

Considérese un tráfico  $a = 0,5$  Erlangs, con  $m=3$  servidores. Calcular la probabilidad de bloqueo y el tráfico perdido. (Nótese que con 3 servidores pueden llegarse a cursar hasta 3 Erlangs).

Debemos calcular  $Er_1(3;0,5)$ .

$$Er_1(3;0,5) = \frac{0,5 \cdot Er_1(2;0,5)}{0,5 \cdot Er_1(2;0,5) + 3}$$

$$Er_1(2;0,5) = \frac{0,5 \cdot Er_1(1;0,5)}{0,5 \cdot Er_1(1;0,5) + 2}$$

$$Er_1(1;0,5) = \frac{0,5 \cdot Er_1(0;0,5)}{0,5 \cdot Er_1(0;0,5) + 1} = \frac{0,5 \cdot 1}{0,5 \cdot 1 + 1} = \frac{1}{3}$$

por tanto  $Er_1(2;0,5) = 0,0769$ , y finalmente  $Er_1(3;0,5) = 0,0126$ .

En suma, la probabilidad de bloqueo es de 1,26%.

El tráfico perdido es de  $a \cdot Er_1(3;0,5) = 0,5 \cdot 0,0126 = 6,32 \cdot 10^{-3}$  Erlangs.

### 3.9.7 Ejemplo de dimensionado

Una red de paquetes de tamaño constante genera un caudal saliente de tipo poissoniano de 16000 paquetes/segundo. Este caudal se manda a una batería de servidores que no disponen de ninguna memoria de espera para que manipulen su contenido antes de enviarlo hacia el exterior de la red. Se desea conocer cuántos servidores son necesarios con el fin de que ningún paquete se pierda por el hecho de encontrar todos los servidores ocupados, con una probabilidad inferior al 4%. Cada servidor es capaz de atender hasta 8000 paquetes/segundo.

En este ejemplo,  $\mu = 8000$  paquetes/segundo, y  $\lambda = 16000$  paquetes/segundo. Por tanto, la intensidad de tráfico será de  $a = \lambda/\mu = 2$  Erlangs.

Esto significa que *a priori* bastaría con 2 servidores para atender todo este tráfico, siempre y cuando se comportara de forma uniforme, sin ninguna aleatoriedad. El hecho de ser tráfico de Poisson provoca la posibilidad de encontrar los 2 servidores ocupados y la necesidad de una cola de espera.

Este escenario puede ser modelado mediante una disciplina M/M/m/m (si existiera cola de espera, el modelo adecuado sería M/M/m), es decir, un modelo de Erlang-B, y en consecuencia, el valor deseado es  $m$  de la expresión  $Er_1(m,2) \leq 0,1$ .

Este valor puede encontrarse de dos formas:

- a) Mediante las tablas de Erlang
- b) Por tanteo, probando diversos valores de  $m$

Aunque el procedimiento más conveniente es el primero, vamos a efectuarlo por tanteo, como simple ejercicio demostrativo:

$$Er_1(1,2) = \frac{2 \cdot Er_1(0,2)}{2 \cdot Er_1(0,2) + 1} = \frac{2}{3} > 0,04$$

$$Er_1(2,2) = \frac{2 \cdot Er_1(1,2)}{2 \cdot Er_1(1,2) + 2} = (\dots) = \frac{2}{5} > 0,04$$

$$Er_1(3,2) = \frac{2 \cdot Er_1(2,2)}{2 \cdot Er_1(2,2) + 3} = (\dots) = \frac{4}{19} > 0,04$$

$$Er_1(4,2) = \frac{2 \cdot Er_1(3,2)}{2 \cdot Er_1(3,2) + 4} = (\dots) = \frac{2}{21} > 0,04$$

$$Er_1(5,2) = \frac{2 \cdot Er_1(4,2)}{2 \cdot Er_1(4,2) + 5} = (\dots) = \frac{4}{109} = 3,67\% < 0,04$$

En resumen, se requieren 5 servidores para atender el requisito de este ejemplo.

### 3.10 Probabilidad de demora en Erlang C. Relación con Erlang B

#### 3.10.1 Expresión analítica

En el caso de disponer de una memoria de espera ante una batería de  $m$  servidores, el modelo adecuado es Erlang C. En este caso, el hecho de que todos los servidores estén ocupados no supone la pérdida de información, sino únicamente la espera en cola. En este caso, la probabilidad de encontrar todos los servidores ocupados supone una demora, y por esta razón esta circunstancia ha dado en denominarse *probabilidad de demora*.

Por lo tanto, la probabilidad de demora será la probabilidad de encontrar a todos los servidores ocupados. Esto es, la suma de probabilidades de todos estados en los que todos los servidores se encuentran ocupados:

$$P_d = \sum_{k=m}^{\infty} P_k = Er_2(m, a) \quad (3.72)$$

Esta expresión define la función de Erlang C,  $Er_2(m, a)$ .

Efectuando el sumatorio a partir de la expresión 3.64, se puede encontrar que

$$P_d = \frac{P_m}{1 - \rho} \quad (3.73)$$

a partir de la cual puede demostrarse que:

$$Er_2(m, a) = \frac{Er_1(m, a)}{1 - \frac{a}{m} + \frac{a}{m} \cdot Er_1(m, a)} \quad (3.74)$$

Esta expresión recursiva permite calcular la función de Erlang C según la función de Erlang B. A partir de ella resulta fácil construir unas tablas de Erlang C del mismo modo que se ha expuesto para la elaboración de las tablas de Erlang B.

### 3.10.2 Número de elementos en cola

$$\bar{n}_q = \sum_{k=m}^{\infty} (k - m) \cdot P_k = (\dots) = \frac{a}{m - a} Er_2(m, a) \quad (3.75)$$

Por tanto, empleando la fórmula de Little, el tiempo medio de espera en cola será:

$$\bar{W}_q = \frac{\bar{n}_q}{\lambda} = \frac{1}{\mu \cdot (m - a)} Er_2(m, a) \quad (3.76)$$

### 3.10.3 Ejemplo

Un sistema genera un tráfico poissoniano de 10 paquetes/seg, que se enviarán hacia la red a través de 2 módems que sirven con servicio de Poisson, cada uno con una capacidad de hasta a 6 paquetes/seg. Las dos arquitecturas de la figura son candidatas, donde cada servidor modela a uno de los módems. ¿Qué opción es mejor desde el punto de vista del tiempo de espera en cola?

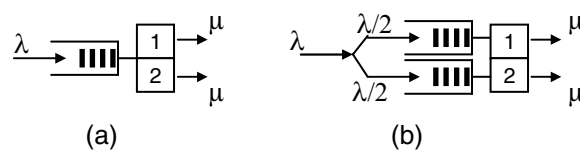


Fig. 3.19 Diversos esquemas de espera a dos servidores

Para resolver este ejercicio, basta con evaluar el tiempo de espera en cada una de las dos situaciones, teniendo presente que  $a = \lambda/\mu = 5/3$  Erlangs.

En el caso a), se observa un sistema M/M/2, al disponer de llegadas de Poisson, con 2 servidores también de Poisson. Ahora bien, con el fin de simplificar el desarrollo, este caso puede analizarse también mediante Erlang C.

En este caso, ya se ha estudiado que:

$$\bar{n}_q = \frac{a}{m-a} \text{Er}_2(m, a) = \frac{5/3}{2-5/3} \text{Er}_2(2, 5/3) = \frac{\text{Er}_1(2, 5/3)}{1 - \frac{5/3}{2} + \frac{5/3}{2} \cdot \text{Er}_1(2, 5/3)}$$

Dado que  $\text{Er}_1(2, 5/3) = \frac{5/3 \cdot \text{Er}_1(1, 5/3)}{5/3 \cdot \text{Er}_1(1, 5/3) + 2} = 0,3425$

resulta que  $\bar{n}_q = 3,79$  paquetes

Y por tanto,  $\bar{W}_q = \frac{\bar{n}_q}{\lambda} = \frac{3,79 \text{ paquetes}}{10 \text{ paq/seg}} = 0,379 \text{ seg.}$

El caso b) se trata de dos colas M/M/1, cada una con una tasa de llegadas  $\lambda' = \lambda/2 = 5$  paquetes/seg. Para cada cola,  $a' = \lambda'/\mu = 5/6$ .

Esta caso puede analizarse como la M/M/1 que es realmente, o incluso como un Erlang C con un único servidor. Usando la primera opción, tal como se ha estudiado, observando que coinciden  $\rho$  y  $a$ :

$$\bar{n}_q = \frac{\rho^2}{1-\rho} = \frac{(5/6)^2}{1-5/6} = 4,16 \text{ paquetes}$$

Por tanto,  $\bar{W}_q = \frac{\bar{n}_q}{\lambda'} = \frac{4,16 \text{ paquetes}}{5 \text{ paq/seg}} = 0,832 \text{ seg.}$

Nótese que este valor es sensiblemente superior al caso a) con una única cola.

De este ejemplo puede extraerse la siguiente e importante conclusión:

Ante tráficos de Poisson, con servidores markovianos, es preferible una única cola esperando al servicio de los servidores que una cola dedicada en cada servidor. Nótese que esta afirmación es válida para este tipo de estadística, pero en general, no puede garantizarse.

### 3.11 La cola M/G/1

#### 3.11.1 La fórmula de Pollaczek-Khinchine

Considérese a continuación un sistema con llegadas de Poisson de tasa  $\lambda$ , servicio sin ninguna disciplina específica (digamos, *general*), con un único servidor y cola de tamaño ilimitado.

En este caso, supóngase conocidos los momentos primero y segundo de la disciplina de servicio, es decir, se conoce  $E[s]$  (tiempo medio de servicio) y  $E[s^2]$  (segundo momento del servicio).

Se puede demostrar que se cumple

$$E[n] = \rho + \frac{\lambda^2 \cdot E[s^2]}{2(1-\rho)} \quad (3.77)$$

Esta importante expresión es conocida como fórmula de Pollaczek-Khinchine.

Definiendo  $B(t)$  como la función densidad del servicio (por ejemplo, en el caso de Poisson,  $B(t) = \mu \cdot e^{-\mu t}$ ), se puede enunciar:

$$E[s] = \int_0^{\infty} t \cdot B(t) \cdot dt \quad (3.78)$$

$$E[s^2] = \int_0^{\infty} t^2 \cdot B(t) \cdot dt \quad (3.79)$$

$$\rho = \lambda \cdot E[s] \quad (3.80)$$

Notar que la expresión 3.80 es completamente análoga a  $\rho = \lambda/\mu$ .

A partir de la fórmula de Little, 3.77 puede adoptar la siguiente forma:

$$E[t] = \frac{E[n]}{\lambda} = E[s] + \frac{\lambda \cdot E[s^2]}{2(1-\rho)} \quad (3.81)$$

de donde por simple inspección se observa que el tiempo de permanencia en cola debe ser el segundo sumando, puesto que el primero es directamente el tiempo de servicio:

$$\overline{W}_q = \frac{\lambda \cdot E[s^2]}{2(1-\rho)} \quad (3.82)$$

### 3.11.2 Ejemplo 1: Servicio exponencial

En este caso, M/G/1 es realmente una M/M/1, por lo cual  $B(t) = \mu \cdot e^{-\mu t}$

El cálculo de las integrales puede efectuarse para obtener los siguientes valores, que son fácilmente esperables:

$$E[s] = \int_0^{\infty} s \cdot \mu \cdot e^{-\mu s} \cdot ds = (\dots) = 1/\mu$$

$$E[s^2] = \int_0^{\infty} s^2 \cdot \mu \cdot e^{-\mu s} \cdot ds = (\dots) = 2/\mu^2$$

Susbtituyendo,

$$E[t] = E[s] + \frac{\lambda \cdot E[s^2]}{2(1-\rho)} = (\dots) = \frac{1/\mu}{1-\rho} \quad (3.83)$$

expresión que ya se obtuvo durante el análisis tradicional con la cadena de Markov asociada a M/M/1.

### 3.11.3 Ejemplo 2: Servicio determinista

En este caso, M/G/1 es realmente una M/D/1, por lo cual  $B(t) = \delta\left(t - \frac{1}{\mu}\right)$ , esto es, una función delta

que fija el tiempo de servicio cada  $1/\mu$ , exactamente. Un ejemplo de este tipo de comportamiento pueden ser las redes ATM, cuyas unidades de información, denominadas celdas, se transmiten por los enlaces a una tasa determinada constante, multiplexando el caudal de todos los usuarios.

El cálculo de las integrales es muy simple en este caso:

$$E[s] = \int_0^{\infty} s \cdot \delta\left(s - \frac{1}{\mu}\right) ds = 1/\mu$$

$$E[s^2] = \int_0^{\infty} s^2 \cdot \delta\left(s - \frac{1}{\mu}\right) ds = 1/\mu^2$$

Susbtituyendo,

$$E[t] = E[s] + \frac{\lambda \cdot E[s^2]}{2(1-\rho)} = (\dots) = \frac{1}{\mu} \frac{2-\rho}{2 \cdot (1-\rho)} \quad (3.84)$$

### 3.12 Diversos escenarios de estudio

Se proponen a continuación unas cuantas situaciones que tienen por objetivo permitir estudiar diversos escenarios desde el punto de vista de la teoría de colas.

La primera situación presentada debe llevar a la evaluación del tiempo medio de espera y el número de unidades en cola para acceder a un servicio atendido por un protocolo de *parada y espera*, con llegadas poissonianas de tasa  $\lambda$ , una probabilidad  $p$  de transmisión incorrecta de un paquete de datos y un tiempo  $t_r$  invertido en la transmisión y retorno de la confirmación (ACK o NACK) desde el destino. Este es un ejercicio de nivel medio, donde el esfuerzo debe dedicarse al modelado de la situación y poderla abordar mediante la fórmula de Pollaczek-Khinchine.

La segunda situación propuesta se basa en una red ATM, que en un simple modelo puede presentar nodos de entrada con servicio determinista de tasa 155 Mbit/s, y 3 fuentes de Poisson de 2, 8 y 34 Mbit/s hacia uno de los enlaces atendido por uno de estos nodos de entrada. ¿Cuál es el tiempo medio de espera en la cola de entrada en la red? ¿Cuál es el número de bits en el nodo de entrada a la red? ¿Cuál el tiempo de entrada en el enlace? Es un ejercicio simple, que permite deducir fácilmente un criterio para el tamaño ocupado en las colas de espera de los conmutadores.

El tercer caso aquí propuesto se centra también en una red de banda ancha basada en tecnología ATM que ofrece a 4 usuarios un enlace de 34 Mbit/s para que sea compartido. El acceso se efectúa de forma determinista, atravesando un búfer de espera (una memoria RAM) para que los bits entren a la red cuando ésta esté disponible para transmitirlos. Se puede considerar que los 4 usuarios presentan un perfil de generación de información poissoniano, de tasas 500 Kbit/s, 1,5 Mbit/s, 6 Mbit/s y 9 Mbit/s. En este contexto, es posible evaluar:

- a) La utilización de la red
- b) El tiempo medio de espera en la cola de acceso
- c) La ocupación media de la memoria RAM
- d) El tiempo total de acceso a la red

Se trata de una variación al caso anterior, donde se incluyen más elementos y por tanto conduce a una situación más real. Su dificultad es baja.

## 4. Redes de área local

Las tecnologías de red de área local vienen definidas por una normativa cuyos aspectos más destacados son los aspectos relacionados con el mecanismo de acceso. Este capítulo presenta los antecedentes de la tecnología Ethernet, que es sin duda la de mayor difusión actual, así como los principios de las modernas soluciones que permiten accesos a 1 Gbit/s.

El capítulo describe los protocolos más importantes de acceso aleatorio, empezando por Aloha y Aloha ranurado o S-Aloha. A continuación se exponen los principios de CSMA y CSMA/CD, base del estándar IEEE 802.3, o su variante Ethernet.

Precisamente por su importancia, Ethernet merece un estudio donde se muestran las restricciones para el incremento de caudal hasta 1000 Mbit/s. Finalmente se concluye con las tendencias de este estándar, especialmente como solución para acceso y transporte metropolitano.

### 4.1 Protocolos de acceso aleatorio

Los protocolos de acceso aleatorio constituyen una familia de mecanismos de acceso cuya característica notoria es la aleatoriedad en el instante de acceso. Este esquema de funcionamiento facilita una simplicidad de diseño y funcionamiento.

Sin embargo, este procedimiento dar lugar a estados en las cuales dos o más estaciones traten de efectuar un acceso simultáneamente. Esta situación se denomina colisión. La colisión debe ser resuelta, es decir, las estaciones deben acabar consiguiendo realizar su transmisión de forma satisfactoria. Las propuestas y soluciones existentes para la resolución de colisión pueden mejorar el rendimiento de este tipo de protocolos, aunque en general debe considerarse bajo (valores del orden del 40-50% del ancho de banda total disponible en el enlace).

El capítulo está organizado presentando las propuestas históricamente más antiguas, basadas en acceso Aloha, y sigue la evolución que ha existido hasta alcanzar las más modernas, implementadas en Gigabit Ethernet.

## 4.2 Aloha

El mecanismo de acceso Aloha tiene su origen en el año 1970 de la mano de Norman Abramson, en la universidad de Hawai. Presenta un nuevo y elegante mecanismo para la solución del problema de la asignación de canal en un medio compartido. Su aplicación principal se centra en los sistemas radio. Aunque se basa en algoritmos muy simples, su sencillez ha dado lugar a un uso muy extenso en protocolos actuales y ha generado una familia de protocolos derivados tales como CSMA, PRMA o DQRUMA, que mejoran su rendimiento.

### 4.2.1 Algoritmo de acceso

Aloha permite el acceso para transmisión en cualquier instante de tiempo, cuando la estación transmisora dispone de datos. Este mecanismo supone un riesgo de 2 o más transmisiones simultáneas o coincidencia de señales en el medio, ya sea aéreo o cable. Este estado se denomina *colisión* e implica la destrucción de las formas de onda y por tanto de la información que las señales transportan.

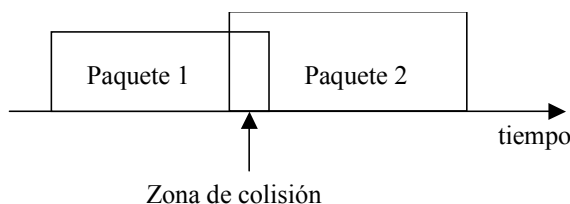


Fig. 4.1 Zona susceptible de colisión en un acceso según protocolo Aloha

Consecuentemente, es necesario un proceso de *retransmisión* en el cual se reinicia el intento. Para ello, se efectúa una espera aleatoria.

Para la detección de la colisión, Aloha usó inicialmente una estación central dedicada a este propósito. Otra alternativa usada posteriormente es el empleo de paquetes de reconocimiento envidados desde el receptor hasta el emisor con el fin de notificar el éxito de la transmisión, con tiempos máximos de espera o *time-out*.

### 4.2.2 Análisis

Asumiremos para el análisis la transmisión de datos mediante tramas de tamaño fijo, digamos de duración  $T$ . Esta hipótesis facilita el análisis y permite la obtención de un mejor *throughput* que con tramas de longitud variable. Asimismo, vamos a considerar que todos los usuarios del sistema transmiten con un tráfico agregado de Poisson de tasa  $\lambda$  paquetes/seg.

Se define el *factor de utilización* como  $\rho = \lambda \cdot T$ . Evidentemente, su valor debe ser inferior a la unidad. Esto es, el máximo servicio que puede ser ofrecido es de valor  $\mu = 1 / T$ , que es la situación en la que un paquete viaja justamente tras el anterior, aprovechando el medio al 100% de su tiempo, sin espacios desocupados.

Teniendo en cuenta que ante cualquier colisión, la estación va a efectuar un reintento y bajo la anterior hipótesis de que el tráfico es de Poisson, pueden aplicarse los principios de descomposición y superposición de tráficos poissonianos para configurar el siguiente modelo:

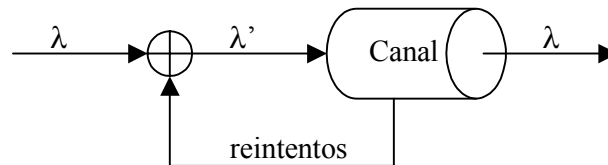


Fig. 4.2 Modelo de colisiones y reintentos en protocolo Aloha

Se puede observar que  $\lambda' > \lambda$ , puesto que incluye el tráfico entrante de los usuarios más todas las retransmisiones debidas a colisión. Por otro lado, por simple condición de estabilidad, el tráfico saliente ( $\lambda$ ) debe coincidir con el entrante.

En estas condiciones, la probabilidad de tener  $n$  llegadas de paquetes en el canal procedentes de los usuarios y de los reintentos en un tiempo  $t$ , es de

$$P_n(t) = \frac{(\lambda't)^n}{n!} e^{-\lambda't} \quad (4.1)$$

donde se emplea  $\lambda'$  puesto que incluye el caudal total a la entrada del canal.

Debe notarse de que existe una vía alternativa de análisis considerando un modelo de cola M/G/1 y el empleo de la fórmula de Pollaczek y Khinchine, mediante la caracterización de la estadística del servicio a través del primer y el segundo momento.

La probabilidad de no tener ninguna colisión, teniendo un paquete emitido en el instante  $t_0$ , implica que no haya ningún otro en el intervalo  $[t_0-T, t_0+T]$ , tal como muestra la figura 4.3.

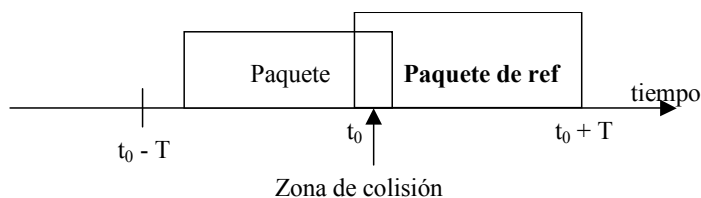


Fig. 4.3 Intervalos de tiempo en zona de colisión

En esta situación, la probabilidad de no existir colisión coincide con la probabilidad de que no haya ningún paquete en un intervalo de tiempo  $2T$ . Por lo tanto,

$$\text{Probabilidad de no colisión} = P_{n=0}(t = 2T) = \frac{(\lambda'2T)^0}{0!} e^{-\lambda'2T} = e^{-\lambda'2T} \quad (4.2)$$

Consecuentemente,

$$\text{Probabilidad de colisión} = 1 - e^{-\lambda'2T} \quad (4.3)$$

que también puede ser interpretada como la probabilidad de retransmisión.

Por lo tanto, la tasa media de retransmisión será la tasa de entrada al canal por la probabilidad de retransmisión, es decir,

$$\text{Tasa media de transmisión} = \lambda'(1 - e^{-\lambda'2T}) \quad (4.4)$$

Dado que, por conservación del tráfico, debe cumplirse que  $\lambda' = \lambda + \text{tasa de reintentos}$ , ello significa que

$$\lambda' = \lambda + \lambda'(1 - e^{-\lambda'2T}) \quad (4.5)$$

Denotando  $S = \lambda \cdot T (= \rho)$  y  $G = \lambda' \cdot T$  como el tráfico cursado de los usuarios y a la entrada del canal, respectivamente, la expresión 4.5 puede ser escrita como

$$S = G \cdot e^{-2G} \quad (4.6)$$

Con esta notación, la probabilidad de no colisión o de transmisión con éxito 4.2 puede ser escrita como:

$$\text{Probabilidad transmisión correcta} = e^{-2G} \quad (4.7)$$

La siguiente figura muestra cuál es el comportamiento del sistema ( $G$ ) frente a diversas entradas de tráfico ( $S$ ).

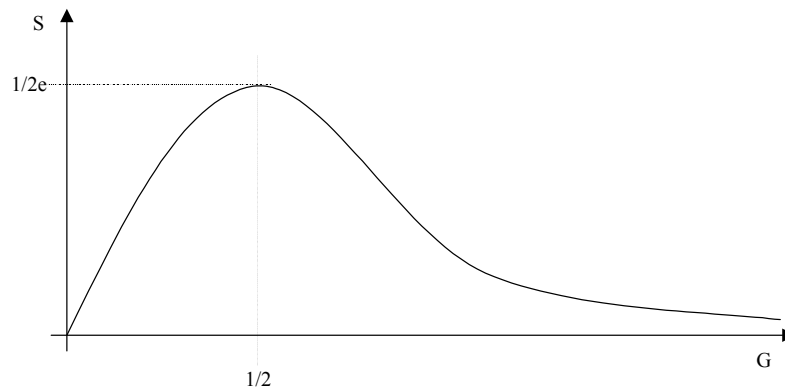


Fig. 4.4 Curva de tráfico cursado frente al ofrecido empleando protocolo Aloha

El valor máximo puede ser fácilmente deducido por simple derivación:  $dS/dG = 0$ .

Las conclusiones principales que pueden extraerse son:

- El valor del tráfico máximo entrante en ningún caso puede superar  $1/2e$  (aproximadamente el 18% de la capacidad teórica del medio). Este máximo se alcanza para  $G=1/2$ , lo cual significa que el uso del canal es únicamente del 50%, donde se incluye la información que acaba de llegar más los reintentos debidos a anteriores colisiones. En estas condiciones, un medio que operara con un tiempo de bit de 1 mseg, con codificación PAM-2, únicamente sería capaz de transportar  $1/2e \cdot 1/10^{-3}$  bit/s = 180 bit/s, de un total teórico máximo de  $10^3$  bit/s. En este mismo sistema, el canal estaría transmitiendo 500 bit/s, de los cuales únicamente 180 serían realmente efectivos, el resto debido a reintentos.
- Se puede constatar la existencia de dos zonas de trabajo separadas por el máximo, como aparece reflejado en la figura 4.5.

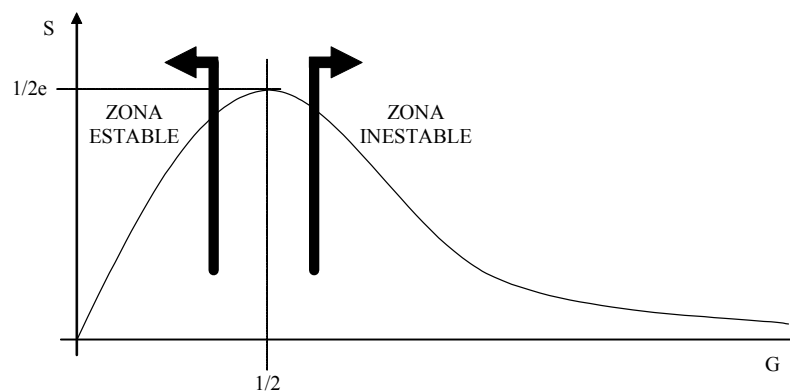


Fig. 4.5 Zonas de estabilidad e inestabilidad en Aloha

Como se observa, se ha definido una zona denominada estable y una inestable. La zona estable se caracteriza porque ante un incremento del caudal entrante ( $S$ ), el sistema responde con un incremento del caudal en el canal ( $G$ ). Sin embargo, en la zona inestable, un incremento en  $S$  impone un decremento en  $G$ , lo cual implica un comportamiento autodegenerativo que impide el funcionamiento del sistema. En consecuencia, es necesario que en ningún caso el sistema alcance la zona inestable, pues la única forma de recuperación sería la desactivación del mecanismo de acceso y su reinicialización.

### 4.3 S-Aloha

El mecanismo de acceso S-Aloha (*Slotted Aloha*, *Aloha Ranurado*) deriva del ya comentado Aloha. Su origen en 1972, de la mano de Roberts, permitió mejorar las prestaciones de Aloha manteniendo algoritmos muy simples y una elegante sencillez, con una pequeña e interesante modificación.

#### 4.3.1 Algoritmo de acceso

Conocido el comportamiento del protocolo Aloha, S-Aloha se puede definir como una versión ranurada en el tiempo. Es decir, el acceso de los paquetes al medio no puede efectuarse en cualquier instante, sino solamente cuando se inicia una de las ranuras de tiempo. El resto de condicionantes son los mismos que para el método de acceso Aloha.

El procedimiento de análisis será muy similar, aunque los resultados son sensiblemente diferentes, para mejorar el rendimiento de forma apreciable.

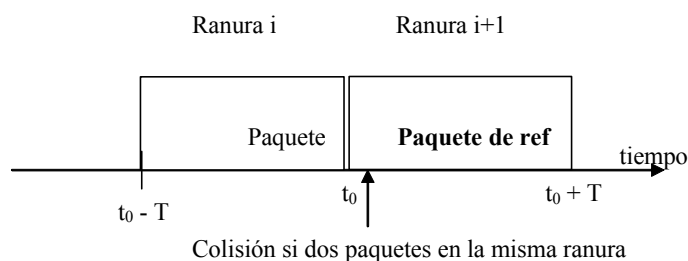


Fig. 4.6 Intervalos de tiempo ranurados en S-Aloha

#### 4.3.2 Análisis

Al igual que con Aloha, asumiremos tramas de tamaño fijo de duración  $T$ . Asimismo, vamos a considerar que todos los usuarios del sistema transmiten con un tráfico agregado de Poisson de tasa  $\lambda$  paquetes/seg.

Teniendo en cuenta que ante cualquier colisión, la estación efectuará un reintento, sigue siendo válida la figura , ya empleada para modelar el acceso en Aloha, con la salvedad de que los intentos de acceso solamente se efectúan a instantes discretos:

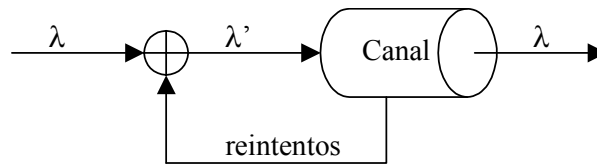


Fig. 4.7 Modelo de colisiones y reintentos en protocolo S-Aloha

La probabilidad de tener  $n$  llegadas de paquetes en el canal procedentes de los usuarios y de los reintentos en un tiempo  $t$ , cumple la expresión 4.1, con total analogía a Aloha.

La probabilidad de no tener ninguna colisión, teniendo un paquete emitido en el instante  $t_0$ , implica que no haya ningún otro en la misma ranura, tal como muestra la figura 4.6.

En esta situación, la probabilidad de no existir colisión coincide con la probabilidad de que no haya ningún paquete en un intervalo de tiempo  $T$ , no  $2T$  como el caso en Aloha, dado que ahora los paquetes deben esperar la llegada del inicio de las ranura para ser depositados en el medio. Por lo tanto,

$$\text{Probabilidad de no colisión} = P_{n=0}(t=T) = \frac{(\lambda'T)^0}{0!} e^{-\lambda'T} = e^{-\lambda'T} \quad (4.8)$$

Consecuentemente,

$$\text{Probabilidad de colisión} = 1 - e^{-\lambda'T} \quad (4.9)$$

La tasa media de retransmisión será en S-Aloha,

$$\text{Tasa media de transmisión} = \lambda' \cdot (1 - e^{-\lambda'T}) \quad (4.10)$$

Dado que debe cumplirse que  $\lambda' = \lambda +$  tasa de reintentos,

$$\lambda' = \lambda + \lambda' \cdot (1 - e^{-\lambda'T}) \quad (4.11)$$

Finalmente, la expresión 4.11 puede ser escrita como

$$S = G \cdot e^{-G} \quad (4.12)$$

Con esta notación, la probabilidad de no colisión o de transmisión con éxito 4.8 puede ser escrita como:

$$\text{Probabilidad transmisión correcta} = e^{-G} \quad (4.13)$$

La siguiente figura muestra cuál es el comportamiento del sistema ( $G$ ) frente a diversas entradas de tráfico ( $S$ ).

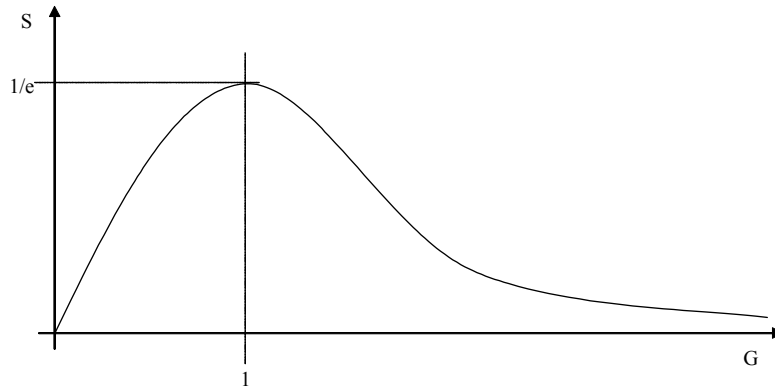


Fig. 4.8 Curva de tráfico cursado frente al ofrecido empleando protocolo S-Aloha

El valor máximo puede ser fácilmente deducido por simple derivación:  $dS/dG = 0$ .

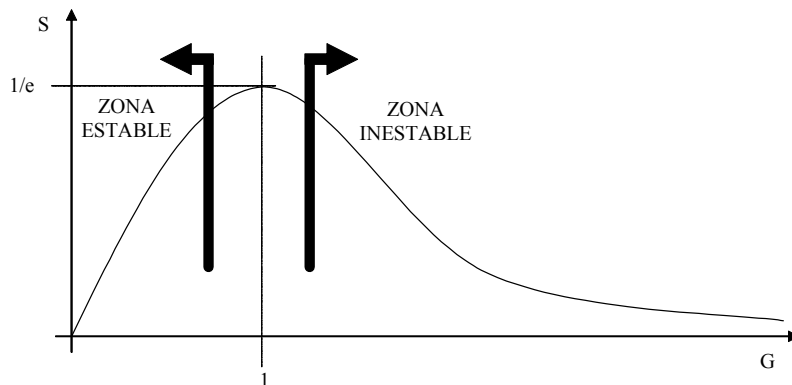


Fig. 4.9 Zonas de estabilidad e inestabilidad en S-Aloha

Las conclusiones principales que pueden extraerse son:

- a) El valor del tráfico máximo entrante en ningún caso puede superar  $1/2e$  (aproximadamente el 37% de la capacidad teórica del medio). Este máximo se alcanza para  $G=1$ , lo cual significa

que el uso del canal es únicamente del 100%, donde se incluye la información que acaba de llegar más los reintentos debidos a anteriores colisiones. En estas condiciones, un medio que operara con un tiempo de bit de 1 mseg, con codificación PAM-2, únicamente sería capaz de transportar  $1/e \cdot 1/10^{-3}$  bit/s = 370 bit/s, de un total teórico máximo de 1000 bit/s. En este mismo sistema, el canal estaría transmitiendo un total de 1000 bit/s, de los cuales únicamente 370 serían realmente efectivos, el resto debido a reintentos. Puede observarse que esta situación es óptima desde el punto de vista del canal, y que ofrece un rendimiento el doble mejor que Aloha.

- b) Se puede constatar la existencia de dos zonas de trabajo separadas por el máximo, como aparece reflejado en la figura 4.9.

Los comentarios relacionados con las zonas de estabilidad e inestabilidad son análogos a los ya efectuados para Aloha.

### 4.3.3 S-Aloha con población finita

Hasta ahora se han estudiado los casos con población infinita, de tal modo que se asegura que las llegadas sean uniformes.

Ahora, se considera la situación con  $N$  usuarios iguales e independientes, con un tráfico total de entrada entregado  $S$ , y un tráfico total en el canal  $G$  (incluyendo los reintentos debidas a las colisiones).

De este modo, el caudal generado en media por el usuario  $i$ -ésimo será:

$$S_i = \frac{S}{N} \quad (4.14)$$

Y la contribución al caudal en el canal debido al usuario  $i$ -ésimo:

$$G_i = \frac{G}{N} \quad (4.15)$$

Nótese que se verifica que  $G_i \leq S_i$ .

Dado que  $S_i$  es el valor del tráfico transmitido exitosamente, podemos afirmar que coincide con la probabilidad de transmisión, es decir:

$S_i = \text{Probabilidad} \{ [\text{Transmisión (exitosa o no) del usuario "i"}] \& [\text{Ningún otro usuario transmita}] \} =$

$$= G_i \cdot \prod_{\substack{j=1 \\ (j \neq i)}}^N (1 - G_j) \quad (4.16)$$

Teniendo en cuenta 4.14 y 4.15, se puede escribir la expresión 4.16 del siguiente modo:

$$\frac{S}{N} = \frac{G}{N} \cdot \prod_{\substack{j=1 \\ (j \neq i)}}^N \left(1 - \frac{G}{N}\right) = \frac{G}{N} \cdot \left(1 - \frac{G}{N}\right)^{N-1} \quad (4.17)$$

Simplificando, queda la relación final que caracteriza a S-Aloha para población finita:

$$S = G \cdot \left(1 - \frac{G}{N}\right)^{N-1} \quad (4.18)$$

De esta expresión se puede notar que:

1. Por simple derivación, se puede obtener la condición de máxima utilización:  $\sum_{i=1}^N G_i = 1$ .
2. En el caso de que el número de usuarios tienda a  $\infty$ , se recupera la expresión 4.12 de Aloha ranurado con población infinita. Basta con efectuar el cálculo de aplicar el límite de  $n \rightarrow \infty$ .

La tabla 4.1 muestra el rendimiento para diversos usuarios. Cuando el número crece, se tiende al comportamiento con población infinita, como era de esperar. Además, el rendimiento máximo decrece, puesto que el riesgo de colisión aumenta con el número de usuarios. Debe recordarse que una colisión provoca ocupar un tiempo adicional el medio hasta conseguir la transmisión satisfactoria.

Tabla 4.1 Rendimiento máximo para diverso número de usuarios

N	1	2	5	10	100	$\infty$
S <sub>max</sub>	1	0,5	0,410	0,387	0,370	1/e

#### 4.4 Redes CSMA

En Aloha la intención de transmisión se efectúa independientemente de si algún otro equipo está transmitiendo o no. Ello puede fácilmente ser causa de colisión, especialmente cuando el caudal total generado por todos los usuarios es elevado, y consecuentemente, factor fundamental en el bajo rendimiento de esta tecnología.

La aportación de CSMA (*Carrier Sense Multiple Access*) consiste en que antes de transmitir, el equipo debe *escuchar* el estado del medio. En el caso de que lo detecte desocupado, puede efectuar la transmisión. Ello supone una mejora frente a Aloha, puesto que gracias a la disminución del número de colisiones, cabe esperar un incremento de rendimiento.

Nótese, sin embargo, que ello no garantiza la ausencia de colisiones, puesto que dos equipos podrían actuar del mismo modo y, consecuentemente, producirse una colisión tras considerar que el medio está libre de transmisiones. Existe un margen temporal que es el que puede producir este problema. Este margen es el tiempo de propagación. Estudiemos esta situación.

La técnica CSMA, es decir, “escuchar” el medio antes de transmitir, es efectiva (evita la mayoría de las colisiones) cuando el retardo de propagación en el medio es pequeño comparado con el tiempo de transmisión de las tramas, es decir, cuando el parámetro  $a$  (presentado en el capítulo 2) es  $\ll 1$ . Puede decirse que, a mayor retardo de propagación, mayor probabilidad de colisiones (llamadas *colisiones primarias*), debido a la mayor incertidumbre con que las estaciones conocen el estado del medio de transmisión.

El procedimiento de acceso a una red CSMA impone ciertas restricciones. Revisemos a continuación el algoritmo desde un punto de vista de la transmisión de las tramas de información:

Si una estación desea transmitir, escucha el medio. Si el medio está ocupado, espera un cierto tiempo y reintenta. Si no, puede acceder al medio para transmitir los datos.

Este procedimiento, como se ha visto anteriormente, tiene el inconveniente de un riesgo de 2 o más accesos simultáneos, que es lo que se denomina colisión. El proceso se ilustra en la figura 4.10.

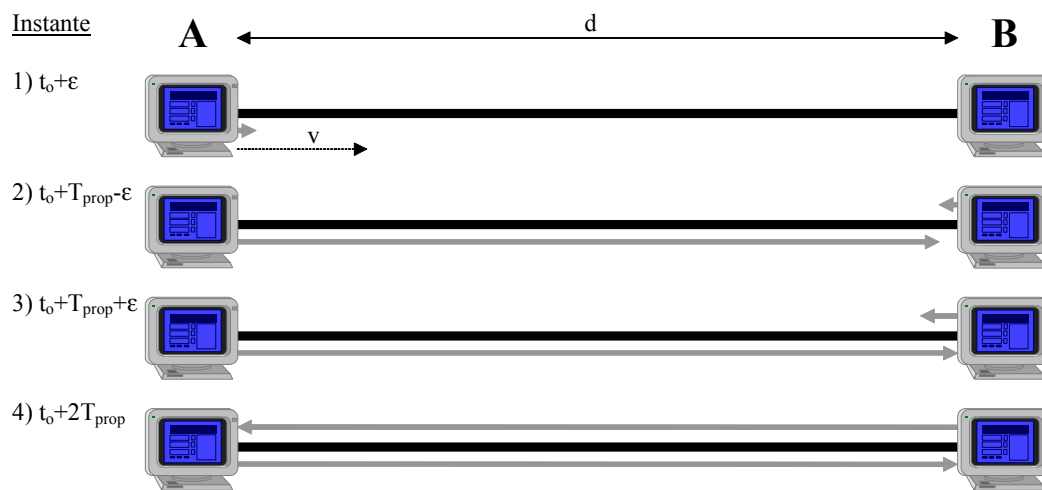


Fig. 4.10 Evolución de una transmisión crítica, entre equipos situados en los extremos de la red

Esta figura muestra diversos pasos en la transmisión entre los equipos  $A$  y  $B$  situados en los extremos de la red, separados por una distancia  $d$ .

Asumiendo que el medio estaba anteriormente totalmente desocupado, el primer paso muestra como la estación  $A$  inicia una transmisión hacia  $B$  en el instante  $t_o + \varepsilon$ , donde  $\varepsilon$  es un valor tan pequeño como se desee. En este instante ha generado una trama, puesto que ha observado el medio desocupado y por tanto no había impedimento. La onda electromagnética asociada a esta trama de información se propagará por el medio a velocidad  $v$  (en el caso de un medio de cobre tal como el par trenzado, esta velocidad es aproximadamente  $2/3$  de la velocidad de la luz en el vacío).

Al cabo de un tiempo  $T_{prop} = d/v$  esta trama alcanzará la estación  $B$ . Situemos la situación justo un instante antes, digamos  $t_o + T_{prop} - \varepsilon$ , que es el paso 2 de la figura. Supóngase en estas circunstancias

que la estación  $B$  desea iniciar una transmisión hacia la estación  $A$ . En  $t_o + T_{prop} - \epsilon$  el equipo  $B$  escucha el medio. Al no detectar ninguna señal, decide de forma lícita, de acuerdo al procedimiento de transmisión CSMA, iniciar su transmisión.

Se alcanza entonces al tercer paso, en el instante  $t_o + T_{prop} + \epsilon$ .  $B$  ha iniciado la transmisión y la señal que genera colisionará (se sumará eléctricamente) con la procedente del equipo  $A$ , que habrá alcanzado el extremo remoto de la red. En este momento,  $B$  apreciará la existencia de la colisión, pero  $A$  todavía no.  $A$  no lo detectará hasta que la onda llegue hasta él.

Al cabo de un tiempo  $T_{prop}$  desde el paso 3, se llega finalmente al último, en  $t_o + 2T_{prop}$ , momento en el cual  $A$  reconocerá la existencia de la colisión al recibir la señal procedente de  $B$ . Debe notarse, que para que  $A$  sepa que ha sido uno de los participantes en la colisión, su transmisión debe continuar cuando reciba la señal de  $B$ , puesto que de no ser así, podría creer erróneamente que la colisión no es debida a él.

De esta última frase se deduce una expresión de un interés fundamental: La duración de una transmisión debe ser al menos la necesaria para poder detectar una colisión, esto es,  $2 \cdot T_{prop}$ :

$$T_{transmisión} > 2 T_{prop} \quad (4.19)$$

Considerando los siguientes parámetros:

- $d$ : distancia extremo-extremo (m)
- $v$ : velocidad de propagación en el medio (m/s)
- $R$ : Velocidad del medio (bit/s)
- $L$ : Longitud de la trama

Se puede expresar el tiempo de transmisión como

$$T_{transmisión} = L/R \quad (4.20)$$

y el tiempo de propagación como

$$T_{prop} = d/R \quad (4.21)$$

Con estas expresiones, puede reescribirse 4.19 como:

$$\frac{L}{R} > 2 \frac{d}{v} \quad (4.22)$$

o lo que resulta equivalente,

$$L > 2R \frac{d}{v} \quad (4.23)$$

De esta expresión se deriva una importante interpretación: el tamaño mínimo de la trama está acotado al valor  $2 \cdot R \cdot d/v$ .

En otras palabras, el tamaño mínimo está limitado por parámetros físicos del sistema (diámetro de la red y velocidad de propagación de la onda) y por la tasa de transmisión de la red (capacidad). Es decir,

$$L_{\min} = 2R \frac{d}{v} \quad (4.24)$$

En suma, el objetivo de CSMA es conseguir evitar colisiones. Los mecanismos de escucha antes de transmisión (*Listen Befote Talk*, LBT) se remontan a 1975, de la mano de Kleinrock y Tobagi.

De entre las diversas variantes existentes de CSMA, deben citarse CSMA/CA (*CSMA with Collission Avoidance*), empleada en las redes Localtalk de Appletalk, o en las recientes Wireless LAN 802.11, y también destaca CSMA/CD (*CSMA with Collission Detection*). Este último será abordado en las siguientes secciones.

#### 4.5 Redes CSMA/CD

Se ha visto que, aun cuando CSMA reduce la posibilidad de colisiones, éstas todavía se producen. Dos o más estaciones transmitiendo tramas que colisionan originan un desperdicio de la capacidad (ancho de banda) disponible, pues la información en el medio físico no es válida durante todo el tiempo que dura la transmisión de dichas tramas.

El inconveniente en CSMA es que cuando se produce una colisión entre tramas, el medio queda ocupado durante todo el tiempo que dura la transmisión de la trama. Parece obvio que sería adecuado cortar lo antes posible la transmisión con el fin de dejar libre el canal lo antes posible para iniciar un nuevo intento.

Dotando a las estaciones de la capacidad de detectar en tiempo real que sus transmisiones se superponen (colisionan) con otras, aquellas podrían interrumpir su transmisión cuando eso sucediera, ahorrando tiempo de “mal uso” del medio. El mecanismo de detección de colisiones (CD), por tanto, reduce la duración de las colisiones en caso de que éstas se produzcan.

La realización del mecanismo CD depende de las características del medio físico de transmisión, de la técnica de codificación/modulación empleada e incluso de la topología de la red y de las variaciones en los niveles de señal en cada punto de la red de las transmisiones procedentes de cualquier estación. La detección de colisiones es, por tanto, una tarea de la capa física aunque ésta no toma decisiones y simplemente notifica de la ocurrencia de colisión a la subcapa MAC, que es la que ejecuta el algoritmo de acceso CSMA/CD.

En algunos contextos, el mecanismo CD no está exento de problemas; por ejemplo, en las redes vía radio la atenuación de las señales es muy grande (como mínimo, aumenta con el cuadrado de la distancia), por lo que, a veces, puede ser muy difícil detectar que una transmisión con un nivel de señal fuerte se superpone con otra cuyo nivel es varios ordenes de magnitud más débil. (Este es el llamado *captura o enmascaramiento*.)

El protocolo CSMA/CD (*Carrier Sense Multiple Access / Collision Detection*), es un mecanismo de acceso aleatorio a un medio de transmisión compartido, ejecutado de forma distribuida por todas las estaciones conectadas a dicho medio.

CSMA/CD representa el máximo refinamiento de una serie de técnicas de acceso aleatorias propuestas a partir de finales de la década de 1960 (Aloha, Aloha-ranurado, CSMA).

Por tanto, CSMA/CD pretende optimizar el rendimiento mediante un mejor aprovechamiento del medio, para lo cual proporciona un mecanismo que aborta toda transmisión de información cuando se detecta una colisión. Sin entrar en detalles como se efectúa, cuando se descubre una colisión, se genera una señal de atasco (*jam*) durante un cierto tiempo, con el fin de conseguir que todas las estaciones de la red conozcan que el medio debe dejarse libre de inmediato para iniciar un nuevo proceso de contienda para conseguir ganar acceso.

#### 4.6 Algoritmo de back-off exponencial binario truncado

Aun cuando el mecanismo CSMA reduce la probabilidad de colisiones, éstas aun pueden darse debido a la existencia de un retardo de propagación no nulo y a la característica 1-persistente citada anteriormente. También se ha visto que dos o más estaciones que detectan colisión en sus transmisiones, proceden a abortar dichas transmisiones (el mecanismo CD).

Surge ahora la cuestión de *cuándo* deben las estaciones en contienda proceder a la retransmisión de sus tramas que han colisionado. Si lo hicieran inmediatamente, al detectar el medio libre, se volvería a tener colisión, y así sucesivamente. Se produce, entonces, un fenómeno de “sincronización perniciosa” de las estaciones en contienda.

La solución es que cada estación, tras experimentar una colisión, espere un tiempo aleatorio (el llamado tiempo de *back-off*) antes de volver a intentar la transmisión. Con esto se “diseminan” en el tiempo los intentos de acceso al medio por parte de las estaciones; aquella estación que haya obtenido aleatoriamente el tiempo de espera menor será la que transmita en primer lugar; las demás estaciones lo intentarán más tarde pero, al observar el medio ya ocupado (mecanismo CS), no transmitirán.

La probabilidad de una nueva colisión es equivalente a la probabilidad de que dos o más estaciones calculen exactamente el mismo tiempo de espera. El tiempo de espera se escoge al azar, con probabilidad uniforme, dentro de un intervalo comprendido entre 0 y un tiempo máximo  $T_{max}$ .

La elección de  $T_{max}$  supone un compromiso entre la posibilidad de nueva colisión y el tiempo medio de retardo de acceso. En efecto, si  $T_{max}$  es grande, se tiene una probabilidad de nueva colisión ínfima (es muy poco probable que dos estaciones obtengan exactamente el mismo tiempo de espera), pero el tiempo medio de retardo en el acceso es elevado (aun cuando el medio esté libre tras la colisión, ninguna estación lo ocupará inmediatamente).

Por el contrario, si  $T_{max}$  es pequeño, el tiempo medio en que las estaciones volverán a transmitir será más reducido a costa del aumento de la probabilidad de una nueva colisión. Puede verse que el  $T_{max}$  óptimo será aquel que maximice la probabilidad de ocupación correcta del medio (es decir, ni desocupado ni en colisión) tras una colisión y que dependerá de la carga (número de estaciones en contienda) de la red en aquel momento.

La solución adoptada en las redes CSMA y en especial Ethernet, que se estudiará en las siguientes secciones, es el algoritmo de *back-off exponencial binario truncado* que intenta adaptarse a la carga estimada de la red en tiempo real. Dicho algoritmo parte de valores de  $T_{max}$  muy pequeños (para minimizar el retardo de acceso) pero los aumenta rápidamente (exponencialmente) ante la presencia de colisiones repetidas (que estimativamente indican una carga –número de estaciones en contienda-elevada). Véase la descripción de dicho algoritmo:

- a) El rango de tiempo de espera aleatoria está *discretizado* en ranuras temporales (*time slots*) cuya duración es de 2 veces el tiempo de propagación máximo entre estaciones ( $2T_p$ ) o, lo que es lo mismo, el tiempo mínimo de duración de trama ( $T_{trans}$  mínimo).
- b) Sea  $n$  el número de intentos de transmisión de una trama realizados hasta el momento (siendo  $n=1$  tras el intento inicial de transmisión de una trama).
- c) El tiempo de espera de  $r$  ranuras se escoge al azar, con igual probabilidad, dentro del intervalo  $0 \leq r < 2^k$ , siendo  $k = \min(n, 10)$
- d) El número máximo de intentos de acceso para transmitir una trama es de 16 ( $n \leq 16$ )

Es decir, una estación que observa colisión al intentar transmitir una trama nueva no esperará a retransmitir con probabilidad  $\frac{1}{2}$  y esperará 1 ranura de tiempo con probabilidad  $\frac{1}{2}$ . Si vuelve a observar colisión ( $n=2$ ), no esperará a retransmitir con probabilidad  $\frac{1}{4}$  y esperará 1, 2 o 3 ranuras con probabilidades de  $\frac{1}{4}$  respectivamente.

Si aún observa colisión ( $n=3$ ), escogerá esperar a retransmitir durante 0, 1, 2, ..., 7 ranuras con probabilidades de  $\frac{1}{8}$  respectivamente. Mientras se produzca colisión, el proceso continúa, aumentando en potencias de dos (exponencial binario) el número de ranuras, hasta llegar a 1023. Si se llega a esta situación (poco probable e indicando una carga extremadamente alta en la red), el algoritmo aún permite 6 intentos de retransmisión adicionales pero sin aumentar el número de ranuras del intervalo de tiempo de espera que se queda en 1023 (truncado). Si, tras 16 intentos, no se consigue transmitir la trama con éxito, el protocolo de subcapa MAC desiste en su empeño y notifica del error (*exceso de colisiones*) a su capa usuaria por medio de una primitiva de indicación. La probabilidad de que esto suceda en una red Ethernet es despreciable y normalmente indica de la existencia de algún problema serio (líneas de transmisión con reflexiones por mala terminación, estaciones con tarjetas de red defectuosas, etc.).

## 4.7 Estándares IEEE para redes locales

En Febrero de 1980, en el seno del IEEE (*Institute of Electrical and Electronic Engineers*), se constituye el comité 802, al mando de Marius Graube (de Tektronic), con el fin de promover el desarrollo de estándares para redes de área local. Así, la serie de estándares IEEE 802.x (donde  $x$  es un número de 1 en adelante) ha venido definiendo los niveles arquitectónicos inferiores (capa física y de enlace de datos según el modelo OSI) para redes locales.

En este ámbito, la organización ISO (*International Standards Organization*) también emite sus estándares, totalmente equivalentes a los anteriores, bajo la denominación ISO 8802-x. En la figura 4.11 se esquematizan los diferentes estándares y su interrelación.

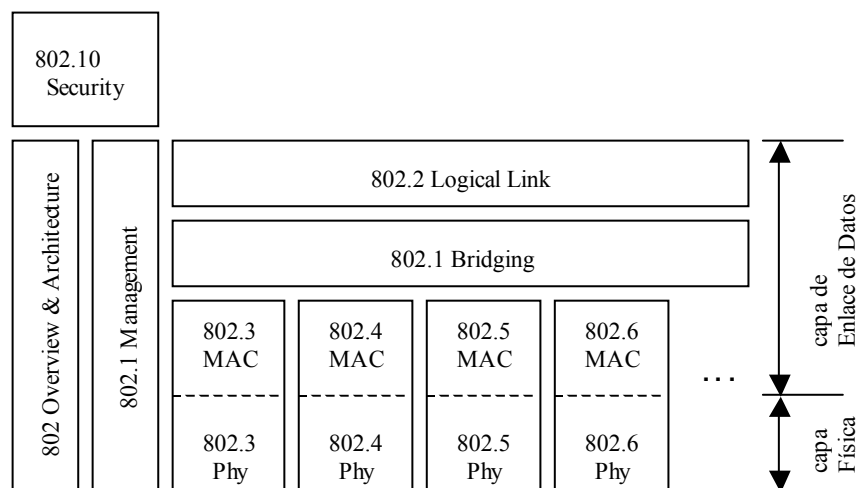


Fig. 4.11 Representación de los diferentes estándares IEEE 802 y su interrelación

La lista de estándares IEEE 802 ha ido creciendo con el tiempo, tanto por la incorporación de nuevos estándares como por la revisión/ampliación de los ya existentes (que se representan mediante IEEE 802.xy, donde  $x$  es el número que denota el estándar e  $y$  es una letra, de la “a” en adelante, que denota la ampliación/revisión llevada a cabo).

Los principales estándares IEEE de la serie 802 son los siguientes:

1. 802.1: Describe el marco general de la serie de estándares, así como aspectos de gestión de red e interconexión entre redes a nivel MAC.
2. 802.2: Describe el LLC (*Logical Link Control*), subcapa superior de la capa de enlace de datos OSI, común a todas las redes locales.
3. 802.3: Describe una familia de redes locales basadas en el protocolo de acceso aleatorio de tipo CSMA/CD. Inicialmente describía una red local con topología en bus, de cable coaxial, y velocidad de 10 Mbit/s (la conocida *Ethernet*, propuesta por Xerox, Digital e Intel en la década de 1970); las numerosas revisiones posteriores introducen topologías en estrella, de cable de pares trenzados o fibra óptica, y velocidades de 100 Mbit/s y 1 Gbit/s.
4. 802.4: Describe *Token Bus*, una red con topología en bus y protocolo de acceso basado en pase de testigo (a partir de una propuesta de red de General Motors para entornos de producción); actualmente en desuso.
5. 802.5: Describe *Token Ring*, una red con topología en anillo, velocidades de 4 y 16 Mbit/s y protocolo de acceso basado en pase de testigo (a partir de una propuesta de IBM hacia 1980); si bien a finales del siglo pasado llegó a tener una implantación del orden del 10-15% de las instalaciones de redes locales, actualmente está en desuso.

6. 802.6: Describe DQDB (*Dual Queue Dual Bus*), una red de área metropolitana con topología de doble bus, velocidades de decenas de Mbits/s y sofisticado protocolo de acceso basado en una cola distribuida; si bien en la década de 1990 tuvo cierta implantación, su futuro es incierto.
7. 802.7: Describe recomendaciones para la realización de redes locales en medios de transmisión de banda ancha (*broadband*).
8. 802.8: Describe recomendaciones para la realización de redes locales en medios de fibra óptica.
9. 802.9: Describe recomendaciones para la integración de voz y datos en redes LAN.
10. 802.10: Describe procedimientos de seguridad (cifrado en niveles arquitectónicos inferiores) para redes locales.
11. 802.11: Describe a las WLAN (*Wireless LAN*), redes locales sin hilos basadas en la transmisión por microondas o infrarrojos y protocolo de acceso aleatorio.
12. 802.12 Describe la red 100 VG-AnyLAN, una red con topología en árbol, velocidad de 100 Mbit/s y protocolo de acceso por demanda (propuesta por Hewlett-Packard y otros); actualmente en desuso.

#### 4.8 El estándar IEEE 802.2 (LLC)

El estándar IEEE 802.2 (o ISO 8802-2) describe la subcapa superior de la capa de enlace de datos, siendo sus características principales:

- a) Utilizando los servicios ofrecidos por la subcapa MAC (*Medium Access Control*), proporciona servicios a la capa de red.
- b) Proporciona capacidad de direccionamiento interno (a nivel de sistema) mediante los L-SAP y control de errores y de flujo (opcionalmente).
- c) Su protocolo interno (o de subcapa) está basado en la familia de protocolos de enlace HDLC.
- d) Independiza las capas superiores de las particularidades de cada LAN.

El LLC ofrece tres niveles (*tipos*) de implementación y, por tanto, de *servicios*:

1. *Tipo 1 (sin conexión)*: intercambio de las L-SDU entre usuarios de LLC, sin establecimiento de conexión; transmisión aislada (sin relación lógica) de las L-SDU; no garantiza la entrega de las L-SDU, sólo que éstas sean correctas (principio del *mejor esfuerzo*); adecuado para diseminación (*broadcast*) de datos no críticos, donde se requiera la máxima simplicidad y el mínimo retardo a costa de una fiabilidad absoluta (por ejemplo, muestreo periódico de sensores, aplicaciones en tiempo real, etc.) o, también, cuando en las capas superiores (típicamente la de transporte) se tengan mecanismos de recuperación de errores.

2. *Tipo 2 (orientado a conexión)*: intercambio de L-SDU entre usuarios LLC a través del establecimiento, mantenimiento y liberación de conexiones; entrega de datos garantizada (sin pérdidas, manteniendo la secuencia) y con control de flujo; adecuado cuando se requiere fiabilidad ante todo (por ejemplo, transferencia de ficheros) o cuando las capas superiores no tengan la capacidad de tratamiento de errores (por ejemplo, algunas implementaciones de la arquitectura SNA de IBM).
3. *Tipo 3 (sin conexión confirmado)*: intercambio de L-SDU entre usuarios de LLC sin conexión pero con confirmación de llegada de cada unidad enviada; adecuado donde se requiera el diálogo con muchos destinos simultáneamente, pero se requiera la máxima simplicidad (por ejemplo, en el control automatizado de cadenas de producción).

Actualmente, dadas las bajas tasas de error que se observan en la transmisión por los medios físicos, y dada la existencia generalizada de capas de transporte “potentes” (que corrigen, bajo una filosofía extremo a extremo, errores de comunicación de toda índole), solamente se utiliza el servicio LLC de tipo 1, es decir, el más básico.

En realidad, puede decirse que dicho servicio no hace prácticamente nada: simplemente “refleja” los servicios de subcapa MAC, añadiendo una cabecera con capacidad de direccionamiento de L-SAP origen y destino. Las figuras 4.12 y 4.13 muestran las primitivas del servicio LLC de tipo 1 y las posibilidades de direccionamiento.

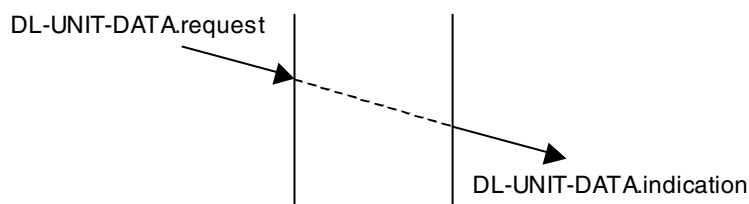


Fig. 4.12 La única primitiva de servicio asociada al servicio LLC de tipo 1

DL-UNIT\_DATA.request (dir. origen, dir. destino, datos, prioridad)

DL-UNIT\_DATA.indication (dir. origen, dir. destino, datos, prioridad)

Donde:

- a) *dir. origen/dir. destino* son una combinación de los SAP (*Service Access Point*) LLC y MAC
- b) *Datos* son los datos del usuario de LLC, es decir, las L-SDU o N-PDU
- c) *prioridad* es un parámetro pasado a la sub-capas MAC para realizar tareas de priorización en el acceso y transmisión de unos datos respecto a otros (por ejemplo, IEEE 802.4 e IEEE 802.5 lo usan; en cambio IEEE 802.3, en su versión original, ignora ese parámetro)

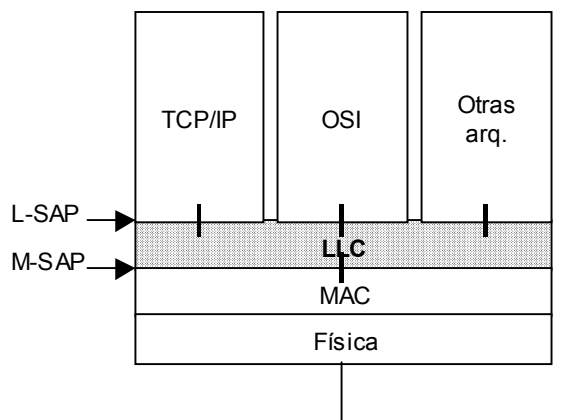


Fig. 4.13 Uso del direccionamiento, a través de los L-SAP, para distinguir diversas "pilas" de protocolos en una red local mixta donde conviven diferentes sistemas con diferentes arquitecturas de comunicaciones en los niveles de red y superiores

Ejemplos de algunos valores definidos para los L-SAP:

- 0100000 dirección (individual) de entidad de gestión de subcapa LLC
- 11000000 dirección (de grupo) de entidad de gestión de subcapa LLC
- 01111111 dirección del protocolo de red OSI
- 01010101 dirección del protocolo SNAP (*Sub-Network Access Protocol*)
- 0110000 dirección del protocolo IP de la arquitectura ARPA

Como se dijo al principio, el protocolo de subcapa LLC está basado en la familia de protocolos de enlace HDLC de ISO (véase dicho estándar para más información). Para el caso del servicio tipo 1, no se utilizan tramas de supervisión ni de establecimiento/liberación del enlace. Solamente se hace uso de las tramas de tipo no numerado (UI, XID, TEST).

Para los servicios de LLC de tipo 3, y especialmente de tipo 2, el conjunto de primitivas de servicio es más complejo y rico, así como el protocolo de subcapa utilizado. Dada la casi nula implantación de dichos tipos de servicio, no se comentarán aquí (sin que ello quiera decir que carezcan de un interés conceptual importante).

## 4.9 El estándar IEEE 802.3 (Ethernet)

El estándar IEEE 802.3 describe el nivel físico y el subnivel MAC de una familia de redes de área local que usan un medio de transmisión de difusión (con topología de bus en su origen) al que acceden las estaciones según un protocolo de acceso aleatorio de tipo CSMA/CD (*Carrier Sense Multiple Access / Collision Detection*).

### 4.9.1 Introducción

Los antecedentes de este estándar se sitúan hacia el año 1974 en el centro de investigaciones de Xerox Corporation en Palo Alto (California, USA). Allí, los ingenieros David R. Boggs y Robert M. Metcalfe, entre otros, diseñaron una red basada en el protocolo CSMA/CD, operando a 2,94 Mbit/s, que conectaba 100 estaciones de trabajo a lo largo de un bus de 1 km de longitud. Poco después, la alianza entre las compañías Digital, Intel y Xerox propone una norma industrial para conectividad local, a la que llamaron *Ethernet*, operando a 10 Mbit/s.

La norma IEEE 802.3 difiere ligeramente de la norma *de facto* Ethernet y, además, describe a toda una familia de redes basadas en el protocolo CSMA/CD.

A efectos de permitir el uso de topologías y medios de transmisión distintos, la norma divide la capa física en varias entidades funcionales, con su interfaz entre ellas, donde:

- a) MAC (*Medium Access Control*): subcapa inferior de la capa de enlace de datos, para el control del acceso al medio físico de transmisión
- b) PLS (*Physical Layer Signalling*): subcapa superior de la capa física.
- c) AUI (*Attachment Unit Interface*): interfaz de la unidad de conexión, entre la MAU y la PLS.
- d) MAU (*Medium Attachment Unit*): unidad de conexión al medio físico de transmisión (también denominada *transceptor*); se divide en:
  1. MDI (*Medium Dependant Interface*): interfaz dependiente del medio físico de transmisión
  2. PMA (*Physical Medium Adapter*): adaptador al medio físico de transmisión

### 4.9.2 Subcapa MAC de IEEE 802.3. Servicio y protocolo

Las redes IEEE 802.3 usan la variante de CSMA denominada *1-persistente*, que significa que las estaciones que desean transmitir, lo hacen tan pronto como detectan el medio libre.

Este algoritmo “agresivo” intenta reducir los tiempos muertos en el medio, ocupándolo lo más pronto posible, con el fin de reducir el retardo de acceso. El inconveniente es que, cuando la carga en la red

empieza a ser notable, habrá frecuentemente estaciones esperando transmitir tan pronto el medio devenga libre; ello conduce inevitablemente (e independientemente del retardo de propagación) a que se produzcan las llamadas *colisiones secundarias*, justo después del fin de una transmisión. Para resolver este problema ya se ha presentado el algoritmo de espera tras colisión (algoritmo de *back-off*), estudiado en la sección 4.6.

La misión de la subcapa MAC de IEEE 802.3 es gestionar el acceso al medio de transmisión compartido y garantizar que:

- a) se detecten todas las colisiones en todas las situaciones
- b) se “resuelvan” todas las colisiones, es decir, se retransmitan las tramas todas las veces que haga falta hasta tener una transmisión exenta de colisión. (Los errores en la transmisión de las tramas debidos a otras causas -por ejemplo, al ruido- no son competencia de la subcapa MAC, sino de la subcapa LLC o, en última instancia, de la capa de transporte.)

Tal como se ha visto en 4.24, existe una longitud mínima de trama transmitida, para detectar efectivamente cualquier colisión. Dicho requisito es que  $T_{trans} > 2T_p$ , es decir que  $a < 1/2$ .

Otro modo de ver lo anterior es que  $L_t > 2RD/V_p$ , donde  $L_t$  es la longitud de la trama (en bits),  $R$  es la tasa de transmisión (en bit/s),  $D$  es la distancia máxima entre estaciones (en m) y  $V_p$  es la velocidad de propagación de las señales (en m/s) a través del medio físico compartido. Es decir, en las redes Ethernet existe un compromiso velocidad-distancia inevitable que tiene su efecto a la hora de “escalar” dichas redes. Así, manteniendo igual la longitud mínima de trama, si se aumenta por 10 la velocidad de transmisión (pasar, por ejemplo, de la Ethernet tradicional a 10 Mbit/s a la denominada *Fast Ethernet* a 100 Mbit/s), la distancia máxima entre estaciones debe reducirse en ese mismo factor.

El tiempo mínimo de transmisión de trama también se denomina *intervalo de contienda*. Dicho intervalo es aquel, al inicio de cada transmisión, en el que puede darse una colisión. Superado dicho tiempo sin haberse producido una colisión, ésta ya no puede producirse y se dice que la estación transmisora ha “capturado” el medio físico con éxito.

### 4.9.3 Configuraciones topológicas de la tecnología Ethernet

Ethernet ha evolucionado desde sus inicios hacia diversas posibilidades topológicas, aunque su funcionamiento básico sigue siendo esencialmente el mismo.

La figura 4.14 muestra la configuración clásica mediante un bus.



Fig. 4.14 Configuración clásica Ethernet en bus

En los últimos años, a raíz de la emergencia de las estructuras de cableado estructurado, se han basado las redes en un dispositivo central o *hub*, que concentra en un punto el acceso CSMA/CD de Ethernet, ya sea a 10 o a 100 Mbit/s, tal como aparece a continuación.

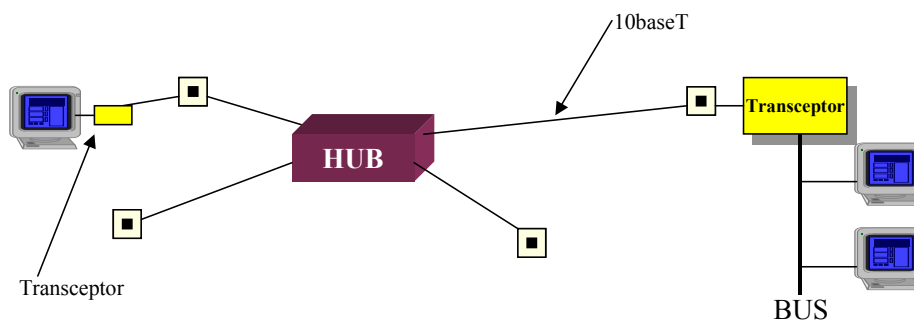


Fig. 4.15 Configuración basada en cableado estructurado y Hub central

Ahora bien, únicamente se ha considerado hasta ahora el uso de acceso tradicional mediante protocolo CSMA/CD. Pero debe decirse que en los últimos años se ha desarrollado un procedimiento de acceso radicalmente eficiente, mediante el empleo de un conmutador de tramas Ethernet, mediante el cual no existe el riesgo de colisión. Consecuentemente, el tamaño de la red no viene determinado por CSMA/CD, y en la práctica queda limitado por otras cuestiones tales como la atenuación del cableado.

En esta nueva solución, el caudal neto coincide con el disponible, puesto que no hay penalización por acceso al medio ni colisión. El precio a pagar se encuentra en el equipo conmutador, cuyo coste resulta elevado debido a la potencia de cómputo necesaria para transmitir la información hacia su destinatario correspondiente.

La figura 4.16 muestra un esquema donde se ha aplicado un conmutador como elemento de interconexión de los equipos cliente, equipados con tarjeta de red Ethernet.

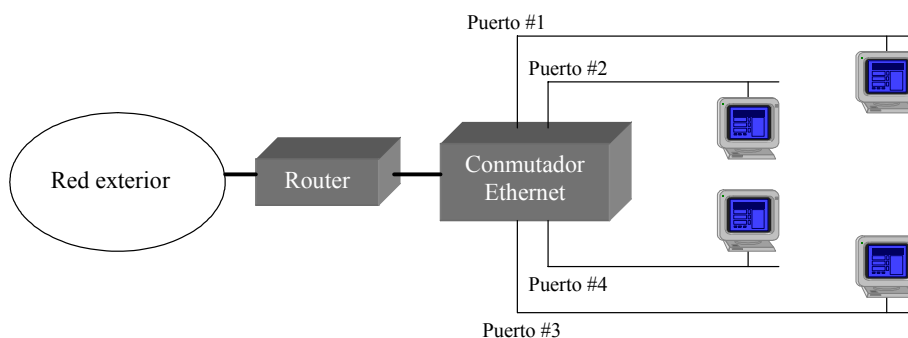


Fig. 4.16 Configuración basada en un conmutador Ethernet

#### 4.9.4 Definición de trama en una red Ethernet

Para el caso de una red Ethernet, considerando que la distancia extremo-extremo prevista en el estándar puede llegar a  $d = 2,5$  Kms, emplea una tasa de bit de  $R = 10$  Mbit/s y que la velocidad de propagación de las ondas en el medio es de  $v = 2/3 \cdot c = 2 \cdot 10^8$  m/s, de acuerdo a la expresión 4.24, se tiene que  $L_{min} = 250$  bits = 31,25 bytes.

Consecuentemente, el tamaño de la trama definida no debe ser nunca inferior a 31,25 bytes, con el fin de garantizar la detección de colisión, en el peor de los casos posibles (es decir, cuando esté situado en un extremo de la red).

Teniendo en cuenta estas consideraciones, Ethernet especifica un tamaño mínimo de trama de 64 bytes, como muestra la figura 4.17. El formato de trama Ethernet, con un tamaño de datos nunca inferior a 46 bytes, queda por tanto justificado. Nótese que a efectos de tamaño, el preámbulo no contabiliza.



Fig. 4.17 Formato de trama Ethernet

De este modo se define el *tiempo de ranura (Slot Time)* como el tiempo mínimo necesario para detectar una colisión. Una forma más útil de referir este concepto es mediante el *tamaño de ranura (Slot Size)* que es el número de bytes que pueden transmitirse en un tiempo de slot. En el caso de Ethernet, el tamaño de slot es de 64 bytes (la longitud mínima de la trama) tal como se ha visto.

#### 4.9.5 Fast-Ethernet, Gigabit Ethernet

Como se ha visto, el rendimiento real de las redes Ethernet está seriamente comprometido por el procedimiento de acceso CSMA/CD. Puede afirmarse que el rendimiento se encuentra entre el 40% y el 80%, pudiendo llegar al 90% solamente en situaciones excepcionales (pocas estaciones y tramas largas).

En general, se puede asumir que en general se dispone de un caudal neto situado entorno al 60%. Ello supone que una red de 10 Mbit/s únicamente ofrece un caudal de unos 6 Mbit/s para todos los usuarios conectados (100 o más), lo cual hoy en día resulta en general insuficiente. La distancia extremo-extremo entre dos estaciones en Ethernet está en los 2 Kms.

La necesidad de nuevos requisitos de transmisión propiciaron la evolución del estándar hacia tasas de transmisión superiores, llegando Fast-Ethernet a 100 Mbit/s (IEEE 802.3u), con el argumento de la compatibilidad con Ethernet, que resulta fundamental para el crecimiento de la red, puesto que permite continuar con los equipos existentes al tiempo que puede procederse a la migración hacia la nueva

tecnología. Sin embargo, motivos físicos debidos a la propagación de la señal en el cable motivan que el tamaño de la red quede limitada los 200 m.

De todos modos, aunque para las aplicaciones actuales suele ser suficiente, incluso 100 Mbit/s puede quedar obsoleto en breve, por lo que la tecnología ha continuado buscando la solución para el incremento de caudal. Por ello, recientemente ha entrado a formar parte de los estándares (802.3z) el nuevo Gigabit Ethernet, operando a 1 Gbit/s, aunque para operar bajo CSMA/CD aparece un problema de tamaño de red, que quedaría limitado a los 20 m. Para evitar este problema, se ha aumentado el tamaño mínimo de trama, pasando de 512 bit en Ethernet y Fast-Ethernet a los 512 bytes (este cambio ha sido denominado *carrier extensión*). En esta situación, el tamaño de extremo a extremo de la red recupera los 200 m que también define el estándar para Fast-Ethernet.

La compatibilidad entre equipos 10/100 Mbit/s se puede conseguir en la situación de la figura 4.18:

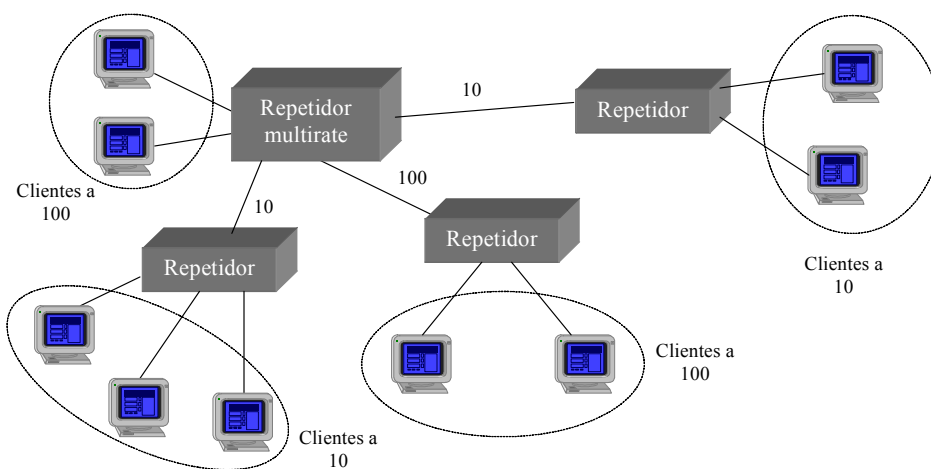


Fig. 4.18 Configuración con subredes a diversas velocidades

Una importante consecuencia es que la conexión a un HUB que funcione por contienda (CSMA/CD convencional) únicamente permite 100 m de cable hasta cada estación.

La tecnología Gigabit Ethernet queda estandarizada en 802.3z, para Acceso CSMA/CD, es decir, por contención. Asegura la compatibilidad con las soluciones a 10 y 100, con el mismo formato de trama. Ofrece soporte para cableado de cobre, mediante cables:

- a) De 25 m, 1000baseCX de 150  $\Omega$ ,
- b) o 100 m usando UTP de categoría 5, estandarizado en IEEE 802.3ab, y recomendado por la *Gigabit Ethernet Alliance*.

Ahora bien, si se emplea un conmutador, el diámetro de la red pasa a no tener más límite que el de la propia transmisión de la señal por el cableado, por atenuación o distorsión.

El funcionamiento de la red Gigabit Ethernet que emplea el mecanismo por contención es una extensión de la solución empleada para 10 o 100 Mbit/s, tal como se estudiará a continuación.

#### 4.9.6 Consideraciones en la compatibilidad entre Ethernet de diversas capacidades

De la expresión 4.23 se puede deducir de forma muy simple un muy importante resultado asociado al procedimiento para emplear una tasa de bit superior: de los parámetros contenidos en esta expresión,  $v$  es una constante dependiente del medio,  $L$  debe mantenerse por motivos de compatibilidad. En consecuencia, si  $R$  aumenta, no hay más remedio que disminuir a  $d$  del mismo modo. En suma,

*aumentar la capacidad en un factor  $n$  provoca la reducción del tamaño de la red en ese mismo factor.*

Por ejemplo, supóngase una red Fast-Ethernet. La diferencia fundamental con Ethernet es la tasa  $R$ , que en este caso ha aumentado de 10 a 100 Mbit/s. Por lo tanto, el diámetro de la red habrá reducido hasta los 250 m, que el estándar deja en 210 m. Por lo demás, el formato de trama es el mismo.

Nótese las particularidades de transmisión de un equipo dedicado a la interconexión de una red Ethernet hacia Fast-Ethernet:

- a) En el sentido Fast-Ethernet hacia Ethernet, debe existir una cola o *buffer* que permita almacenar la trama mientras ésta es transmitida a velocidad 10 veces inferior hacia Ethernet. Esta cola actúa con funciones de retención de los bytes que no pueden ingresar en Ethernet al mismo ritmo que salen de Fast-Ethernet. Debe observarse que no es posible que todo el caudal de Fast-Ethernet (100 Mbps teóricos) pueda introducirse hacia Ethernet (10 Mbps teóricos), evidentemente.
- b) En el sentido opuesto, desde Ethernet a Fast-Ethernet, se requiere un *buffer* para acoger la trama, de modo que pueda entregarla hacia Fast-Ethernet a tiempo de bit contraído a la décima parte, evitando que la disminución del tiempo de bit suponga que se agote la información recibida a 10 Mbit/s, y no se mande la trama completa. Por tanto, el *buffer* debe ser llenado antes de iniciar la transmisión hacia Fast-Ethernet.

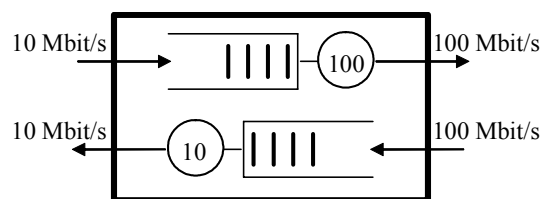


Fig. 4.19 Circulación de tráfico entre redes Ethernet y Fast-Ethernet

En el caso de aumentar la velocidad hasta 1 Gbit/s, el estándar Gigabit Ethernet ha encontrado un grave problema, pues la red quedaría en unos paupérrimos 25 m. Para evitar este inconveniente, ha

sido necesario idear un mecanismo que permita disponer de una distancia extremo a extremo superior. En la práctica, el estándar habla de 210 m. ¿Cómo ha sido posible?

#### 4.9.7 Incremento de capacidad a 1000 Mbit/s

A la vista nuevamente de la expresión 4.23, el hecho de fijar  $d$  y  $R$ , siendo  $v$  un parámetro explícitamente físico, obliga a alterar el valor  $L$ . desgraciadamente esta es la solución que ha debido adoptarse. La cuestión es llevarlo a cabo manteniendo la compatibilidad.

Para comprender la solución empleada, debe recordarse que los 46 bytes del campo de datos de la trama Ethernet (fig. 4.17), deben estar siempre presentes, incluso aunque los datos reales que deben encapsularse sean de tamaño inferior. En este caso, se procede a rellenar hasta alcanzar este tamaño mínimo. La cuestión es entonces determinar la longitud real de los datos, lo cual podrá llevarse a cabo en la entidad destino por medio del protocolo encapsulado, gracias al campo de tipo que permitirá reconocer ese protocolo operando encima de Ethernet. Obsérvese que este mecanismo no ofrece inconvenientes, más que la simple disminución de rendimiento cuando se desean enviar menos de 64 bytes, causado por el relleno artificial que debe efectuarse.

Justamente este mismo esquema es el que ha permitido resolver el problema para aumentar el tamaño de trama mínimo en Gigabit Ethernet al tiempo que se mantiene la compatibilidad. Gigabit Ethernet define un tamaño mínimo de slot 512 bytes, esto es, 8 veces superior al de Ethernet. En este caso, el relleno es muy superior.

Ahora bien, este tamaño únicamente afecta al tiempo de *slot*, pero no al tamaño mínimo de la trama, que ahora quedará ubicada dentro de un *slot* de mayor tamaño que en el caso de Ethernet o Fast-Ethernet. El relleno que puede ser necesario se realiza mediante símbolos especiales, diferentes a los que pueden aparecer en los bytes de la trama. Este proceso es conocido como *carrier extension*. Para tramas con la portadora extendida, los símbolos especiales (distintos a los de datos) que efectúan la extensión rellenando el *slot* se incluyen al final de la trama estándar Ethernet y el CRC calculado no los tiene en consideración.

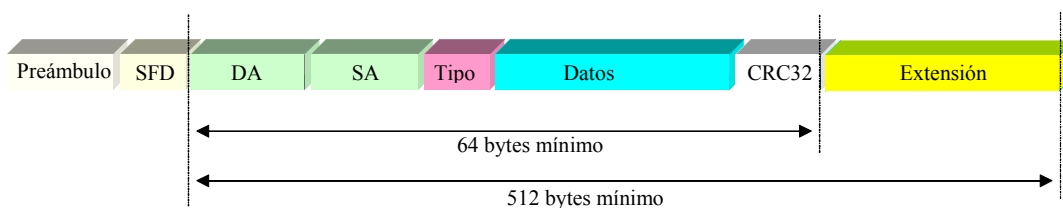


Fig. 4.20 Formato con mecanismo de carrier extension

En resumen, Gigabit Ethernet permite la interoperación con las redes 802.3 existentes. El mecanismo de *carrier extension* es una forma de garantizar los tamaños mínimos y máximos de trama con distancias extremo a extremo de la red razonables.

La posible disminución de rendimiento debido al uso de un campo de extensión artificial merece también un comentario: si con Fast Ethernet el rendimiento medio es del orden de, pongamos, el 60 %, y con Gigabit Ethernet, debido al relleno, determinásemos que su rendimiento en una situación muy pesimista es de únicamente un 30 %, resultaría que el caudal disponible para Fast Ethernet sería de  $60\% \cdot 100 \text{ Mbit/s} = 60 \text{ Mbit/s}$ , y para Gigabit Ethernet de  $30\% \cdot 1000 \text{ Mbit/s} = 300 \text{ Mbit/s}$ .

De este simple e intuitivo razonamiento se deduce que, aunque pueda existir una importante disminución del aprovechamiento neto de la red, el caudal final obtenido es sensiblemente superior en Gigabit Ethernet y, consecuentemente, ofrece gran interés para las futuras redes que deban transportar caudales elevados con acceso al medio CSMA/CD.

Sin embargo, dado que *carrier extension* es un mecanismo que desperdicia una buena parte del ancho de banda (hasta  $512 - 64 = 448$  bytes de relleno) se ha diseñado una mejora añadida, denominada *packet bursting*. *Packet bursting* consiste en el empleo de *carrier extension* combinado con la realización de una ráfaga de paquetes en transmisión. El procedimiento seguido presenta ventajas cuando se van a transmitir desde una estación varios paquetes de pequeño tamaño: tras transmitir el primer paquete, ya sea empleando *carrier extension* o no en un *time slot*, los siguientes paquetes se transmiten también hasta que se consumen 1500 bytes. Para la transmisión de la ráfaga de paquetes se respeta el valor de *inter-packet gap* (IGP) (o distancia mínima entre paquetes) definida por el estándar. Puede verse el funcionamiento de forma gráfica en la figura 4.21.

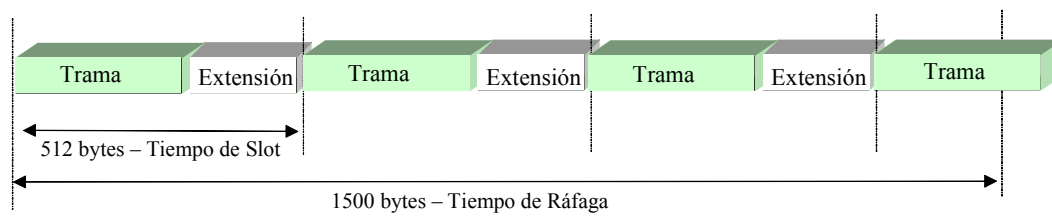


Fig. 4.21 Mecanismo de packet bursting

Con el fin de establecer los límites existentes, cabría cuestionar cuál sería la velocidad máxima a la que podría operar esta tecnología usando  $d = 210 \text{ m}$ . Para resolver este simple ejercicio, basta con considerar que el límite viene fijado por la compatibilidad, esto es, por el tamaño máximo de las tramas Ethernet, que es de 1518 bytes. Considérese pues una red Ethernet con todas las tramas de la misma longitud, que coincida con la máxima de 1518 bytes. En esta situación, la red podría operar hasta

$$R_{\max} = \frac{1}{2} L \frac{v}{d} = \frac{1}{2} 1518 \text{ bytes} \frac{2 \cdot 10^8 \text{ m/s}}{210 \text{ m}} = 5,78 \text{ Gbit/s} \quad (4.25)$$

Nótese que ante mecanismos de Ethernet conmutada, no existen estas limitaciones, ya que la distancia máxima viene determinada por la atenuación de la señal en el cableado, y por tanto podría ser técnicamente ilimitada. Este es el caso de enlaces Ethernet de varios kilómetros, que interconecten equipos distantes mediante tecnología Ethernet, con caudales en la actualidad de hasta 10 Gbit/s, pero que presumiblemente la evolución los llevará hacia los 100 Gbit/s y, por qué no, hacia el terabit/s.

#### 4.9.8 La solución de *Metro Ethernet Forum*

Metro Ethernet Forum es un organismo sin ánimo de lucro que tiene por principal objetivo la promulgación de las redes Ethernet ópticas para su adopción como solución tecnológica en entornos de red metropolitana. Está constituido por empresas líderes del sector, con 67 miembros a fecha de 1 de enero de 2002.

En este momento se están desarrollando las soluciones siguientes:

- a) Protección de la red Metro Ethernet. El subcomité de Transporte y Protocolos del Metro Ethernet Forum ha desarrollado un modelo de protección innovador, de modo que consigue ofrecer respuesta en 50 mseg manteniendo el funcionamiento de la red en caso de que un enlace o nodo de la red hubieran fallado, y garantizando la flexibilidad de la topología.
- b) Definición de servicios.
- c) Perfil de red para la gestión de QoS.

Este consorcio pretende potenciar la implantación de las soluciones Ethernet (hasta 10 Gbit/s en la actualidad, IEEE 802.3ae, en el marco de las redes metro.

Los servicios considerados por Metro Ethernet incluyen la emulación de circuitos (CES, *Circuit Emulation Service*) para el soporte de tráfico TDM, tal como T1/E1, T3/E3 y OC-3/12 sobre redes basadas en Ethernet. De este modo, se puede decir que esta nueva Ethernet de alta capacidad ofrece características de transmisión equiparables a las de SDH.

Las redes basadas en Ethernet ofrecen una relación prestaciones/coste muy ventajosa, así como una gran facilidad de uso. En general, puede afirmarse que Ethernet es una tecnología muy adecuada para el transporte de tráfico IP. Estas ventajas la determinan como una tecnología adecuada para el entorno metropolitano a la vista de los organismos que actualmente impulsan nuevas tecnologías, como es el caso del 10 Gigabit Ethernet Alliance (10GEA).

Uno de los aspectos fundamentales en las consideraciones de adopción de la red metro Ethernet frente a SDH / Sonet es que Ethernet puede ser empleada para el transporte de una gran variedad de propósitos, coexistiendo con SDH/Sonet. Ahora bien, las previsiones apuntan a que, aunque la convivencia está garantizada, el mercado tenderá a adoptar progresivamente Ethernet como solución de transporte, sustituyendo a la larga a SDH/Sonet.

## 5. Redes públicas de datos

### 5.1 La red digital de servicios integrados

#### 5.1.1 Introducción

El concepto de red digital de servicios integrados, en adelante RDSI, surge a mediados de la década de 1980. En dicha época la red telefónica, con aproximadamente 100 años de existencia y presencia prácticamente mundial, se utilizaba, aparte del servicio de voz, para el soporte de nuevos servicios de telecomunicación de demanda creciente (transmisión de datos entre ordenadores, facsímil, vídeo, etc.).

El soporte de esos servicios (gracias al uso del módem), y de otros que se vislumbraban para un futuro inmediato, no era del todo eficiente.

Por ello, el CCITT (actualmente el ITU-T), en sus reuniones de 1984 y 1988, define la RDSI como una red basada, a grandes rasgos, en los siguientes principios:

- a) Compromiso internacional
- b) Sustitución progresiva de la red telefónica tradicional
- c) Enteramente digital (extremo a extremo)
- d) Integración de todos los servicios de telecomunicación presentes y futuros

La RDSI soportaría/integraría los siguientes servicios:

- a) *telefonía*: de momento seguiría siendo el principal servicio y además lo mejoraría en cuanto que permite una calidad de voz algo mayor, el establecimiento de llamadas casi instantáneo (menos de 1 s), la identificación del llamante, la redirección de llamadas, las conferencias (más de 2 interlocutores), los servicios de contestador a través de mensajes, etc.
- b) *videotexto* (hoy en desuso y sustituido, probablemente, por el servicio *web*): para el acceso interactivo a bases de datos remotas, las consultas de direcciones y personas (páginas blancas, páginas amarillas), las reservas de billetes para viajes y espectáculos, las operaciones bancarias, las telecompras, etc.

- c) el correo electrónico
- d) el facsímil (o fax)
- e) la videotelefonía, la videoconferencia
- f) la telemetría, el telecontrol, la televigilancia, etc.

En cuanto a los aspectos de implantación y requisitos de diseño de las RDSI, podrían apuntarse los siguientes:

- a) implantación progresiva (durante décadas)
- b) convivencia/coexistencia con las redes existentes
- c) mejora de la capacidad de señalización del sistema telefónico tradicional (con una baja capacidad de señalización, propensa a los errores e, incluso, al fraude)
- d) uso de una red de señalización por canal común (usuario-red, red-red, fuera de la banda, basada en la conmutación de mensajes, con funciones para: el establecimiento, encaminamiento y liberación de llamadas, el acceso a bases de datos internas, la supervisión, mantenimiento y gestión de la red, la tarificación, etc.)
- e) integración de las redes de conmutación de paquetes (que, a partir de la década de 1970, mostraron una mayor eficiencia para el soporte del tráfico datos intermitente o a ráfagas)
- f) aprovechamiento de la planta de abonado o de bucle local telefónico (es decir, de los millones de km de cable de pares de cobre instalados en el mundo)

Como se muestra en las figuras 5.1 y 5.2, la RDSI integraría el acceso a todos los servicios, pero no las redes de transporte, para, eventual y posteriormente (en el siglo XXI), integrar también el transporte

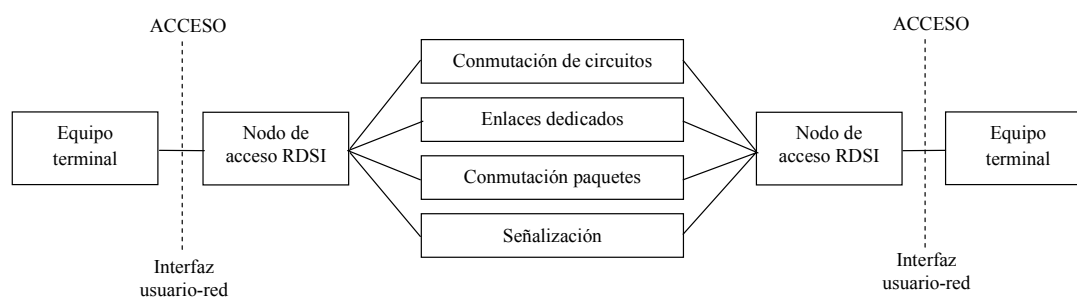


Fig. 5.1 Inicialmente la RDSI integra el acceso a todo tipo de servicios y redes, pero no las redes de transporte

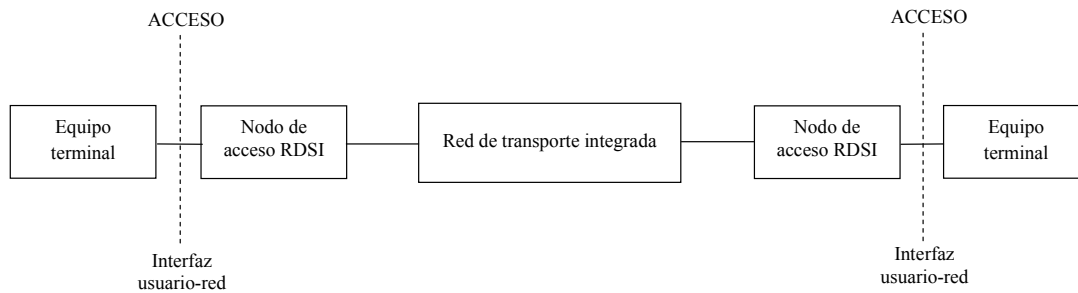


Fig. 5.2 En el futuro la RDSI integraría eventualmente también a las redes de transporte (ATM, por ejemplo)

### 5.1.2 Arquitectura de la RDSI

El acceso a la RDSI-BE (de banda estrecha) se caracteriza por:

- la *digitalización* del último tramo de la red (el acceso del terminal de usuario a la red)
- la definición de *servicios portadores* digitales, o conductos digitales, extremo a extremo
- el uso de técnicas de multiplexado en el tiempo (TDM) para el soporte de los diversos canales (de señalización y de datos de usuario)
- la definición de 2 tipos de acceso: el *básico* (para usos particulares/domésticos o de pequeñas oficinas) y el *primario* (para empresas medianas o grandes)

La arquitectura se estructura alrededor de la definición de los *puntos de referencia* y de las *entidades funcionales*, como se muestra en las figuras 5.3 y 5.4.

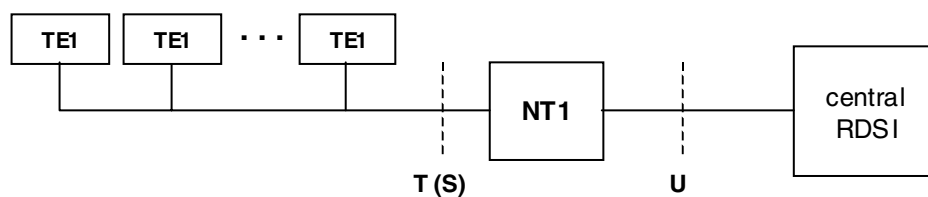


Fig. 5.3 Entidades funcionales y puntos de referencia en un entorno sencillo como un domicilio particular o pequeña oficina con teléfono y fax, por ejemplo

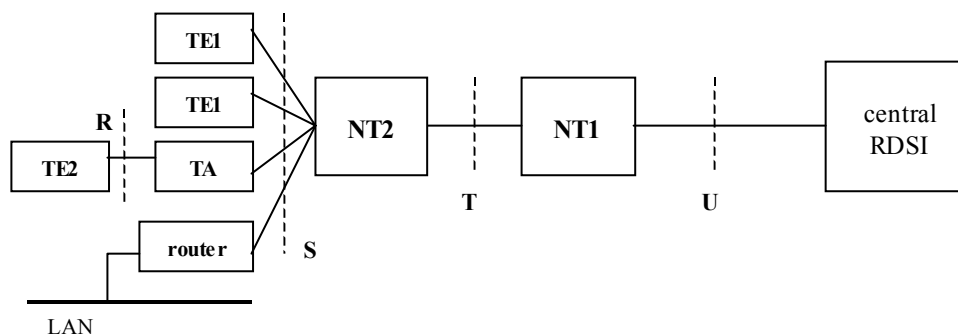


Fig. 5.4 Entidades funcionales y puntos de referencia en un entorno algo más complejo; obsérvese que cabe considerar el uso de un encaminador con interfaces S y Ethernet, por ejemplo, para conectar dos redes LAN distantes o para permitir el acceso de los equipos de la LAN a Internet

Las entidades funcionales de la arquitectura del acceso básico a la RDSI son las siguientes:

- NT1 (Network Termination 1)*: constituye la frontera usuario-red; pertenece al proveedor de los servicios de red; ofrece el conector de acceso a red (similar al telefónico tradicional); es un elemento activo que requiere de suministro eléctrico (local, por parte del usuario, o desde la central RDSI, en situaciones de emergencia); proporciona asimismo alimentación eléctrica restringida a un equipo terminal definido como de emergencia; controla el bucle local, dentro del espacio del usuario, y el bucle remoto, hacia la central RDSI; controla el acceso de varios equipos terminales de usuario; es un elemento con funciones de capa física según el modelo OSI (codificación/decodificación, multiplexado, básicamente)
- NT2 (Network Termination 2)*: elemento opcional que aparece en entornos de usuario más importantes; posee funciones de conmutación (PABX, por ejemplo); con funciones de capas 1 a 3 según el modelo OSI
- TE1 (Terminal Equipment 1)*: equipo terminal RDSI con interfaz S (véase más abajo); puede ser un terminal telefónico, un equipo de fax, un ordenador dotado de tarjeta RDSI, etc.
- TE2 (Terminal Equipment 2)*: equipo terminal no-RDSI (cualquier terminal de comunicaciones tradicional)
- TA (Terminal Adaptor)*: elemento adaptador para la conexión de equipos no RDSI, es decir TE2, a un acceso RDSI

Los puntos de referencia de la arquitectura del acceso básico a la RDSI son las siguientes:

- U*: es el interfaz entre la entidad NT1 y el nodo RDSI, es decir, el bucle de abonado (inicialmente el mismo que el telefónico ya existente formado por un par trenzado de cobre; previsiblemente en fibra óptica en el futuro). Para el acceso básico (ver más adelante) existen dos variantes del interfaz U en función del código empleado: código 2B1Q (que significa que cada grupo de 2 bits se transmiten como 1 símbolo cuaternario, es decir, que puede adoptar 4 niveles de tensión distintos), dando una tasa de 80 kbaudios en el bucle de abonado; código

4B3T (que significa que cada grupo de 4 bits se transmite como 3 símbolos ternarios, es decir, que pueden adoptar 3 niveles de tensión distintos), dando una tasa de 120 kbaudios. En este último caso se tiene cierta redundancia ( $3^3$  grupos distintos de 3 símbolos para  $2^4$  grupos de 4 bits posibles), aprovechable para la detección de errores

- b) *S*: es el interfaz entre el equipo terminal de usuario TE1 y la entidad NT1 (o la NT2, si existe); su topología es la de un bus pasivo a 4 hilos (en realidad hay hilos adicionales que soportan el suministro eléctrico a algunos TE) de hasta 1 km de longitud y que permite la conexión en derivación de hasta 8 TE1 o TA formando una especie de pequeña red local. Para el acceso básico se utiliza un código *AMI* (*Alternate Mark Inversión*). La velocidad bruta de la trama básica es de 192 kbit/s y la velocidad neta es de 144 kbit/s. Ver figura 5.7
- c) *T*: es el interfaz entre las entidades NT1 y NT2; si la entidad NT2 no existe, entonces los puntos T y S son equivalentes
- d) *R*: define algún interfaz no RDSI con equipos terminales tradicionales (por ejemplo: teléfono analógico, RS-232, X-25, etc.)

### 5.1.3 Tipos de acceso a la RDSI de banda estrecha

En un acceso RDSI se definen dos tipos de canales:

- a) *Canales B*: canales digital full-dúplex a 64 kbit/s para el transporte extremo a extremo de voz o datos de usuario; dichos canales son generalmente tratados de forma “transparente”, es decir, la RDSI no especifica el significado de los bits que transportan o, dicho de otro modo, las normas de RDSI sólo dan especificaciones de capa física para dichos canales. El significado de los bits transportados por dichos canales depende del usuario (pudiendo ser muestras de voz codificadas, imágenes de video comprimidas, datos entre ordenadores, datagramas de Internet, etc.)
- b) *Canales D*: canales digitales full-dúplex a 16 o 64 kbit/s para el transporte de toda la información de señalización, es decir de diálogo usuario-red, así como de pequeñas cantidades de datos de usuario en forma de paquetes, opcionalmente. Las normas RDSI especifican que dichos canales operan en modo paquete, dando especificaciones de capa física (como se transmiten y multiplexan los bits), de capa de enlace de datos (formato de las tramas, su significado y el protocolo que se establece para su intercambio) y de capa de red (formato de los paquetes y su significado, por cuanto llevan información de señalización de red)

El *acceso básico*, también denominado  $2B+D$ , ofrece al usuario dos canales B que, por conmutación de circuitos, pueden conectarse con cualquier destino. Así, por ejemplo, un usuario puede mantener una conversación telefónica con alguien y enviar un fax a otro destino simultáneamente, o bien acceder a Internet usando los dos canales B “en paralelo” para tener una velocidad de acceso de 128 kbit/s. El canal D lleva toda la información de señalización, es decir, de establecimiento de llamadas, de información de llamadas entrantes o de tarificación en tiempo real durante una conversación, etc.

El *acceso primario*, también denominado  $30B+D$  (en Europa) o  $23B+D$  (en USA y Japón), ofrece al usuario 30 (23) canales B que, por conmutación de circuitos, pueden conectarse con cualquier destino. Así, por ejemplo, una empresa de alojamiento de páginas web puede tener conectados sus servidores a Internet, a una velocidad cercana a los 2 Mbit/s, usando los 30 canales B de un acceso primario “en paralelo”, o bien otra empresa puede disponer de un número telefónico de atención al cliente con 30 líneas, etc. El canal D, en este caso, por llevar información de señalización relativa a 30 canales, opera a 64 kbit/s.

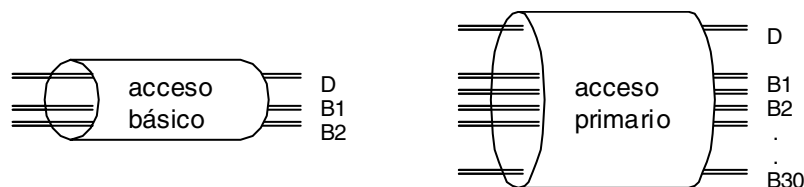


Fig. 5.5 Representación simbólica (pues todos los canales van multiplexados en el tiempo sobre una única línea de transmisión) de los dos principales accesos a la RDSI de banda estrecha

En la figura 5.6 se representa la arquitectura de la RDSI según el modelo OSI.

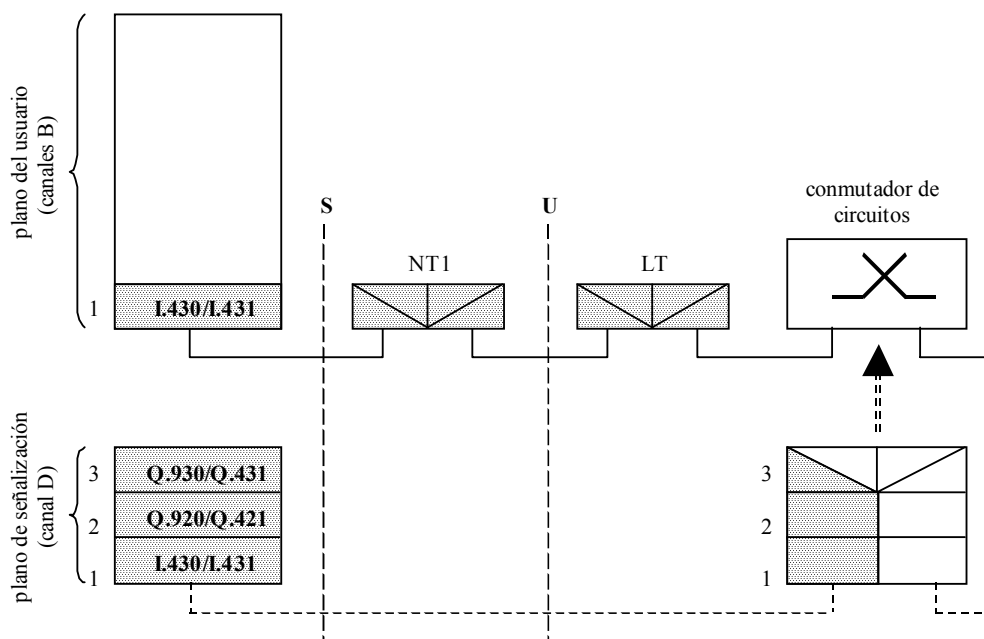


Fig. 5.6 Arquitectura del acceso a la RDSI de banda estrecha. En sombreado, las capas definidas por ésta para el plano de usuario (canales B) y para el plano de señalización (canal D). En negrita, los principales protocolos para el acceso básico/primario

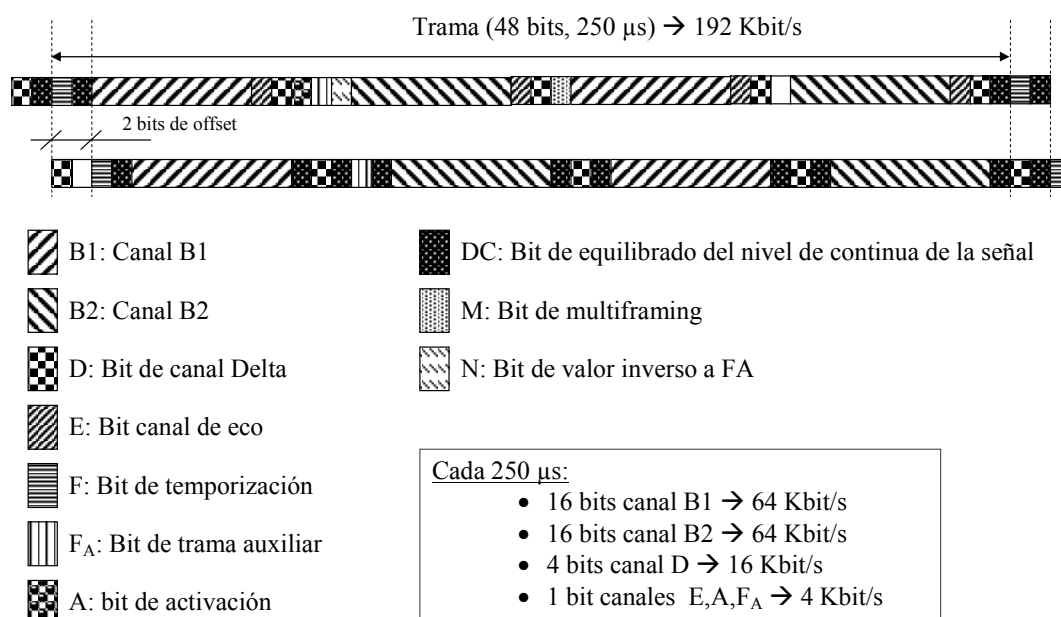


Fig. 5.7 Entramado del acceso básico RDSI

## 5.2 La red Frame Relay

Frame Relay es una propuesta de ANSI de 1990, planteada como una evolución del servicio de X.25, que incorpora las mejoras tecnológicas disponibles, para sustituir a las tecnologías de ese momento.

### 5.2.1 Objetivos

Frame Relay pretende sintetizar una tecnología articulada sobre mecanismos simples. Por este motivo, Frame Relay potencia los aspectos destinados a alcanzar esta simplicidad, entre los cuales destacan la desaparición de los elementos de gestión de la transmisión de los datos, la señalización fuera de banda y la supresión de los controles de error en los nodos intermedios.

De esta forma, Frame Relay se perfila como una tecnología que con una menor potencia en los equipos consigue mejorar la velocidad de transmisión de datos y el funcionamiento general de la red al simplificar los aspectos de gestión.

Entre las motivaciones que llevaron al desarrollo de esta tecnología, pueden distinguirse las causas funcionales y a las de implementación.

Las causas *funcionales*, que surgen como respuesta a las necesidades del avance de la tecnología y los servicios, pueden desglosarse en las siguientes:

- a) Aparición de las aplicaciones interactivas multimedia, en las cuales el usuario no se limita únicamente a recibir audio o imágenes, sino que además interacciona con otro usuario o servidor, por lo que inmediatamente surge la necesidad de garantizar la calidad de servicio. Esto es, el usuario empieza a exigir un determinado caudal con un retardo acotado y, lo que es más importante, permitiendo el paso esporádico de ráfagas de caudal, muy frecuente en tráfico de tipo multimedia.
- b) La necesidad de integrar las redes de área local con las de área extensa con el fin de permitir la comunicación entre las pequeñas redes a través de una tecnología de transporte que garantizara el caudal de transporte.

En cuanto a las causas *por motivos de implementación*, se articulan en dos aspectos fundamentales, fruto de la evolución de las técnicas de transmisión de datos y del incremento de potencia de los procesadores:

- a) La disponibilidad de un mejor soporte para la transmisión, con tasas de error de bit menores, condujo a cuestionar la necesidad de comprobar la existencia de errores en cada uno de los nodos de la red, a nivel de enlace.
- b) Este hecho, unido a equipos terminales más inteligentes gracias a una mayor potencia de cálculo, permitió asignar la responsabilidad del control de errores completamente al terminal. La importancia consecuencia que se deriva de este hecho es que los nodos de conmutación de la red quedan automáticamente liberados de esta labor, y pueden concentrar sus esfuerzos en la propia operación de conmutación de la información, de modo que esta pueda ser efectuada por dispositivos de menor coste.

### 5.2.2 Características básicas

Frame Relay es una tecnología de red cuya simplicidad afecta a dos aspectos de control fundamentales, que en otras redes como X.25 son muy importantes para su correcto funcionamiento:

En primer lugar, no hay controles de flujo, por lo que si se produce congestión en la red, el tráfico entrante puede perderse por simple desbordamiento de colas

En segundo lugar, no hay control de errores en los nodos de la red, gracias a que el medio físico proporciona una probabilidad de error de símbolo baja.

Su base se asienta en el estándar I.122 del CCITT, ofreciendo una estructura para los servicios portadores en modo paquete. Con posterioridad, el Q.921 también del CCITT demostró la utilidad del multiplexado de circuitos virtuales para protocolos de nivel de enlace. De ahí que Q.921 trata del protocolo LAPD (*Link Access Procedure for the D Channel*). La recomendación V.120, finalmente, especificó el procedimiento de multiplexado a través de interfaces S/T de la red digital de servicios integrados de banda estrecha, la RDSI.

El multiplexado es fundamental para el funcionamiento de Frame Relay.

En la actualidad, los estándares reposan en dos ramas diferentes, una de ellas constituida por el CCITT con los estándares I.233, Q.922 anexo A y Q.933, y por otro el ANSI, con los estándares T1.606, T1.618 y T1.617.

El servicio ofrecido es orientado a conexión. Durante la fase de conexión, se establece la ruta del circuito virtual y se señala a la red con el fin de garantizar parámetros de calidad al usuario (caudal y retardo). La conmutación se caracteriza por operar en modo conmutación rápida de mensajes. Este procedimiento trata de encaminar la información tan pronto la trama de datos entra al conmutador.

Frame Relay no prevé mecanismos de prioridad para los paquetes de información, de tal modo que todos ellos reciben el mismo trato por parte de la red. Sin embargo, esta afirmación deberá ser matizada más adelante puesto que aunque efectivamente la red no proporciona herramientas explícitas de prioridad, sí es cierto que existe una forma práctica que permite definir dos niveles de calidad, basados en el campo DE de la trama.

Las tramas son de longitud variable, y su tamaño viene determinado por unos indicadores (*flags*) al principio y al final.

El usuario contrata una tasa de transmisión que se denomina CIR (*Committed Information Rate*) o tasa de información comprometida, que puede interpretarse como aquel caudal que el usuario puede transmitir hacia la red y que ésta siempre debe cursar. Más adelante se comentarán otros parámetros que el usuario también puede contratar y que consienten la generación de picos de tráfico hasta cierto valor máximo.

### 5.2.3 La conmutación

Frame Relay define un servicio orientado a conexión, mediante circuitos virtuales. Esto significa que durante la fase de conexión, los nodos intermedios de la red buscan una ruta que establezca por donde pasa el circuito virtual de modo que todos los paquetes transiten por el mismo camino.

Para cada circuito un contrato de tráfico, con un perfil de caudal y de retardo de transmisión. En este contrato, se indica el valor de CIR asociado. Un circuito se establece en ambos sentidos de la comunicación, aunque no por ello el valor del CIR debe ser el mismo.

Se distinguen dos tipos de conexiones, en base a la forma en que se establecen: Los circuitos virtuales permanentes (PVC, *Permanant Virtual Circuit*) disponen siempre del mismo valor de contrato, negociado de antemano con el operador, con un establecimiento de conexión prefijado.

Por otro lado, los circuitos virtuales conmutados (SVC, *Switched Virtual Circuit*), negocian todos los parámetros de la conexión durante la fase de establecimiento. Estos últimos pueden desconectarse una vez termina la comunicación. El tratamiento de mensajes de señalización definido en los estándares para el establecimiento y desconexión del circuito queda fuera del alcance de este libro.

Cabe indicar que una vez establecidos, los PVC y los SVC reciben el mismo trato por parte de la red. Es decir, la única diferencia entre ellos se encuentra en las acciones durante la fase de establecimiento de conexión.

Tal como se ha indicado en la sección anterior, Frame Relay emplea un modo de conmutación rápida de paquetes. Para que sea efectivo, es necesario que la información de direccionamiento se encuentre al principio del paquete, de modo que el conmutador pueda conocer la ruta de destino lo antes posible para proceder a dirigir los datos hacia el puerto de salida adecuado. De acuerdo a esta necesidad, el identificador del canal virtual o DLCI (*Data Link Control Identifier*) se sitúa al principio del paquete.

El establecimiento de una ruta, ya sea para circuitos virtuales permanentes o conmutados, consiste en la programación de unas tablas en cada uno de los conmutadores, de modo que quede establecida la ruta que deben seguir los mensajes. En la tabla de conmutación se indican los pares {puerto de entrada del conmutador / número de conexión (DLCI)} para el puerto de entrada (*in*) por el que se espera la llegada, y el de salida (*out*) por el que el equipo de conmutación debe transferirlo.

Durante este proceso, el identificador DLCI puede cambiar entre enlaces de nodos pertenecientes al mismo trayecto. Nótese que ello no supone ningún inconveniente, puesto que el circuito queda claramente definido y sin ambigüedades por la concatenación de los diversos valores de DLCI adoptados en los diversos enlaces entre conmutadores desde el equipo origen hasta el destino.

Aunque la existencia de valores diferentes para un mismo trayecto pueda parecer un inconveniente, permite la administración local del valor de DLCI. Es decir, durante la fase de estableciendo de la ruta, cuando los conmutadores deben programar sus tablas, el valor del DLCI se negocia entre los dos equipos de los extremos de dicho enlace, eligiendo cualquier valor no ocupado.

De este modo, se evita tener que conocer todos los valores ya usados de DLCI en toda la red. Como el lector podrá apreciar, esto es una gran ventaja para la gestión de identificadores en Frame Relay, y no conlleva ningún tipo de perjuicio. La figura 5.8 muestra una pequeña red Frame Relay con algunas rutas definidas.

Resulta fácil encontrar los circuitos establecidos, tal como muestra la figura 5.9.

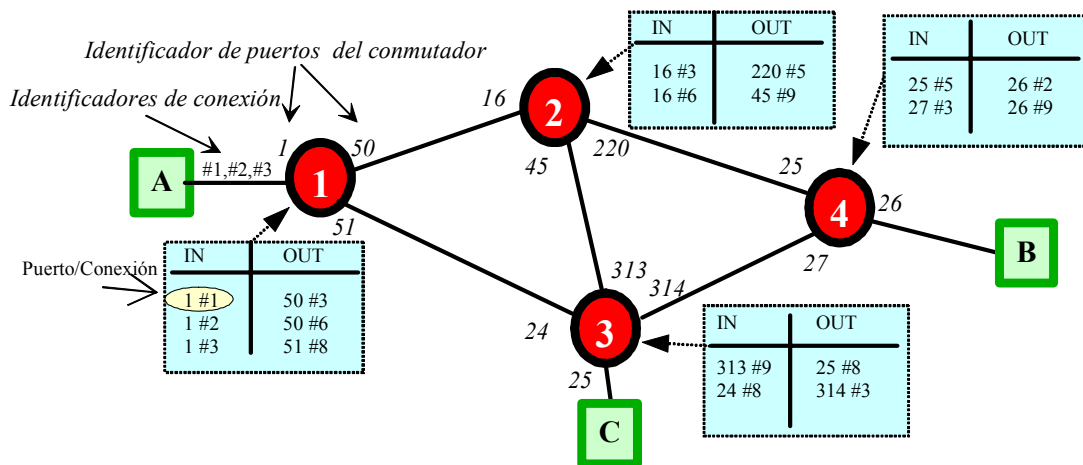


Fig. 5.8 Tablas de conmutación en una red Frame Relay

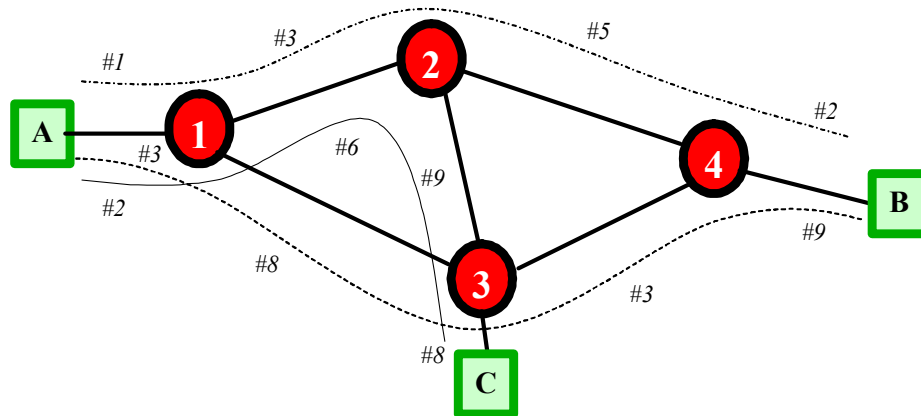


Fig. 5.9 Rutas establecidas a partir de las tablas de la figura 5.8

#### 5.2.4 Arquitectura de protocolos

Frame Relay define una arquitectura donde únicamente aparecen los dos niveles inferiores, de acuerdo a la torre de protocolos de OSI.

En esta arquitectura, el nivel físico es responsable del transporte de los bits y el nivel de enlace se ocupa del control de congestión y multiplexado estadístico de conexiones. En este nivel, el encargado de las tareas de señalización es el protocolo LAP-D. El control de errores se debe efectuar en las capas superiores.

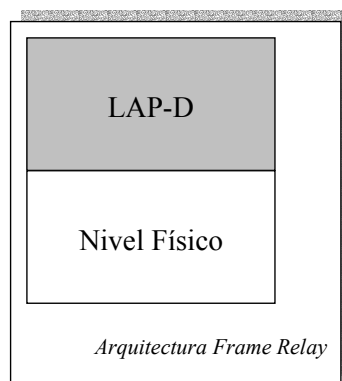


Fig. 5.10 Torre de protocolos de Frame Relay

En cuanto al direccionamiento, necesario para poder establecer los circuitos virtuales, en el nivel de enlace se definen direcciones de 2 bytes, de modo que cada nodo viene identificado por un SAPI

(*Service Access Point Identifier*) o Identificador de Punto de Acceso al Servicio. Además. Cada terminal dispone de un identificador o TEI (*Terminal Endpoint Identifier*). De este modo es posible multiplexar diversos circuitos virtuales desde un mismo nodo. El TEI y el SAPI se denominan conjuntamente como el identificador de conexión del nivel de enlace, que es precisamente el DLCI.

### 5.2.5 Acceso a una red Frame Relay

La forma tradicional de acceder a una red Frame Relay es mediante el dispositivo FRAD (*Frame Relay Access Device*), cuyo nombre rememora el PAD (*Packet Assembler / Disassembler*) de las antiguas redes X.25. Como puede verse en la figura 5.11, la conexión se efectúa mediante el dispositivo FRAD conectado a la red Frame Relay que se encarga del encapsulado y desencapsulado de los paquetes hacia o desde la red.

De este modo resulta posible la conexión de una red LAN hacia una red de transporte Frame Relay contratada a un operador de modo que una dos redes de área local distantes. Tal puede ser el caso de dos departamentos, dos oficinas, etc.

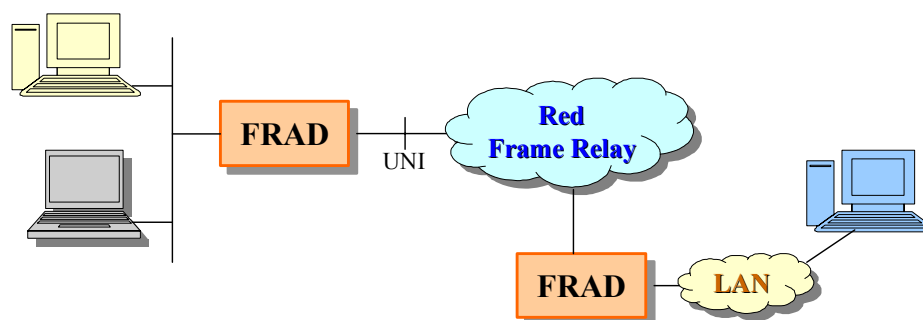


Fig. 5.11 Esquema tradicional de acceso a una red Frame Relay desde una red LAN

### 5.2.6 El nivel físico

De acuerdo a la recomendación J.122 del ITU-T, se puede proporcionar servicio de conmutación de paquetes sobre RDSI, empleando para ello canales B (64 Kbit/s), canales D (de 16 ó 64 Kbit/s) y canales H ( $H_0$  de 384 Kbit/s ó  $H_1$  de 2048 Kbit/s).

El ITU-T indica que se pueden emplear accesos a Frame Relay mediante interfases V.35 para servicio a 56 ó 64 Kbit/s, G.703 para servicio a 2 Mbit/s, y también interfases de X.21.

El Frame Relay Forum, por su parte, refiere a ANSI T1.403 para medios a 1,5 Mbit/s.

### 5.2.7 El nivel de enlace

Tal como muestra la figura, la trama se inicia con un flag de inicio y final, que permite determinar la longitud del campo de información. Este flag sigue una secuencia reservada cuyo valor es 01111110, y que el sistema tiene reservada mediante un mecanismo que garantiza que no exista ningún dato que pueda usar esta misma combinación. El campo de CRC se emplea como herramienta de detección de errores. Por lo que hace referencia al campo de cabecera o header, se descompone en diversos campos.

El DLCI ya ha sido comentado a raíz del procedimiento de definición del identificador de conexión en la sección dedicada a la conmutación.

El campo de Petición / Respuesta (C/R) se emplea para conocer si una trama es una petición o bien se manda como respuesta a una solicitud. Es útil para el caso de estudiar o reconocer los diversos flujos de información asociados a una conversación entre dos equipos.

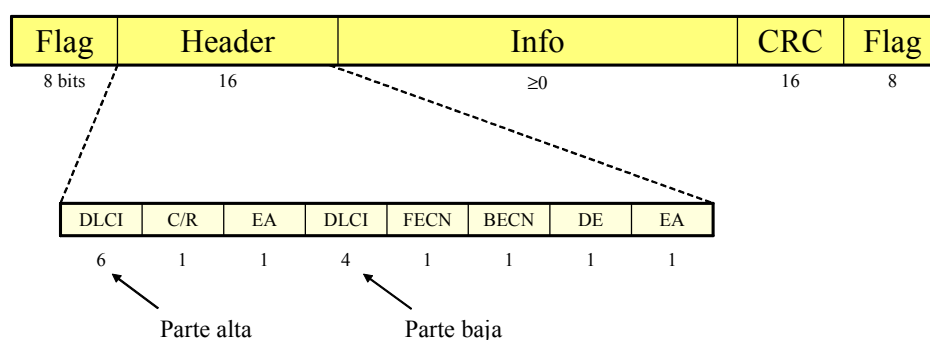


Fig. 5.12 Formato de trama de Frame Relay

El campo EA o de Extensión de Dirección (Extended Address) permite definir campos de cabecera de mayor longitud con el fin de poder asignar más bits para DLCI, si esto fuera necesario. Cuando EA está fijado a '0' significa que el campo continúa. Con un '1' que acaba. La configuración más larga permitida autoriza hasta un máximo de 4 bytes, de modo que quedaría del modo que aparece en la figura 5.13.

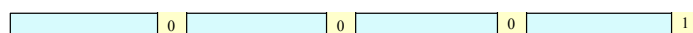


Fig. 5.13 Configuración más larga posible del campo de cabecera, atendiendo al campo EA

El campo DE (*Discart Eligibility*) o Elegible para Descarte, permite al usuario o a la red marcar tramas como candidatas para ser descartadas, caso de que el estado del sistema lo requiriera. En el mecanismo de control de congestión se estudiará como afecta este campo. También debe mencionarse

que aunque Frame Relay no disponga de mecanismos de prioridad, es bien cierto que gracias a este bit es posible definir dos niveles: El nivel bajo, caracterizado por tener este bit activado, y el nivel alto, en que no lo está.

De este modo, los operadores de red pueden ofrecer dos niveles de prioridad, gracias a este procedimiento ingenioso.

Finalmente, quedan por mencionar los bits FECN y BECN (*Forward, Backward Explicit Congestion Notification*). Son bits asociados a los niveles de congestión de la red, y permiten indicar a los terminales si existe congestión en algún punto del circuito contratado.

Su funcionamiento queda más allá de los objetivos de este libro, aunque puede consultarse la bibliografía para obtener más detalles al respecto.

### 5.2.8 Parámetros de contrato

Frame Relay define un conjunto de parámetros, que definen el perfil de tráfico de las diversas conexiones contratadas. Los parámetros principales son los siguientes: CIR,  $B_c$ ,  $B_e$  y  $T_c$ .

1. El CIR (*Committed Information Rate*), es el caudal cursable en la conexión, promediado en unidades de tiempo  $T_c$ .
2. El  $B_c$  (*Committed Burst Rate*) es el máximo volumen de bits que el usuario puede transmitir por la conexión en un intervalo  $T_c$ . Se desprende de estas dos definiciones que  $CIR = B_c / T_c$ .
3. EL  $B_e$  (*Excess Burst Rate*) es el máximo volumen de bits que el usuario puede transmitir por la conexión excediendo a  $B_c$  en un intervalo  $T_c$ . A partir de este valor, se define el EIR (*Excess Information Rate*) o tasa de información en exceso, como  $EIR = B_e / T_c$ .

Teniendo en cuenta estos parámetros, la tabla de la siguiente figura muestra diversas situaciones:

CIR	$B_c$	$B_e$	
> 0	> 0	= 0	$T_c = B_c / CIR$
> 0	> 0	> 0	$T_c = B_c / CIR$ ; $EIR = B_e / T_c$
= 0	= 0	> 0	$T_c = B_e / \text{tasa acceso en el enlace}$

Fig. 5.14 Diversos contratos de conexión Frame Relay

En el segundo caso ( $CIR > 0$ ,  $B_c > 0$ ,  $B_e = 0$ ), se tiene una conexión en la cual se dispone de un caudal CIR y no hay posibilidad de transmitir un exceso de tráfico por encima de este CIR. El valor de  $T_c$  puede obtenerse simplemente considerando cada qué volumen de bits  $B_c$  se desea controlar el caudal.

En el segundo caso ( $CIR > 0$ ,  $B_c > 0$ ,  $B_e > 0$ ), se tiene una típica conexión en la cual se dispone de un CIR y también de la posibilidad de transmitir un exceso de tráfico por encima de este CIR. El valor de  $T_c$  puede obtenerse al igual que en el caso anterior, considerando cada qué volumen de bits  $B_c$  se desea controlar el caudal. En este caso, se podrá obtener el valor de la tasa EIR como cociente entre  $B_e$  y  $T_c$ .

El último caso resulta cuanto menos, curioso, puesto que el valor de CIR es nulo. Sin embargo, se permite la transmisión de caudal gracias a  $B_e$ .

En esta situación, como se verá en la siguiente sección dedicada al control de congestión, todas las tramas entrarán en la red marcadas como DE por estar por encima de CIR. Este tipo de contratos constituyen conexiones de baja prioridad. No por ello dejan de ser útiles, porque existen usuarios dispuestos a emplear servicios económicos para funciones de red que no precisan de ningún tipo de caudal garantizado. El valor de  $T_c$  se contabiliza como  $B_e$  sobre el valor de la tasa de acceso del enlace que llegue hasta el punto de acceso del abonado.

Por ejemplo, puede ser un enlace a 1 Mbit/s, y nótese que aunque el usuario deseara inyectar este caudal, al llegar al punto de entrada a la red, sería presumiblemente descartado por el operador.

### 5.2.9 Control de congestión

Ante todo, es preciso definir la congestión. Es usual comprobar como erróneamente se considera la congestión como el exceso de tramas en la red.

Resulta fundamental comprender que esto es la causa de la congestión, no la congestión como tal. De hecho, un usuario seguro que admitiría sin queja alguna una conexión que garantice perfectamente el caudal que haya contratado, sea cual sea el estado de la red. El problema aparecería cuando el servicio recibido no se ajustase al esperado.

Desde este punto de vista, puede definirse la congestión como aquella situación en la cual la red no es capaz de ofrecer la calidad de servicio que los usuarios han contratado. Esto es, no es capaz de garantizar un determinado retardo ni caudal, que los usuarios han contratado.

La causa es efectivamente, al margen de disfunciones en equipos, el exceso de tramas generadas.

Para garantizar el correcto funcionamiento de la red, es preciso controlar que los usuarios generen el caudal que han contratado.

Para comprender el funcionamiento empleado en Frame Relay para controlar la congestión, consideremos la figura 5.15 donde un usuario genera un determinado caudal, representado por la línea más gruesa.

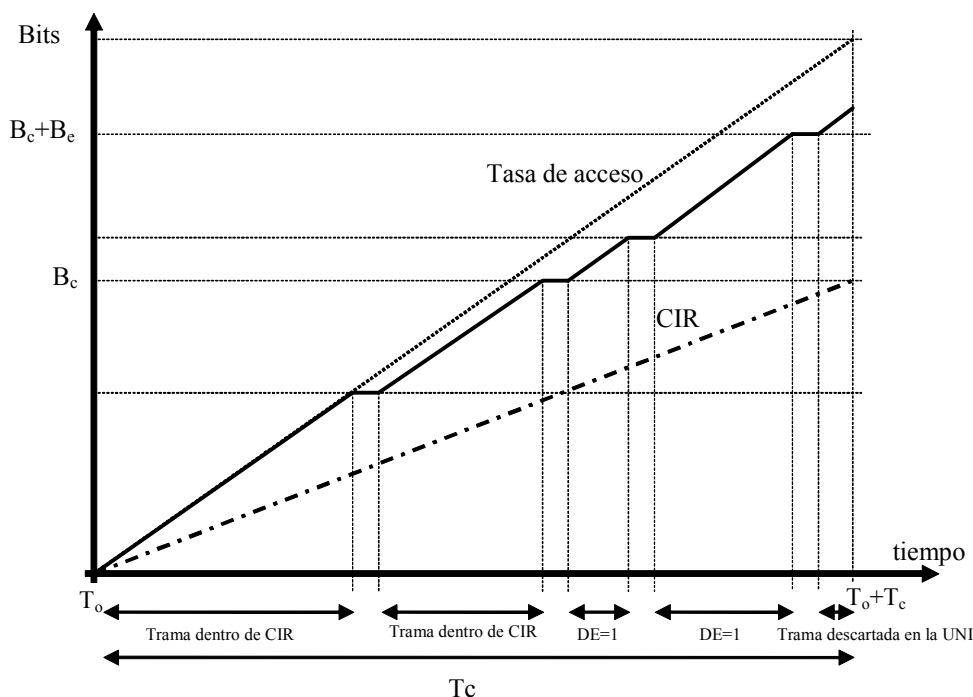


Fig. 5.15 Ejemplo de funcionamiento del control de congestión

Obsérvese que existe una línea gruesa representada con punto-ralla, que se inicia en el eje de coordenadas, a tiempo  $T_0$  y ningún bit en el eje de ordenadas, y llega hasta  $T_0 + T_c$ , con  $B_c$  bits. Por lo tanto, la pendiente de esta recta es  $B_c / T_c$ , es decir presenta pendiente CIR.

Por tanto, todas las líneas de mayor pendiente a esta recta serán tasas superiores, y las de menor pendiente de caudal inferior.

En consecuencia, la línea de trazos discontinuos que nace donde cruzan los ejes presenta una tasa bastante superior. Es el caudal del enlace, que puede ser muy superior a la tasa contratada y por tanto permitida por la red.

Si analizamos el comportamiento de una hipotética fuente (trazo continuo más grueso), se observa que inicia la transmisión generando un determinado volumen de información inferior a  $B_c$ , a velocidad igual a la del enlace. Por lo tanto, esta primera trama no supera ningún umbral del eje de ordenadas, y debe considerarse como trama aceptable.

Tras una breve pausa, la fuente reemprende la transmisión, siguiendo con la tasa máxima, que es la del enlace, y alcanza un total de bits acumulado de  $B_c$ . Dado que no se ha superado este umbral, la trama se considera también perfectamente aceptable, dentro de CIR.

Tras otra pausa, se genera una nueva trama, que hace que el total de bits generados supere a  $B_c$  pero no a  $B_c + B_e$ . Por lo tanto, la trama se acepta pero se marca su campo DE a 1. De este simple modo, se indica que puede ser descartada en caso de que se produzca congestión o los conmutadores lo consideren. El mismo comentario puede ser aplicado a la siguiente trama, que no llega a superar el

segundo umbral  $B_c+B_e$ . Finalmente, la figura muestra como antes de haber transcurrido  $T_c$ , se genera una nueva trama, que desborda el umbral superior. En consecuencia, esta trama será descartada.

La principal consecuencia de este procedimiento de control es que la red permite el paso de ráfagas de tráfico, de acuerdo siempre a no superar el segundo umbral. Este hecho podrá observarse con un refinamiento superior en las redes de banda ancha basadas en la tecnología ATM, que serán objeto de estudio en el siguiente capítulo.

## 6. La red de banda ancha

### 6.1 Introducción

La tecnología ATM es la base de la Red Digital de Servicios de Banda Ancha (*Broadband Integrated Services Digital Network, B-ISDN*).

En la definición de Banda Ancha de la recomendación I.113 del ITU-T, se afirma que es un servicio o sistema que necesite canales de transmisión capaces de soportar tasas superiores a la primaria (es decir, mayor a 2 Mbps). Esta definición es hoy en día obsoleta, dado el salto que en los últimos se ha producido en las redes de comunicaciones, alcanzando cotas cercanas a 1 Tbit/s, bajo tecnologías tales como DWDM (*Dense Wavelength Division Multiplexing*), que quedan fuera del alcance de este libro.

#### 6.1.1 Problemática

La aparición de la Red Digital de Servicios Integrados de Banda Ancha se explica por la existencia en su momento de una red de banda estrecha (la popular RDSI comentada en el capítulo 5), que ofrece poca flexibilidad y poco ancho de banda, puesto que se tratan de caudales de tasa constante, sin posibilidad de variaciones de tráfico.

Las necesidades actuales se dirigen hacia un gran ancho de banda, con dispositivos de conmutación y transporte rápidos, basados en canales a alta velocidad, que admita servicios orientados y no orientados a conexión, y comunicación punto a punto y multipunto.

#### 6.1.2 Caracterización de los servicios

El paradigma actual establece que es precisa una clasificación de los servicios con el fin de adecuar el perfil demandado por los usuarios con la oferta que hagan los operadores de red. Resulta utópico pensar que los servicios ofrecidos por las empresas de servicios de red coincidan con la demanda de los usuarios, pero es cierto que es posible acercar ambas posturas conociendo el comportamiento de las fuentes de usuario y sus necesidades, con el fin de conseguir multiplexar la información adecuadamente.

De este modo, se puede establecer la siguiente clasificación de servicios:

1. Conversacionales: aquellos que son bidireccionales y operan con requisitos de tiempo real. Por ejemplo, los servicios de telefonía.
2. Mensajería: los que se efectúan a través de almacenamiento y sin requisitos temporales. Un ejemplo se encuentra en los buzones de correo electrónico.
3. Servicios de recuperación: aquellos que permiten la obtención de información almacenada, ya sea con o sin requisitos temporales. Por ejemplo, el vídeo bajo demanda.

De acuerdo a las variaciones de caudal que pueda experimentar la fuente de datos, se define la *relación de rafagueo* o *burstiness* como la fluctuación del ancho de banda respecto el valor medio. Así pues, un valor de burstiness = 1 se interpreta como un caudal constante. Por otro lado, un burstiness  $\gg 1$  supone que el ancho de banda de la transmisión es muy variable, presentando picos de tráfico muy pronunciados. Este último caso es difícil gestión en las redes tradicionales.

Un ejemplos de burstiness elevado es el vídeo comprimido en modo de tasa variable (es habitual también la compresión a tasa constante), como por ejemplo, en los estándares MPEG 1, 2 o 4.

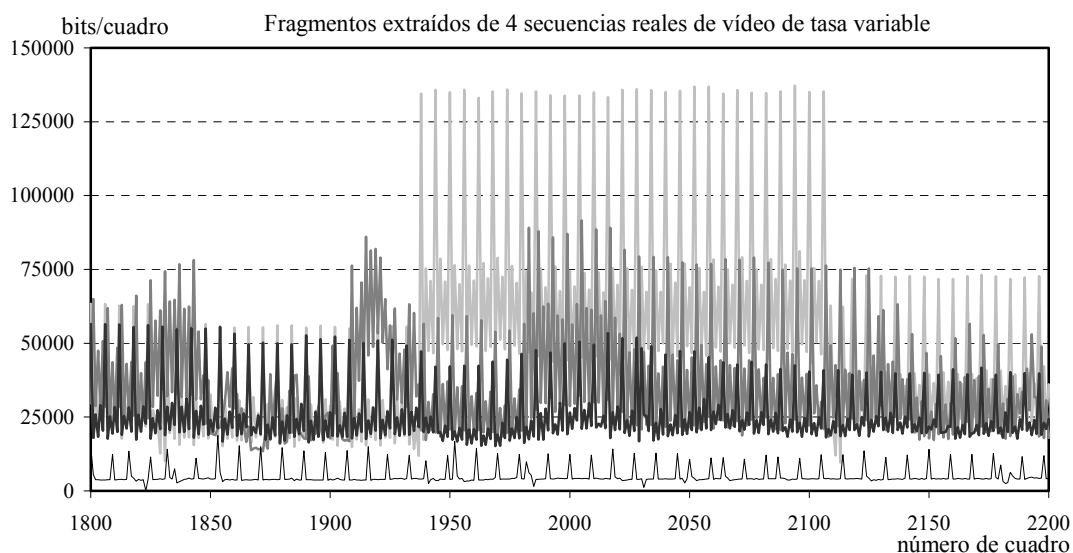


Fig. 6.1 Trazas de vídeo de formato MPEG-1

La figura 6.1 muestra diversas capturas de tráfico generado por diversas fuentes de vídeo reales de formato MPEG-1. Además de los usuales picos debidos al entramado MPEG con formatos I,P y B. puede observarse que el valor medio varía. Ello es debido a pasajes de la secuencia con poco movimiento (tales como puestas de sol) y otras con un caudal más elevado (como podría ser el caso de una escena con mucho movimiento, como una persecución de vehículos).

## 6.2 Técnicas de multiplexado para la B-ISDN

La multiplexación de datos en B-ISDN se lleva a cabo en la actualidad basándose en tecnología *ATM* (*modo de transferencia asíncrono*), aunque inicialmente se descartó la solución *STM* o *modo de transferencia síncrono*.

Síncrona: STM (Synchronous Transfer Mode)

Asíncrona :ATM (Asynchronous Transfer Mode)

En STM se consideró un multiplexación basada en TDM, con canales separados y un ancho de banda de cada canal fijo, sin información de propiedad del paquete. Este esquema es el mismo que el empleado en la RDSI estudiada en el capítulo 5, y por tanto es adecuado para fuentes de tasa constante pero inadecuado para tasa variable.

La figura 6.2 muestra este esquema de funcionamiento.

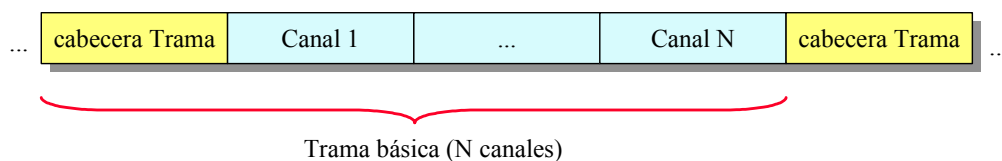


Fig. 6.2 Trama STM

Por otro lado, ATM propone una asincronía en el sentido de no hay canales reservados del tipo STM, aunque el medio perfectamente síncrono (existen señales de reloj). El acceso de los usuarios al medio se efectúa a instantes determinados de tiempo, en ranuras específicas, pero sin existir ninguna ranura reservada. Por este motivo se denomina *asíncrono*.

La unidad de datos definida en ATM es un bloque de longitud fija de  $5 + 48 = 53$  bytes, denominada *celda* ATM. Este tamaño facilita la flexibilidad para el transporte de todo tipo de datos y resulta cómodo para los dispositivos de conmutación, con una cabecera muy simple para simplificar las tareas.

La figura 6.3 presenta un conjunto de celdas, usadas por diversos usuarios arbitrarios.

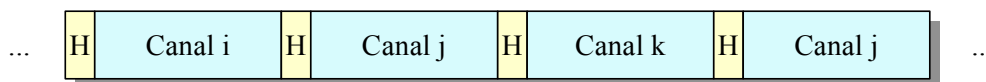


Fig. 6.3 Trama ATM

Las ventajas de ATM se resumen en:

Permite una asignación dinámica del ancho de banda, mediante el empleo consecutivo de varias celdas para conseguir picos de tráfico para atender a ráfagas de usuario.

Facilita la flexibilidad a la hora de repartir los recursos.

### 6.3 Modelo de protocolos de la B-ISDN

El modelo de protocolos de la B-ISDN viene expuesto en la figura 6.4. Aunque el detalle de todos los planos que conforman esta arquitectura queda lejos de los propósitos de este libro, se mencionan a continuación los aspectos principales.

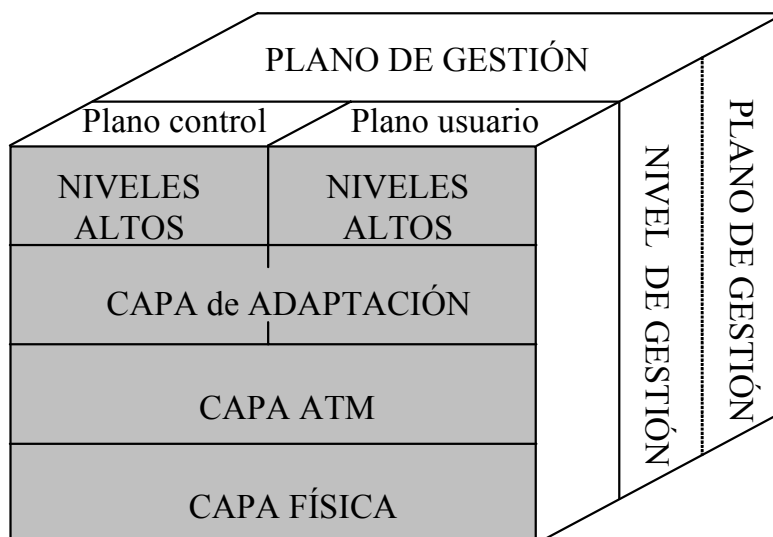


Fig. 6.4 Modelo de protocolos de la Red Digital de Servicios Integrados de Banda Ancha

El plano de usuario es responsable del control de flujo y la recuperación de errores.

El plano de control está encargado del control de la llamada y la conexión. Por lo tanto, del establecimiento, control y liberación de los circuitos que se establezcan.

El plano de gestión está dedicado a la gestión de todo el sistema. No presenta ninguna estructura de capas.

## 6.4 División jerárquica ATM

ATM establece una división jerárquica dentro de su funcionamiento de transmisión de información. Esta división distingue la capa ATM, situada en el nivel 2 o nivel ATM, tal como se observa en la figura 6.4, y del nivel físico.

En relación a la capa ATM, se establece un nivel de VC (Circuito Virtual), que presenta un identificador para un conjunto de celdas, denominado VCI o identificador de circuito virtual. En él, los canales son unidireccionales. Por lo tanto, para establecer una comunicación bidireccional (tal como una llamada telefónica) es preciso establecer dos circuitos.

En esta misma capa ATM existe también el denominado nivel de VP o de camino virtual, que define un identificador para un conjunto de circuitos virtuales VC. Este grupo de VC tienen en común el mismo VPI (Identificador de VP).

En el nivel físico, existe un *camino de transmisión* que se conforma por los flujos de celdas, una *sección digital*, por bytes y bits, y una *sección de regeneración*, caracterizado por el estado del enlace.

## 6.5 Transporte y conmutación

Hasta el momento se ha explicado que ATM define dos niveles para el transporte de celdas: El nivel de VC, y el de agrupaciones de VC, que se denominan VP. La conmutación en cada caso es diferente.

En el caso de la conmutación mediante el valor del VPI transportado en las celdas, la situación es análoga a la que se efectúa en las redes Frame Relay, donde debe entenderse que en este caso el papel del DLCI es jugado por el VPI. En la figura 6.5 se muestra un posible esquema, y fácilmente se pueden ver las analogías entre ambas tecnologías.

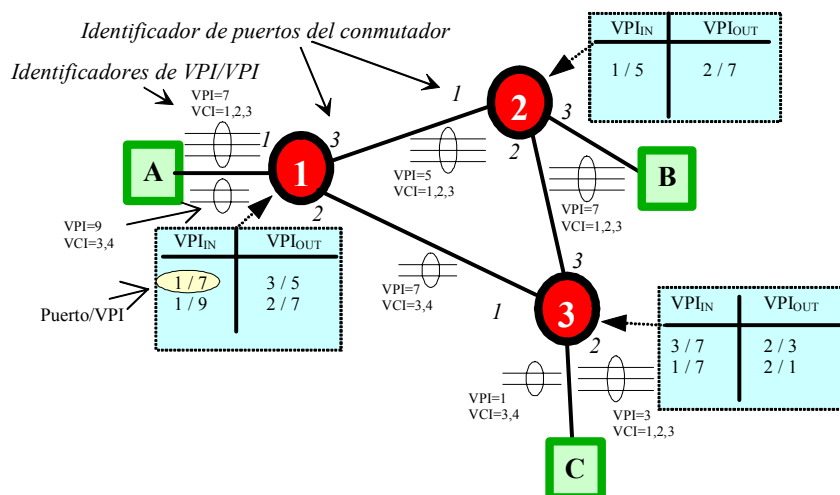


Fig. 6.5 Ejemplo de tablas de conmutación de VPI

En el caso de la conmutación por VCI, es preciso comprender ante todo que la conmutación basada en VCI es posterior a la de VPI, puesto que un conjunto de VCI ha sido agrupado dentro de un VPI. Consecuentemente, el conmutador se encarga primero de deshacer el haz de VPI para dejar disponible la posibilidad de conmutar los VC, mediante una tabla específica.

Para ello, como aparece en la figura 6.6, el conmutador puede decidir generar nuevos VPI para poder hacer llegar hasta sus destinos las celdas de datos que quizá hasta el momento habían viajado junto a otras bajo el mismo identificador VPI.

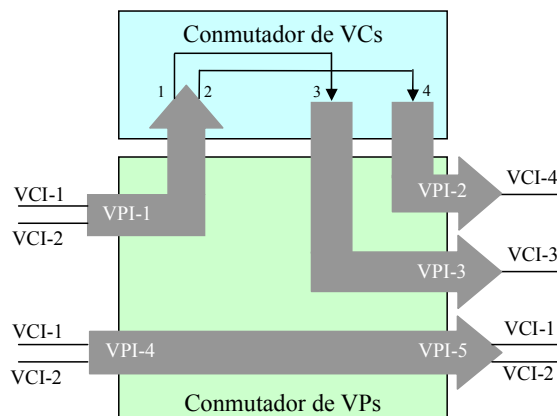


Fig. 6.6 Ejemplo de conmutación de VPI y VCI

La ventaja de emplear un esquema con VCI y VPI estriba en que mientras sea posible emplear el valor de VPI para la conmutación, supone un esfuerzo menor para los equipos, pudiendo por tanto conmutar un volumen de información mayor.

## 6.6 Establecimiento de conexiones

Las redes ATM distinguen dos tipos de circuitos, en función de la naturaleza de su establecimiento:

Los circuitos virtuales permanentes (CVP) y los circuitos virtuales conmutados (CVC). Los primeros son aquellos que se establecen manualmente, sin necesidad de un proceso de señalización. Los segundos se contratan mediante un intercambio de mensajes de señalización, los cuales identifican el equipo ATM destino y las características de la conexión a establecer, mediante el denominado descriptor de tráfico, tal como más adelante se definirá.

En consecuencia, los CVP son establecidos mediante un operador que se encarga de fijarlos manualmente, tanto en ruta como en características del caudal.

Los CVC se establecen o se cierran automáticamente mediante mensajes de señalización, en los cuales se indica el destino y las características del tráfico a contratar. En este caso, es necesario poder referir el equipo ATM destino de algún modo.

Por este motivo, se emplean las direcciones ATM, de 160 bits, con estructura jerárquica. Existen diversos formatos, distinguibles gracias al campo AFI (Authority and Format Identifier). La figura 6.7 muestra los formatos definidos para las direcciones ATM.

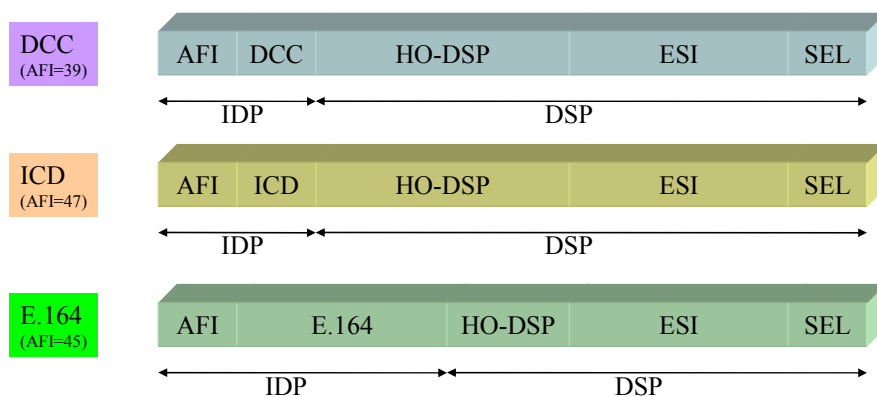


Fig. 6.7 Formatos de direcciones ATM

Los campos más destacados de entre los mostrados son los pertenecientes a la identificación del equipo final, que son el ESI (End System Identifier) y SEL (Selector). El resto, corresponden a la jerarquización de la red y se asocian al conmutador con el que el equipo está conectado.

IDP (Inicial Domain Part):

AFI (Authority and formato Identifier): Identificador de tipo de dirección

IDI (Initial Domain Identifier): 2 bytes si DCC ó ICD

DSP (Domain Specific Part):

HO-DSP (High-Order DSP): Por construcción de jerarquías de direcciones

ESI (End System Identifier): 48 bits para la identificación específica de un host ATM

SEL (8 bits): Selector. Identifica la interfaz virtual específico de comunicación

Las direcciones ATM motivan la necesidad de disponer de algún dispositivo de registro de dichas direcciones de modo de pueda ser consultado por los equipos de la red, y obtener direcciones de forma automática o interrogar sobre cuestiones específicas. Este dispositivo se conoce por ILMI (*Interim Local Management Interface*).

La interacción del equipo final con el ILMI es la siguiente:

ILMI permite al switch conocer todos los hosts que tiene conectados a través del VPI=0 y VCI=16:

El conmutador les envía su ESI y SEL y espera una conformación del equipo solicitante con la dirección completa. Una vez entregada la dirección ya es posible una solicitud de establecimiento de circuito CVC, para lo cual los conmutadores usan el protocolo PNNI que permite la localización de una ruta hasta el equipo destino deseado.

Si no hubiera un ILMI, es necesaria la configuración manual mediante la intervención directa del usuario u operador de red.

## 6.7 Interfases y formatos de celda

La figura 6.8 muestra el formato de celda definido para redes ATM. El formato es ligeramente diferente en función de si se trata de una celda que entra a la red ATM (para lo cual emplea la interfaz UNI, *user-network Interface*), o bien si es interior a la red, para lo cual el formato es el de NNI (*node-node interface*).

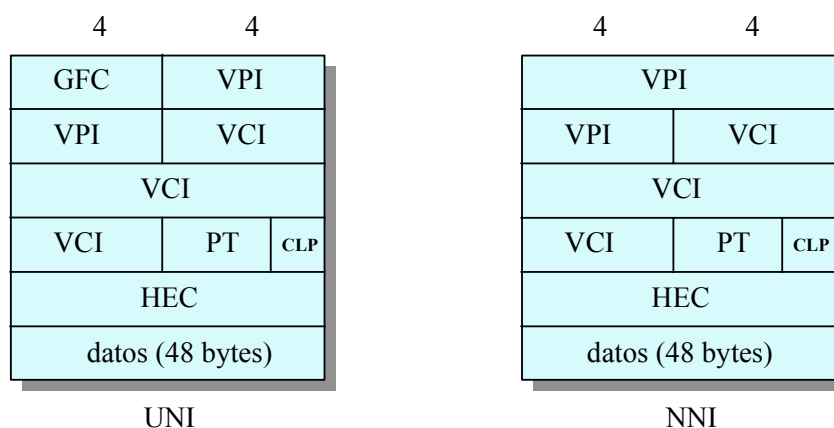


Fig. 6.8 Formatos de celda en interfases UNI y NNI

Los campos son los siguientes:

- El campo GFC o *Generic Flow Control*, es un campo previsto para incorporar mecanismos de control de flujo, y del cual se han estudiado protocolos para que pueda diseñarse una red de acceso ATM. Sin embargo, en la práctica, no se usa.
- El VPI es un campo de 8 / 12 bits (según si se trata de la UNI / NNI, como se muestra en la figura 6.9), para identificar el identificador de camino virtual. El valor de VPI es mayor en la NNI, es decir dentro la red, con el fin de poder dar acomodo a un mayor número de conexiones. Nótese que el incremento se produce por la desaparición del GFC, que en el interior de la red no tendría uso.
- El campo VCI permite identificar con 16 bits el valor de circuito virtual asociado a una conexión.

- PT: Define varios tipos de carga, como muestra la tabla 6.1. Por ejemplo ,destaca el tipo 110, que son celdas denominas RM o de gestión de recursos, cuya importancia será puesta de relieve más adelante al abordar el estudio de unos servicios denominados ABR o de tasa disponible.
- CLP: Cell Loss Priority: Un valor '0' denota alta prioridad, un '1', baja prioridad. Su función es prácticamente la misma que el bit DE de las redes Frame Relay.
- HEC: El Header Error Control es simplemente un código de control de errores de la celda.

Tabla. 6.1 Interpretación de los valores del campo PT según el ATM Forum

PT	Interpretación ATM Forum
000	datos usuario tipos 0, sin congestión
001	datos usuario tipos 1, sin congestión
010	datos usuario tipos 0, con congestión
011	datos usuario tipos 1, con congestión
100	celda OAM (Gestión)
101	celda OAM (Gestión)
110	Celdas Resource Management (RM)
111	Reservado

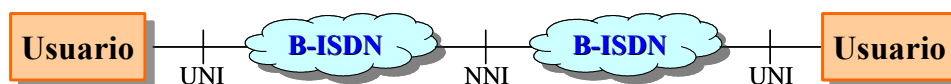


Fig. 6.9 Tipos de interfaces en B-ISDN

## 6.8 La capa de adaptación

Hasta el momento, se ha estudiado que ATM proporciona un transporte de celdas ATM, que son unas unidades de datos de pequeño tamaño, y que la conmutación se efectúa mediante los identificadores de VCI-VPI y unas tablas de conmutación que se han programado en cada nodo durante la fase de establecimiento de conexión.

Ahora bien, las celdas ATM son demasiado pequeñas para cubrir las necesidades de los usuarios, puesto que en general los tamaños de trama son grandes (por ejemplo, los paquetes IP, y además, debe organizarse el acceso a las celdas dentro del flujo ATM, tal como aparece en la figura 6.3).

Por este motivo, se requiere una capa situada justo por encima de ATM, que se beneficie de este transporte de celdas asíncrono, pero que permita mejorar el servicio de la capa ATM, mediante un conjunto de servicios específicos, entre los cuales destacan los siguientes:

Segmentación y reensamblado de la información de capas superiores, para dejarlo en tamaño ubicable en las celdas.

En ciertos casos, recuperación de reloj, usando mecanismos de etiquetado de paquetes como SRTS (*Synchronous Residual Time Stamp*), en especial para servicios donde se requieran garantías de tiempo real y el receptor deba conocer los tiempos de generación de las celdas en el emisor.

Recuperación de errores

Se denomina capa de adaptación (AAL, *ATM Adaptation Layer*) a esta capa. Se define en ella 5 tipos o modos de operación, denominados:

AAL1, 2, 3/4 y 5

Existe también la llamada AAL0, que consiste en el uso directo de celdas ATM, sin disponer del beneficio de los servicios específicos antes mencionados como la segmentación.

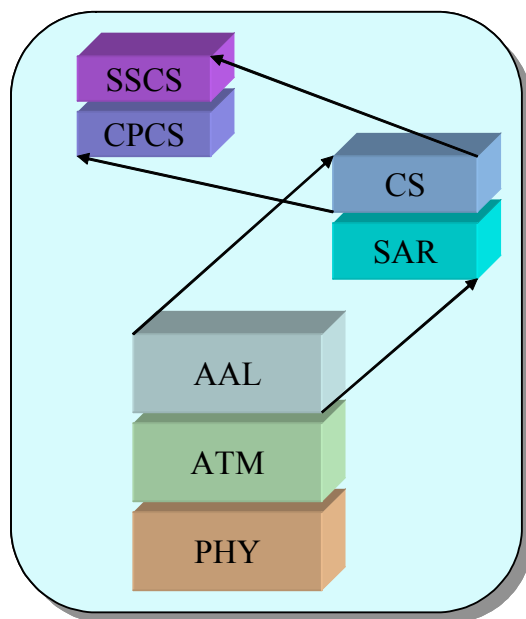


Fig. 6.10 Esquema de subcapas de protocolos de la AAL en ATM

Esta capa AAL define ciertas subcapas, cuyo estudio queda fuera del alcance de este libro. Estas subcapas son:

CS, Convergence Sublayer.

SSCS, Service Specific CS

CPCS, Common Part CS

SAR, Segmentation and Reassembly

La figura 6.10 muestra gráficamente como se articulan todas ellas. A efecto de comprender los servicios, basta con tener presente cuáles de ellas son soportadas en los equipos que se adquieran, ya que cada una define unas propiedades específicas de funcionamiento, como a continuación se expondrá.

### 6.8.1 AAL1

La AAL1 está definida en la recomendación I.363.1 para establecer un transporte de tasa constante o CBR (*constant bit rate*), con transmisión de información de reloj SRTS, e información de error o pérdida. El formato de trama a nivel de encapsulado en la celda ATM es el de la figura 6.11, donde se observa que de los 48 bytes de datos de la celda, 1 se emplea en el soporte a AAL1, lo cual significa pérdida de capacidad de transporte de datos de usuario.

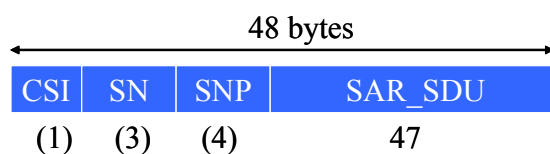


Fig. 6.11 Formato de trama de AAL1

La capa SAR de AAL1 presenta el campo CSI, como indicador de existencia capa CS activa, SN para numerar la secuencia de celdas, y un campo SNP para protección.

La capa CS es la que se encarga del SRTS, del control celdas perdidas o mal insertadas (es decir, las que llegan a un destino incorrecto) y a la corrección y monitorización de los campos de usuario.

### 6.8.2 AAL2

La AAL2 está definida en la recomendación I.363.2 para realizar transmisión con uso eficiente del ancho de banda, es decir, con beneficio de multiplexado estadístico de diversos usuarios. Además, se aplica para aplicaciones sensibles al retardo, puesto que incorpora mecanismos para el soporte de

transporte en tiempo real. Una aplicación típica de AAL2 es el VBR (*Variable Bit Rate*) de vídeo o audio.

El formato se muestra en la figura 6.12, donde se aprecia una sensible pérdida de bits útiles, con el fin de poder dar soporte a los servicios de esta capa de adaptación.

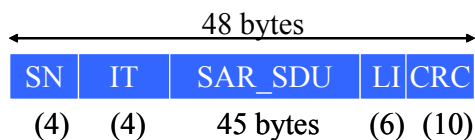


Fig. 6.12 Formato de trama de AAL2

En la capa SAR, se establece el campo SN para el número de secuencia, el IT con un indicador de BOM, COM, EOM (*Begin of Message, Continuation of Message, End of Message*, para informar de si esa celda es de inicio, intermedia o de fin de conexión) o sincronismo.

LI es un indicador de la longitud del campo de datos, y finalmente el CRC, que se emplea para detección y corrección.

En la capa CS se establecen los valores para el sincronismo, se efectúan las tareas para la posible recuperación de celdas perdidas, y se calcula un valor que se aplica para la corrección de errores en servicios multimedia.

### 6.8.3 AAL3/4

La AAL3/4 está definida en la recomendación I.363.3, ofreciendo un servicio de tasa variable. Inicialmente existió una AAL3 y AAL4, pero se decidió unificar en una única.

Admite servicios tanto no orientados a conexión como los que si lo son, pero sin dar soporte a sincronismo fuente-destino, ni contemplar ninguna sensibilidad a retardo.

Es por ello, que AAL3/4 está pensado especialmente para datos.

Adicionalmente permite sobre un mismo VCI/VPI efectuar un multiplexado usando el campo MID (*Message Identifier*), de 10 bits, heredero directo de una tecnología ya caduca denominada IEEE 802.6 MAN (DQDB, *Distributed Queue Dual Bus*).

La figura 6.13 representa el formato de este entramado.

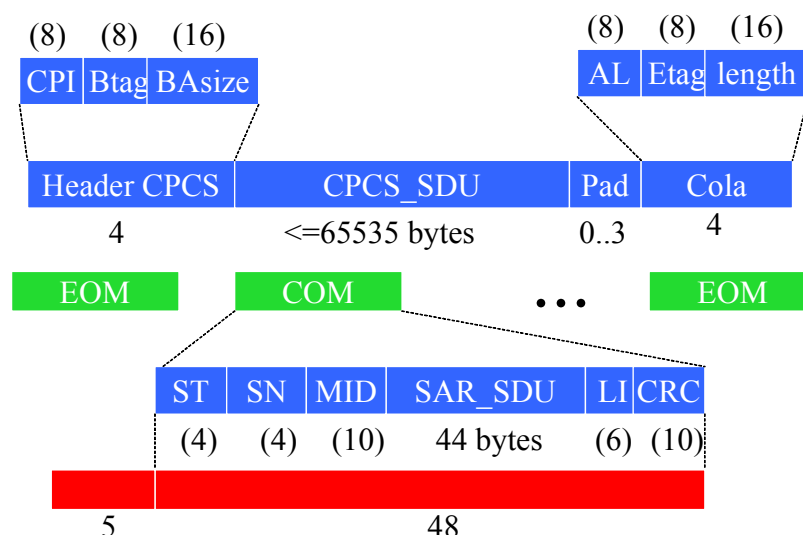


Fig. 6.13 Formato de trama de AAL3/4

En la capa SAR se define el campo CPI con el identificador de tipo de PDU, un campo Al de alineamiento a 32 bits, ST indicando el Tipo de Segmento (BOM, COM, EOM, SSM, como en AAL2, donde SSM es *Single Segment Message*, para aquellos casos en que basta con el espacio de una sola celda para transmitir todo el mensaje), y el campo MID ya mencionado.

En la capa CS, la longitud datos puede ser de hasta 65535 bytes, que son inmediatamente segmentados para poder ser transmitidos en celdas ATM.

#### 6.8.4 AAL5

La AAL5 está definida en la recomendación I.363.3, como una simplificación de AAL 3/4 dado el alto grado de complejidad que esta última presenta. Con este fin, se han reducido el número de campos aunque sea a costa de disponer de menos funcionalidades. Ahora bien, la idea es dejar 48 bytes disponibles para usuario.

En este momento, surge la duda de qué diferencia presenta entonces AAL5 con AAL0. La respuesta es que AAL5 proporciona mecanismos sutiles como la segmentación, cosa que en AAL0 debe ser gestionada por el propio usuario.

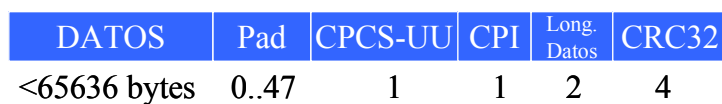


Fig. 6.14 Formato de trama de AAL5

El formato de la AAL5 es el de la figura 6.14. La descripción de los campos queda fuera del ámbito de esta obra, aunque puede decirse que en general es para gestión del sistema o información de las capas superiores a AAL.

## 6.9 Gestión de tráfico

### 6.9.1 Introducción

Como anteriormente se ha mencionado, el usuario accede a la red de banda ancha a través de la interfaz UNI. El usuario debe contratar una conexión, lo cual significa que la red debe establecer una ruta, donde lo más importante es que esa ruta debe satisfacer unos determinados parámetros de calidad, tales como caudal o retardo. Por ejemplo, un equipo puede solicitar a la red una tasa de 3 Mbit/s, con un retardo de 10 mseg, o también podría añadir en la petición la posibilidad de transmitir picos de 100 Mbit/s con una duración máxima de 1 mseg.

Todas estas formas de solicitar a la red una *calidad de servicio* (*Quality of Service, QoS*) por medio de unos determinados parámetros, será estudiado en esta sección.

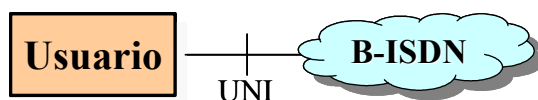


Fig. 6.15 Interfaz UNI entre el usuario y la red

En cualquier caso, la red dispone de unos mecanismos de control que deben servir para efectuar el control de los recursos que el usuario va a emplear. Entre estos controles, los más importantes son los siguientes:

CAC o Control de Admisión de la conexión

Control de Prioridad, efectuado mediante el bit CLP de la celda ATM

UPC (Control del uso de parámetros), también denominada *función de policía*

Control de congestión

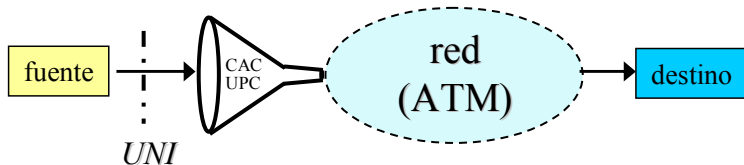


Fig. 6.16 Funciones de control sobre el tráfico de usuario en la UNI

La figura 6.16 los ilustra, donde el embudo trata de ser una imagen alegórica de la función de restricción que la red tiene sobre el caudal que el usuario genere. Si el volumen de datos supera un cierto umbral, el embudo desbordará. Sin embargo, admite una cierta tolerancia o ráfagas.

### 6.9.2 Definición de parámetros

Para poder comprender como se contrata una determinada calidad de servicio en una red ATM, deben definirse los parámetros que se emplean en esta tecnología con esta fin.

Los parámetros de tráfico son los siguientes:

PCR, Peak Cell Rate, o tasa de pico, medida en celdas por segundo.

SCR, Sustainable Cell Rate, o tasa sostenida, también medida en celdas por segundo.

MBS, Maximum Burst Size, o tamaño de la ráfaga, medido en celdas.

MCR, Minimum Cell Rate o tasa mínima que la red debe garantizar.

No todos estos parámetros se contratan. Depende del servicio que se desee. En la tabla 6.2 se indica cuáles.

Se definen los descriptores de tráfico como el conjunto de parámetros que permiten describir un perfil de tráfico en función de un conjunto de valores, tales como el SCR, PCR, etc. El conjunto de parámetros varía en función del tipo de conexión contratada. Por ejemplo, las conexiones de tasa constante únicamente requieren del PCR para indicar el caudal, mientras que las de tasa variable exigen el empleo de otros valores adicionales.

De fuente: Usado durante call set-up para contratar conexión

De conexión: Parámetros descriptores de la conexión

Existen además los parámetros de QoS, negociables según el servicio. Son los siguientes:

CDVT, Cell Delay Variation Tolerance, referido a una tolerancia por el mero hecho de que el acceso al medio de celdas ATM se efectúa de forma ranurada.

Peak-to-Peak CDV, que es la variación de retardo extremo a extremo de la comunicación.

Max CTD, que es el máximo retardo fijo extremo-extremo

La relación de celdas perdidas o CLR, definido como celdas perdidas/Total celdas transmitidas.

En la figura 6.17 se muestra un comportamiento estadístico hipotético en una transmisión de celdas, donde quedan claramente reflejados los parámetros CDV y CTD.

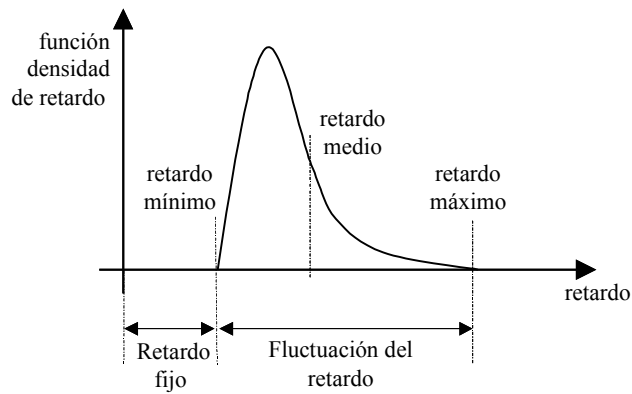


Fig. 6.17 Función densidad de probabilidad del retardo de llegada de las celdas

### 6.9.3 Notación

El ATM Forum define las siguientes categorías de servicio:

- a) CBR o servicios de tasa constante
- b) VBR o servicios de tasa variable
- c) ABR o servicios de tasa disponible, que serán estudiados más adelante. Son denominados también servicios “semi” best-effort, porque mejoran los servicios best-effort gracias a información de red en base al conocimiento del caudal instantáneo disponible
- d) UBR o servicios best-effort

La figura 6.18 muestra como todos estos servicios quedan dispuestos en un enlace. El servicio ABR aprovecha los sobrantes de CBR y VBR, que tienen mayor prioridad. Finalmente, UBR se queda con el caudal sobrante. Nótese que esta clasificación permite encajar perfectamente los requisitos de los servicios definidos en el apartado 6.1.2.

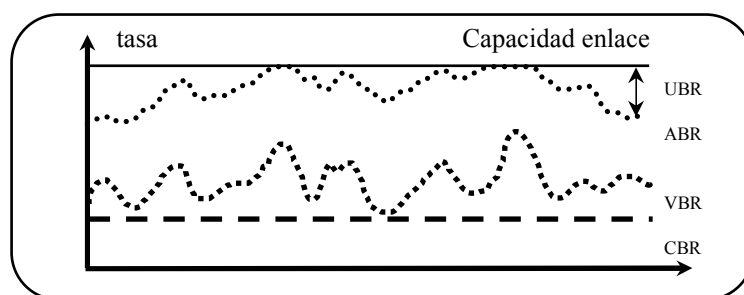


Fig. 6.18 Superposición de tráfico de diverso tipo sobre un enlace ATM

A la hora de establecer qué parámetros deben indicarse a la red ATM para solicitar una calidad de servicio, la tabla 6.2 muestra qué parámetros deben obligatoriamente indicarse para cada categoría de servicio.

Tabla. 6.2 Características definitorias de cada categoría de servicio, según el ATM Forum

Atributo \ Categoría Servicio	CBR	rt-VBR	nrt-VBR	UBR	UBR+	ABR
<b>Parámetros de tráfico</b>						
PCR, CDVT	o	o	o	o	Sup: o	o
SCR, MBS, CDVT	n/a	o	o	x	Sup: x	x
MCR	n/a	n/a	n/a	x	Sup: o	o
<b>Parámetros de QoS</b>						
Pico-a-Pico CDV	o	o	x	x	Sup: x	x
Máx CTD	o	o	x	x	Sup: x	x
<b>Otros Atributos</b>						
Realimentación	x	x	x	x	Sup: x	o
<b>Uso típico</b>	T-Real, QoS	T-Real, mux	Mux	Best-effort	Best-effort	adaptación tasa

#### 6.9.4 Control de Admisión de la Conexión

El CAC constituye las acciones llevadas a cabo durante el establecimiento / renegociación de la conexión.

Si una conexión admitida significa que la red dispone de suficientes recursos. Ahora bien, la conexión podría no ser admitida.

El criterio que se sigue es que “la admisión de una nueva conexión no puede perjudicar a las ya existentes”. Esta regla es fundamental para garantizar que los usuarios que ya disponen de una conexión, no sean perjudicados por nuevas conexiones.

#### 6.9.5 Función de UPC

La función de Usage Parameter Control se encarga de la protección de los recursos de la red, en especial de posibles mal funcionamiento de las fuentes o de comportamientos malintencionados.

Se puede emplear para tarificación, y si aplicación se efectúa en la UNI, pero también es posible en interior de la red.

El criterio básico de funcionamiento es el siguiente:

Mientras el usuario “cumpla” con su contrato de tráfico, la UPC sólo monitoriza. Si no cumple la función de UPC puede:

Marcar celdas a baja prioridad (CLP=0 → CLP=1)

Descartar celdas

Cerrar conexiones

Han existido diversos algoritmos para dar implementación a la UPC:

Jumping Window( figura 6.19), que permite un número limitado de celdas por ventana.

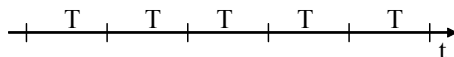


Fig. 6.19 Mecanismo de ventana fija

Moving Window, o ventana móvil (figura 6.20), que permite detectar picos de tráfico que en el caso de emplear jumping window podrían quedar ocultos en el paso entre una ventana y la siguiente.

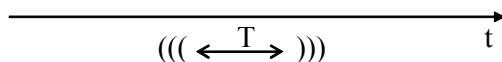


Fig. 6.20 Mecanismo de ventana deslizante o móvil

Ahora bien, el algoritmo adoptado es GCRA, como recomendación ITU-T, I.371. Existen 2 versiones equivalentes, Virtual Scheduling (VS) y Continuous State Leaky Bucket (LB).

En ambos casos, el algoritmo simplemente indica si una celda es o no conforme, dejando las tareas de descarte para otros mecanismos.

Los parámetros que requiere son:

GCRA (Incremento, Límite)

El algoritmo de funcionamiento queda descrito en la figura 6.21.

Ejemplos de aplicación son los siguientes:

Para una conexión CBR, el control se efectúa empleando GCRA(1/PCR, CDVT)

Para una conexión VBR se emplean 2 GCRA:

GCRA(1/PCR, CDVT) & GCRA(1/SCR, MBS)

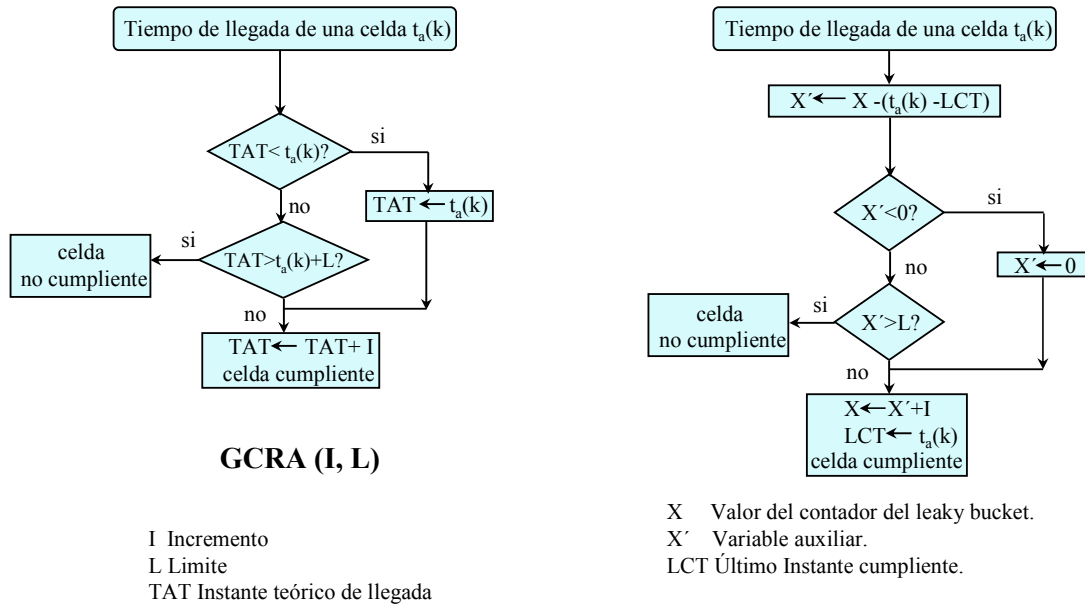


Fig. 6.21 Algoritmos VS y CSLB para GCRA

### 6.9.6 Conformación de tráfico

Las redes ATM permiten efectuar un moldeado del tráfico por medio de la aplicación de elementos de espaciado como indica la figura 6.22. Es lo que se denomina conformado de tráfico.

Con ello se pretende redondear los picos de tráfico, de tal modo que el caudal generado sea menos exigente con la red. El precio es un cierto incremento en el retardo.

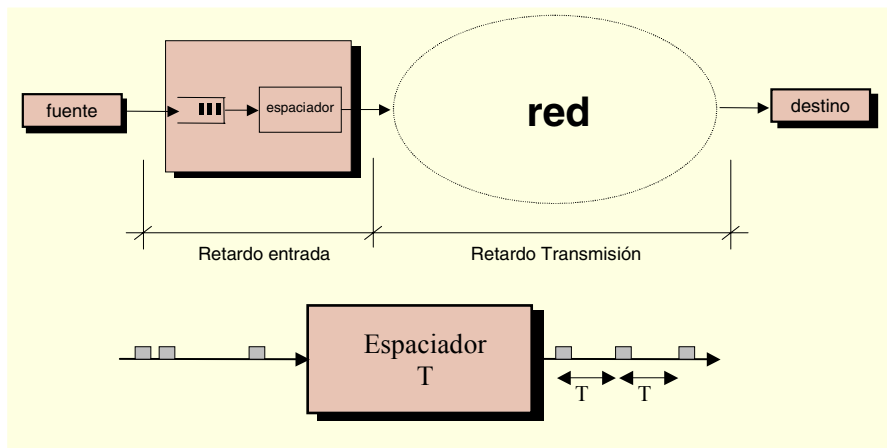


Fig. 6.22 Técnica de espaciado de celdas ATM

En el caso de multiplexar diversas fuentes, la realización puede llevarse a cabo como se muestra en la figura 6.23:

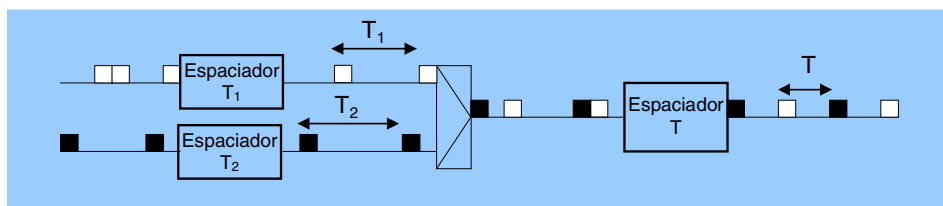


Fig. 6.23 Funcionamiento de espaciadores de fuente y de red

Que puede modelarse de la siguiente manera:

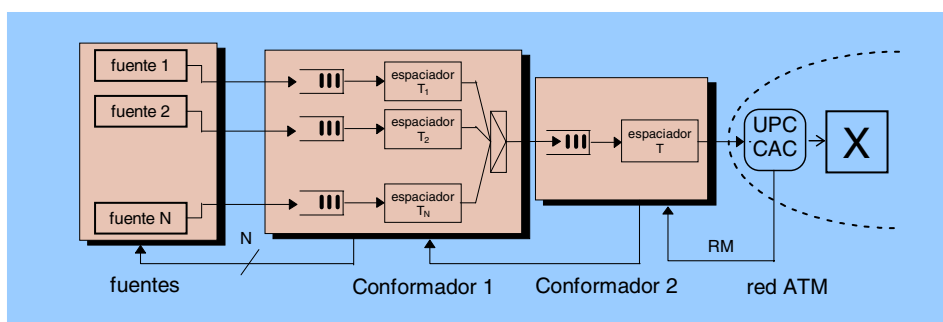


Fig. 6.24 Modelo de diseño del funcionamiento de espaciadores de fuente y de red en la UNI

### 6.9.7 El servicio ABR

El servicio ABR se nutre del caudal sobrante dejado por las conexiones CBR y VBR. Ello significa que el caudal disponible es desconocido. Por este motivo, la red dispone de un mecanismo de realimentación hacia el usuario que permite informarle del caudal disponible para él.

En este caso, el usuario debe adaptarse a un valor conocido. La red garantiza que si el usuario se adapta correctamente, la tasa de pérdidas será nula.

Para la implementación, se establece un algoritmo por cada conexión, mediante un mecanismo de realimentación (feedback), que emplea paquetes de exploración a través de la red: Son las celdas RM, ya mencionadas anteriormente. Estas celdas disponen de unos campos que son actualizados en cada elemento de conmutación, de tal manera que cuando retorna a la fuente, en los campos aparecen los

valores disponibles, a los que el usuario debe adaptarse. La figura 6.25 muestra gráficamente este comportamiento.

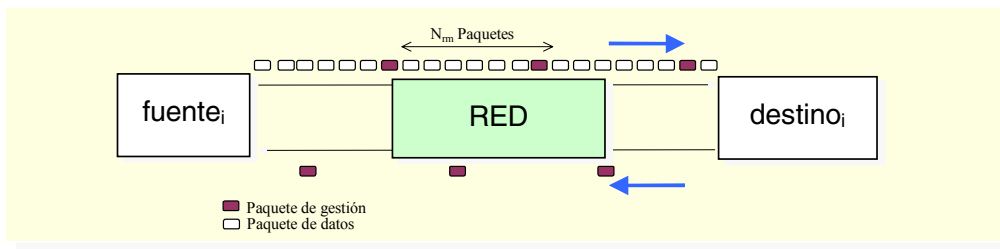


Fig. 6.25 Mecanismo de realimentación de celdas RM hacia la fuente

Los campos principales de la celda ABR son:

ID: Identificación de protocolo

BN (BECN): Backward Expl.Cong.Notif.

CI: Congestion Indication

NI: No Increase, para indicar que no debe incrementarse en la fuente la tasa de transmisión.

ER: Explicit Rate, con la tasa explícita de transmisión que debe emplear la fuente.

CCR: Current Cell Rate, o tasa actual a la que transmite la fuente.

MCR: Min. Cell Rate, que es la tasa mínima que debe garantizarse por contrato a la conexión.

### 6.9.8 Comportamiento de los conmutadores ante congestión

Ante todo, deben distinguirse dos tipos de conmutadores:

Los conmutadores binarios, que efectúan el marcado bit EFCI. Es esta una técnica muy simple. Además, si realizan el marcado de bits CI y NI se dice que realizan RR (*Relative Rate Marking*).

Los conmutadores ER (*Explicit Rate*) que modifica el campo ER para indicar el valor preciso al que la fuente debe transmitir. Lógicamente, estos equipos resultan más costosos puesto que requieren algoritmos capaces de computar este valor.

### 6.9.9 Comportamiento de las fuentes

En esta sección se describe a modo de algoritmo el comportamiento seguido por las fuentes:

Se transmite una celda RM cada  $N_{rm}$  celdas de datos. La transmisión se realiza a Tasa ACR (*allowed cell rate*). Esta variable se encuentra en cada terminal, controlando el caudal de cada conexión.

Al inicio de la transmisión:

El ACR se fija al valor predefinido ICR (*initial cell rate*). ICR se encuentra en el margen  $[MCR, PCR]$ .

En cada celda RM transmitida:

CCR adopta el valor del ACR actual.

En el campo ER se sitúa la tasa a la que se desea transmitir (usualmente, PCR)

En la recepción de una celda RM

Si CI no activado, se fija  $ACR = \min\{\text{incr.lin de RIF} \cdot PCR, PCR\}$

Sino, se decrementa por al menos  $RDF \cdot ACR$ , y siempre menor a ER pero nunca  $ACR < MCR$ . RDF es una constante predefinida.

Si el bit NI está activado, no se podrá incrementar el ACR

De acuerdo a este comportamiento, la evolución del caudal autorizado de una fuente sigue el siguiente mecanismo, según si se atiende a las actualizaciones recibidas, o bien al tiempo real transcurrido. Debe tenerse en cuenta que cuanto mayor es el caudal generado, mayor es el número de celdas RM por unidad de tiempo, y por lo tanto, mayor la frecuencia de actualización.

Según las actualizaciones recibidas, la evolución del caudal es el siguiente:

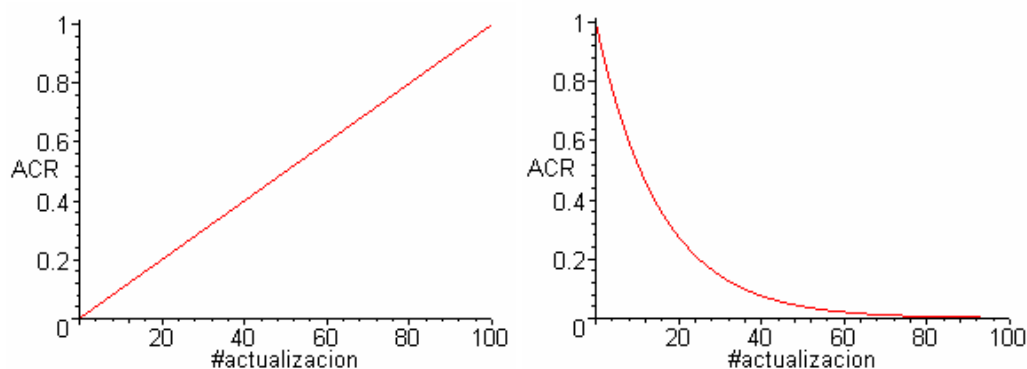


Fig. 6.26 Actualización del valor ACR en función del número de celdas RM recibidas

Según la evolución temporal, se tiene que:

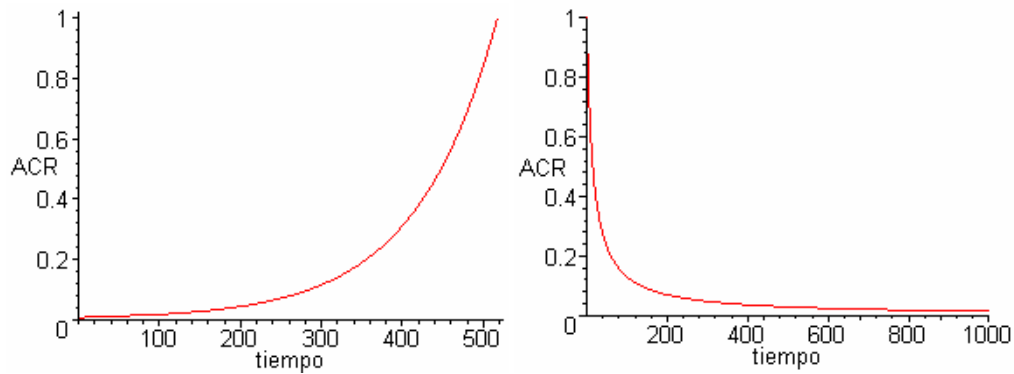


Fig. 6.27 Actualización del valor ACR en función del tiempo transcurrido

### 6.9.10 La repartición de los recursos disponibles ABR

El concepto de repartición justa (*Fairness*) puede ser definida como la distribución del caudal disponible sin discriminación entre las fuentes ni favoritismos. Para ello es necesario adoptar algún tipo de criterio. En el caso de  $MCR=0$ , el usado más frecuentemente es el denominado Max-Min. El objetivo del criterio Max-Min es asignar el mismo caudal para todas las conexiones que estén en “la misma situación dentro de la red”. Ello significa que fuentes que procedan de distinto origen, tendrán asignado un ancho de banda dependiente del estado de todas las conexiones que atraviesen, pero siempre partiendo de la base del criterio de igualdad. Este y otros criterios existentes se muestran en la siguiente tabla:

Tabla. 6.3 Diversos posibles criterios de repartición justa de recursos

Max-Min	$B(i) = B/N$
MCR + compartición equitativa	$B(i) = MCR_i + \frac{B-M}{N} \text{ con } M = \sum_i MCR_i$
Máx de MCR ó Max-Min	$B(i) = \max \left\{ MCR_i, \text{compartición\_según\_criterio\_MaxMin} \right\}$
Proporcional a MCR	$B(i) = B \cdot \frac{MCR_i}{M}$
Ponderada	$B(i) = B \cdot \frac{w(i)}{\sum_i w(i)}$

### 6.9.11 Cálculo del caudal justo mediante criterio Max-Min

La asignación de recursos debe efectuarse teniendo en cuenta las siguientes propiedades:

- Convergencia: Estabilización hacia un vector de tasas justas sin oscilaciones desde la condición inicial.
- Transitorio breve
- Robustez: Inmunidad a tráfico imprevisto o fallos de red.
- Escalabilidad: Capacidad de crecer con la red, al aumentar el núm.conexiones, conmutadores, distancia y tasa.
- Interoperatividad: Trabajo conjunto de equipos de diversos fabricantes.
- Baja complejidad de implementación.

Definiendo como conexiones estranguladas (*bottlenecked*) aquellas que no consiguen alcanzar su tasa justa en el enlace (es decir, el valor disponible para ellas en un enlace es superior al que existe debido a la asignación en otros enlaces más limitantes al repartir), de tiene que la tasa justa deberá se calculada como la total disponible descontando todas las conexiones que están estranguladas. Por ello, resulta inmediatamente la siguiente expresión:

$$Tasa\_justa = \frac{C_l - \sum Tasas\_de\_conexiones\_estranguladas\_en\_cualquier\_parte}{N_l - \sum Conexiones\_estranguladas\_en\_cualquier\_parte} \quad (6.1)$$

donde  $C_l$  es el caudal total disponible en el enlace  $l$  y  $N_l$  son el número de conexiones en dicho enlace.

Ejemplo: Considérese la siguiente topología de red, donde todas las conexiones establecidas sean de tipo ABR (Si hubiera conexiones de tipo CBR o VBR, el caudal consumido por ellas debería ser substraído antes de iniciar el cálculo. Nótese que, en general, este caudal depende del instante de tiempo).

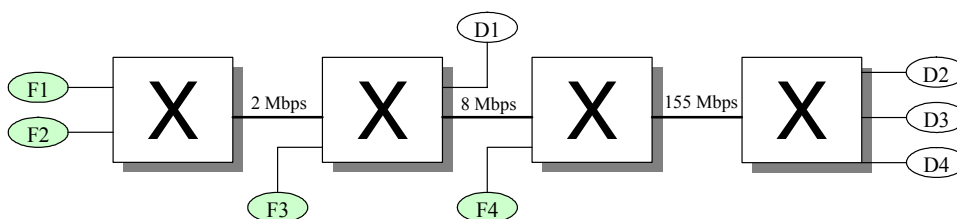


Fig. 6.28 Ejemplo de referencia para el estudio de la asignación de tasa

En este caso, se puede calcular que en el enlace de 2 Mbps, el caudal asignado a la F1 y la F2 será a priori de 1 Mbps. Dado que el destino de la F1 está en paralelo al enlace de 8 Mbps, resulta que el caudal para las fuentes F2 y F3 en este enlace será (a priori) de  $8 \text{ Mbps} / 2 \text{ fuentes} = 4 \text{ Mbps}$ , lo cual es superior al caudal asignado para F2. Es decir, la F2 está estrangulada debido al enlace de 2 Mbps.

En este caso, el caudal asignado para la F3 será, empleando la expresión anterior, de  $(8 \text{ Mbps} - 1 \text{ Mbps}) / (2 - 1) = 7 \text{ Mbps}$ .

Finalmente, repitiendo el mismo proceso para el enlace de 155 Mbps, se obtiene que el caudal para la F4 será de 147 Mbps.

Pudiera suceder que el enlace que en este caso es de 155 Mbps fuera más restrictivo que los anteriores. En este caso, debería rehacerse el cálculo.

La asignación para el caso aquí ilustrado es el siguiente:

Tabla 6.4 Resultado de la asignación de caudal mediante criterio Max-Min

# fuente	Tasa justa (Mbps)
1	1
2	1
3	7
4	147

De este ejemplo se deriva una conclusión muy importante: El cálculo de la tasa justa basado en criterio Max-Min parte de una expresión simple, aunque la obtención de los caudales no es un proceso simple, especialmente para un conmutador de la red que no tiene la suerte de disponer de la visión general de la red, que sí dispone el lector al efectuar el cálculo. El conmutador debe basarse en la observación parcial, de acuerdo a los valores de los campos de las celdas RM.

Se pueden distinguir dos tipos de algoritmos para la determinación del caudal justo: Los aproximados y los exactos. De los primeros, destaca el propuesto por Roberts y denominado Enhanced Proportional Rate Control Algorithm (EPRCA). Este algoritmo se basa en el denominado mercado inteligente.

Aunque su funcionamiento se aleja de los propósitos de este libro, puede mencionarse que se basa en el empleo de una cola de espera en la cual confluyen todas las conexiones ABR que atravesarán el conmutador. El tamaño de esta cola da noción del nivel de congestión del enlace de salida.

De acuerdo a dos umbrales, QT y DQT, se definen 3 zonas, tal como muestra la siguiente figura:

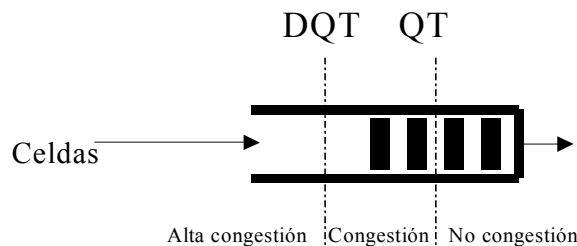


Fig. 6.29 Definición de zonas de congestión para el marcado inteligente de celdas

En la zona de no-congestión, se informa a las fuentes mediante las celdas RM de que pueden aumentar el caudal a las fuentes que confluyen en el conmutador. En la de alta-congestión, las celdas RM informan de que deben reducir su caudal. En la zona de congestión (congestión moderada), cada algoritmo actúa de modo particular, y es en general el parámetro distintivo de sus propiedades. En cualquier caso, debe tenerse en cuenta que la zona de congestión moderada es el punto adecuado de trabajo para el sistema, puesto que se asegura el empleo de todos los recursos disponibles sin desbordar la cola, es decir, con cotas de retardo y sin pérdidas.

### 6.9.12 Ejemplo práctico: Adaptador de red VMA-200 de Fore Systems

Las principales características de este adaptador ATM son las siguientes: Soporte para: TCP/IP, Circuitos Virtuales, Permanentes (PVCs), Conmutados (SVCs), Classical IP (CLIP), Emulación de LANs (LANE), Agente SNMP, UNI 3.0, Segmentación y Reensamblado (SAR) hardware para AAL 5 y AAL 3/4.

El hardware está constituido por los siguientes elementos:

Procesador i960 a 25 MHz en la tarjeta

Hardware para cálculos de HEC, CRC, AAL5 y AAL3/4.

155 Mbit/s SONET STS-3c /SDH STM-1

Cableado Dúplex 62.5/125 micron de fibra óptica multimodo (en 2 Km. máx, pérdidas de 10 dB), con conectores tipos ST y SC para fibra multimodo, o bien con UTP categoría 5 y conector RJ-45

Como ilustración de funcionamiento, el proceso de instalación del controlador de la tarjeta se lleva a cabo de acuerdo a los siguientes pasos:

```
Enter the name for the FORE ATM interfaces [fa]: fa
The interfaces will be called fa.
```

```
Which device driver should be installed?
```

- 1) GIA-200
- 2) ESA-200
- 3) VMA-200

Device driver number? 3

Installing kernel configuration files for the VMA-200

Ready ? [y] y

VMA-200 driver object module installed

VMA-200 master file installed with minor number 114

VMA-200 master.d entry installed

Entries for VMA-200 installed in /var/sysgen/system/fore\_atm\_.sm

Creating device nodes in /dev ...

Device nodes successfully created.

Into which directory should firmware, utilities, and man pages be installed?

[/usr/etc/fore]

Installing initialization script, /etc/init.d/fore\_atm ...

Installing FORE applications in /usr/etc/fore/etc ...

Installing man pages in /usr/etc/fore/man ...

Would you like to use FORE's SNMP agent? [y] n

Will you be using ILMI for Address Registration? [y] n

Would you like to configure Classical IP? [n] y

Would you like to configure qaa0? [y] y

Enter the ATM address for this node

47.0005.80.ffe100.0000.f115.0bab.002048102fe4.00

Enter the ATM address for the ARP server on qaa0

123456789abcdef0123456789abcdef012345678

Would you like to configure qaa1? [y] n

Would you like to configure LAN Emulation? [n] n

Ready to configure kernel ? [y] y

Working (this may take a few minutes)...

Kernel build successful. The new kernel is /unix.install.

Please move the original kernel, /unix, to /unix.orig and then reboot your machine to begin running the new kernel.

Automatic installation has completed.

NOTE: to start the FORE ATM LAN during system boot, you must modify your initialization configuration. For example, you could make the following changes:

1) In /etc/config/netif.options, add the lines

```
if2name=fa0
```

```
if2addr=$HOSTNAME-atm
```

2) Create /etc/config/ifconfig-2.options with the line

```
up netmask 0xffffffff00
```

3) Add \$HOSTNAME-atm to /etc/hosts

## Bibliografía

ALQUEZAR, R.; otros. "Algorithms for signal processing of ultrasonic images". Contrato ESA nº 7320/87/NL/PP. Noviembre 1988.

ANDERBERG, M.R. *Cluster Analysis for Applications*. Academic Press, New York, 1973.

ARULAMBALAM, A.; CHEN, X. "Allocating Fair Rates for Available Bit Rate Service in ATM networks" IEEE Communications. Vol.34, No.11, pp.92-100. Noviembre 1996.

ASATANI, K.; otros. *Introduction to ATM networks and B-ISDN*. Wiley, 1997.

ATM FORUM. "ATM User-Network Interface Specification". Version 3.1. Septiembre 1994.

ATM FORUM. "Traffic Management Specification". Version 4.0. Abril 1996.

BARNHART, A.W. "Explicit Rate Performance Evaluations". Contribución al ATM Forum. 94-0983, Octubre 1994.

BENAIM, M.; TOMASINI, L. "Approximating functions and predicting time series with multi-sigmoidal basis functions". Artificial Neural Networks, Vol.1, pp.407-411. Elsevier Science Publisher. Amsterdam, 1992.

BENNET, J.C.R.; ZHANG, H. "WF<sup>2</sup>Q: Worst case Fair Weighted Fair Queueing". pp.120-128, INFOCOM, San Francisco, 1996

BERTSEKAS, D.; GALLAGER, R. *Data Networks*. Prentice Hall, 1987.

BONOMI, F.; FENDICK, K. W. "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service". IEEE Network. Vol.10, No.2, pp.25-39. Marzo/Abril, 1995.

BRADEN, R.; ZHANG, L.; BERSON, S.; HERZOG, S.; JAMIN, S. "Resource Reservation Protocol (RSVP). Version 1 Functional Specification". Internet Draft. Noviembre, 1996.

BROWNING, D. W. "Flow control in high-speed communication networks". IEEE Transactions on Communications, Vol.42, No.7, pp.2480-2489, Julio 1994.

CHEN, T.M.; LIU, S.S.; SAMALAM, V.K. "The Available Bit Rate Service for Data in ATM Networks". IEEE Communications Magazine. 34, 5, pp.56-71, 1996.

CHIUSI, F.M.; XIA, Y.; KUMAR, V.P. "Dynamic Max Rate Control Algorithm for Available Bit Rate Service in ATM Networks". Globecom'96, pp.2108-2117. Londres, 1996.

DSS. *Core aspects of Frame Relay protocol for the use with Frame Relay Bearer service*. American National Standards Institute, Telecommunications Committee (ANSI T1S1). Diciembre, 1990.

FENDICK, K.W. "Evolution of Controls for the Available Bit Rate Service". IEEE Communications. Vol.34, No.11, pp.35-39. Noviembre 1996.

FERNÁNDEZ, C.; SALLEN, S.; HESSELBACH, X. "Contributions to the Evaluation of MAC protocols over HFC Access Networks". International Symposium on Computer and Information Sciences, ISCIS'98, Antalya (Turquía) 1998.

FIOL M.A.; otros. *Análisis Numérico*. Centro de Publicaciones de la Escuela de Telecomunicaciones, Barcelona, 1990.

GEORGIADIS, L.; otros. "Efficient Network Qos Provisioning Based on per Node Traffic Shaping". IEEE/ACM Transactions on Networking. Vol.4, No.4. pp.482-501. Agosto, 1996.

GITLIN, R.D.; HAYES, J.F.; WEINSTEIN, S.B. *Data Communications Principles*. Ed. Plenum, 1992.

GUÉRIN, R.; PERIS, V. "Quality of Service in packet networks: basic mechanism and directions". Computer Networks, Vol.31, pp.169-189. Febrero 1999.

HAMDI, M.; ROBERTS, J.; ROLIN, P. "Rate control for VBR video coders in Broadband networks". IEEE Journal on Selected Areas in Communications, Vol.15, No. 6, pp.1040-1051. Agosto 1997.

HESSELBACH, X. "On the Quality of Service Enhancement in Variable Bit Rate Networks". Looking forward. IEEE Computer. Vol.6, No.1. Primavera 1998.

HESSELBACH, X.; BARBA, A.; SALLEN, S. "Estudio de un alisador de tráfico multiplexado de vídeo síncrono en ATM bajo servicio ABR". VI Jornadas de I+D en Telecomunicaciones. Telecom I+D. Madrid. 1996.

HESSELBACH, X.; CAMPILLO, D.; MORILLO, O. "Estudio de la calidad de servicio de un alisador de tráfico destinado a aplicaciones de audio y vídeo". XII Symposium Nacional de la Unión Científica Internacional de Radio, URSI. Bilbao. 1997.

HESSELBACH, X.; LÍNDEZ, J.M.; SALLEN, S. "Aplicación de técnicas de predicción para la gestión de recursos en redes de servicios multimedia bajo demanda". XIII Symposium Nacional de la Unión Científica Internacional de Radio, URSI. Pamplona. 1998.

HESSELBACH, X.; OLIVER, M.; BORRÀS, J. "Model of an adaptive access node in a packet network: The ABR service in ATM". Eunice'99. pp.125-130. Barcelona, 1999.

HESELBACH, X.; SALLEN, S. "A compatible ATM-DQDB interconnection in a Broadband Multi-Internetworking Unit". 12th International Conference on Computer Communication, ICC'95. Seoul (Corea). 1995.

HESELBACH, X.; SALLEN, S. "Control de la tasa de fuente para el multiplexado de celdas bajo minimización del CTD y CDV en redes ATM". Youth Forum in Computer Science and Engineering. Yuforic'97. Barcelona. 1997.

HESELBACH, X.; SALLEN, S. "Dimensionado del Nodo de Acceso con Control Explícito de tasa DMRCA para redes de paquetes". Jitel'99. Madrid. 1999.

HESELBACH, X.; SALLEN, S. "Dimensioning the input buffers in the access node for fair bandwidth allocation with source rate control in packet networks using prediction methods". IMSA'99, Nassau, Grand Bahamas, Octubre, 1999.

HESELBACH, X.; SALLEN, S. "Estudio del control de la tasa de fuente en servicio ABR para aplicaciones de audio y vídeo". Jornadas de Ingeniería Telemática, Jitel'97. Bilbao. 1997.

HESELBACH, X.; SALLEN, S. "Evaluación de la calidad de servicio en un nodo de acceso con control de tasa para conexiones ABR en redes de Banda Ancha". Telecom I+D. Barcelona-Madrid, 1999.

HESELBACH, X.; SALLEN, S.; BARBA, A. "Alisado de tráfico de vídeo síncrono para su transmisión en redes ATM bajo servicio ABR". XI Symposium Nacional de la Unión Científica Internacional de Radio, URSI. Madrid. 1996.

HESELBACH, X.; SALLEN, S.; FERNÁNDEZ, C. "Quality of Service management for audio and video on demand services". IEEE Conference on Protocols for Multimedia Systems - Multimedia Networking, PROMSMNet'97. Santiago de Chile, Chile. 1997.

Hock, C. *Queueing Modelling Fundamentals*. Wiley, 1996.

HÜBNER, F.; TRAN-GIA, P. "Discrete-time analysis of cell spacing in ATM systems". Telecommunication Systems3, pp.379-395, 1995.

ISO/IEC. "Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbps". ISO/IEC 11172-2, 23 Noviembre 1991.

ITU-T. "Traffic Control and Congestion Control in B-ISDN". Recomendación I.371 del ITU-T, 1993.

JAIN, A. K. *Fundamentals of digital image processing*. Prentice Hall, 1989.

JAIN, R.; KALYANARAMAN, S.; FAHMY, S.; GOYAL, R. "Source Behavior for ATM ABR Traffic Management: An Explanation". IEEE Communications. Vol 34, No. 11, pp.50-57. Noviembre 1996.

JAIN, R.; KALYANARAMAN, S.; GOYAL, R.; FAHMY, S.; VISWANATHAN, R. "ERICA Switch Algorithm: A Complete Description". Contribución ATM Forum. 96-1172, Agosto, 1996.

- KESHAV, S. *An Engineering Approach to Computer Networks*. Addison Wesley, 1997.
- KLEINROCK, L. *Queueing Systems, Volume I: Theory*. John Wiley & Sons, 1975.
- KLEINROCK, L. *Queueing Systems, Volume II: Computer Applications*. John Wiley & Sons, 1976.
- KYAS, O. *ATM Networks*. International Thomson Publishing, 1995.
- LAKSHMAN, T.V.; MISHRA, P.P.; RAMAKRISHMAN, K.K. "Transporting compressed video over ATM networks with explicit rate feedback control". Proc.IEEE INFOCOM, pp.38-47. Kobe, Japón, 1997.
- LEDUC, J.P. *Digital Moving Pictures - Coding and Transmission on ATM Networks*. Elsevier, 1994.
- LEON-GARCÍA, A. *Probability and Random Processes for Electrical Engineering*. Addison-Wesley, 1989.
- LUO, W.; EL ZARKI, M. "Quality Control for VBR video over ATM networks". IEEE Journal on Selected Areas in Communications. Vol.15, No.6, pp.1029-1039. Agosto, 1997.
- MAGLARIS, B.; otros. "Performance Models of Statistical Multiplexing in Packet Video Communication". IEEE Transactions on Communications. Vol.36, No.7, pp.834-844. Julio, 1988.
- MUDDU, S.; CHIUSSI, F.M.; TRYFONAS, C.; KUMAR, V.P. "Max-Min Rate Control Algorithm for Available Bit Rate Service in ATM networks". ICC'96, pp.412-418. Dallas. 1996.
- NIKOLAIDIS, I.; AKYILDIZ, I.F. *Source Characterization and Statistical Multiplexing in ATM networks*. College of Computing. Georgia Institute of Technology. Atlanta. Julio, 1992.
- OHSAKI, H.; MURATA, M.; SUZUKI, H.; IKEDA, C.; MIYAHARA, H. "Rate-Based Congestion Control for ATM Networks". ACM SIGCOMM, pp.60-71. 1996.
- OHTA, N. *Packet Video. Modeling and Signal Processing*. Artech House Inc. 1994.
- OMIDYAR, C.G.; PUJOLLE, G. "Introduction to Flow and Congestion Control". IEEE Communications Magazine. 34, 11, pp.30-32, 1996.
- PITTS, J.M.; SCHORMANS, J.A. *Introduction to ATM design and performance*. Wiley, 1996.
- PRYCKER, M. *Asynchronous Transfer Mode. Solution for Broadband ISDN*. Prentice Hall, 1995.
- RAMAKRISHAN, K.K.; NEWMAN, P. "Integration of Rate and Credit Schemes for ATM Flow Control". IEEE Network. 9, 2, pp.49-56, 1995.
- ROBERTS, L. "Enhanced PRCA (Proportional Rate-Control Algorithm)". Contribución ATM Forum. 94-0735R1, Agosto, 1994
- ROSBERG, Z. "Cell Multiplexing in ATM Networks". IEEE/ACM Transactions on Networking. 4, 1, pp.112-122, 1996.

ROSS, S. M. *Introduction to Probability Models*. Academic Press Inc, 1989.

SCHWARTZ, M. *Broadband Integrated Networks*. Prentice Hall PTR, 1996.

SHROFF, N.; SCHWARTZ, M. "Video Modeling within Networks using Deterministic Smoothing at the Source". pp.342-349, INFOCOM, Toronto, 1994.

SOTO, M.; HESSELBACH, X.; SALLEN, S. "Acceso a una red ATM a través de un medio compartido". XI Symposium Nacional de la Unión Científica Internacional de Radio, URSI. Madrid. 1996.

STALLINGS, W. *Comunicaciones y redes de computadores*. Prentice. Mayo, 1997.

STROBACH, P. *Linear Prediction Theory*. Springer-Verlag, 1990.

TANENBAUM, A S. *Computer Networks*. Prentice-Hall International Editions, 1988.

VARMA, A.; STILIADIS, D. "Hardware Implementation of Fair Queueing Algorithms for Asynchronous Transfer Mode Networks". IEEE Communications. Vol.35, No.12, pp.54-68. Diciembre, 1997.

ZHANG, H. "Service Disciplines for Guaranteed Performance Services in Packet Switching Networks". Proceedings of the IEEE. Vol.83, No.10. pp.1374-1396. Octubre, 1995.

ZHANG, L.; DEERING, S.; ESTRIN, D.; SHENKER, S.; ZAPPALA, D. "RSVP: A new Resource ReSerVation Protocol". IEEE Network, Vol.7, No.5, pp.8-18, Septiembre, 1993.

ZHENG, B.; ATIQUZZAMAN, M. "Traffic Management of Multimedia over ATM Networks". IEEE Communications. Vol.37, No.1, pp.33-38. Enero, 1999.