ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE INGENIERÍA

ESTUDIO DE LOS FACTORES TÉCNICOS Y OPERATIVOS QUE INTERVIENEN EN LA INFRAESTRUCTURA DE CALIDAD DE SERVICIO EN INTERNET

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO ESPECIALISTA EN ELECTRÓNICA Y TELECOMUNICACIONES

TOMO I

LOOR FONSECA DIEGO CHRISTIAN
PICHOASAMÍN MORALES LUIS HUMBERTO

DIRECTOR: Ing. María Soledad Jiménez

Quito, Enero 2001

DECLARACIÓN

Nosotros, Diego Christian Loor Fonseca y Luis Humberto Pichoasamín Morales, declaramos que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley, Reglamento de Propiedad Intelectual y por la normatividad institucional vigente.

Diego Loor Fonseca

Luis Pichoasamín Morales

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Diego Christian Loor Fonseca y Luis Humberto Pichoasamín Morales, bajo mi supervisión.

Ing. María Soledad Jiménez
DIRECTOR DE PROYECTO

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Diego Christian Loor Fonseca y Luis Humberto Pichoasamín Morales, bajo mi supervisión.

Ing. Milton Ludeña Trujillo
CO-DIRECTOR DE PROYECTO

AGRADECIMIENTO

Nuestros sinceros agradecimientos a los Ingenieros María Soledad Jiménez y Milton Ludeña por su valioso aporte y predisposición para el desarrollo y culminación del presente trabajo.

A la Escuela Politécnica Nacional, porque además de los sólidos y valiosos conocimientos, nos inculcó una filosofía de superación y trabajo.

A todas las personas que colaboraron directa e indirectamente para la consecución de nuestro objetivo.

A Dios por concederme el don de la vida.

A mis padres Luis Humberto, y Olga Yolanda
por el inmenso amor y abnegación que me han brindado.

A mis hermanos por ser fuentes inagotables de
apoyo y comprensión.

A todos mis amigos pues gracias a ellos he
logrado superar muchas dificultades y llegar
a la culminación de este objetivo.

Luis

ÍNDICE DE CONTENIDOS

RESUMEN i

PRESENTACIÓN iii

CAPÍTULO 1

MOTIVACIONES PARA LA PROVISIÓN DE CALIDAD DE SERVICIO

1.1 EL MODELO *BEST EFFORT* **1**

1.2 CAMBIO DE CARÁCTER EN EL INTERNET ACTUAL **5**

1.2.1 NUEVOS SERVICIOS Y APLICACIONES 5

1.2.1.1 Voz sobre circuitos de datos 5

1.2.1.2 Servicios de Videoconferencia 7

1.2.1.3 Red de Negocios de Próxima Generación 8

1.2.2 DESARROLLO DE HARDWARE Y SOFTWARE 9

1.2.2.1 Desarrollo de los computadores 9

1.2.2.2 Desarrollo de la tecnología de conectividad 10

1.3 ADMINISTRACIÓN DEL ANCHO DE BANDA **12**

1.4 CALIDAD DE SERVICIO: CONCEPTOS Y DEFINICIONES **14**

1.4.1 MECANISMOS DE MANIPULACIÓN DE TRÁFICO 16

1.4.1.I 802.1p 16

1.4.1.2 DiffServ 17

1.4.1.3 IntServ 17

1.4.1.4 ATM, ISSLOW y otros 18

1.4.1.5 Mecanismos de manipulación de tráfico por conversación vs. Agregados 20

1.4.2 MECANISMOS DE PROVISIÓN Y CONFIGURACIÓN 21

1.4.2.1 Provisión vs. Configuración 21

1.4.2.2 Mecanismos *Top-Down* vs. Señalizados 21

1.4.2.3 Protocolo de reservación de recursos (RSVP) 22

1.4.2.4 Administrador del ancho de banda de la subred (SBM) 22

1.4.3 MECANISMOS Y PROTOCOLOS DE POLÍTICAS 23
1.4.3.1 Bases de datos de Políticas – Servicios de Directorio 24
1.4.3.2 PDPs y PEPs 24
1.4.3.3 Protocolos de Políticas 25
1.5 BENEFICIOS DE LA IMPLEMENTACIÓN DE QoS 26
I.5.1 BENEFICIOS ECONÓMICOS 26
I.5.2 DESARROLLO DE NUEVA TECNOLOGÍA 27
1.5.3 BENEFICIOS PARA LAS APLICACIONES 27
1.5.4 BENEFICIOS PARA LAS EMPRESAS 28
1.5.5 BENEFICIOS PARA LOS PROVEEDORES DE SERVICIO 28

CAPÍTULO 2

PARÁMETROS TÉCNICOS QUE DEFINEN LA CALIDAD DE SERVICIO

2.1 RETARDO o LATENCIA 30
2.1.1 RETARDO EN LA INTERRED 31
2.1.1.1 Retardo en el Acceso en las Redes Ethernet 34
2.1.1.2 Retardo en el Acceso en las Redes Token Ring 37
2.1.1.3 Retardo en el Acceso Remoto – módems 38
2.1.2 RETARDO EN LOS PROTOCOLOS 43
2.I.2.1 Retransmisión en redes X.25 44
2.1.2.2 Retransmisión en redes Frame Relay 45
2.1.2.3 Retransmisión en el Protocolo TCP (*Transport Control Protocol*) 46
2.1.3 RETARDO EN LOS SERVIDORES 49
2.2 FLUCTUACIÓN DEL RETARDO O *JITTER* 51
2.2.1 FLUCTUACIÓN FÍSICA 52
2.2.2 FLUCTUACIÓN EN EL ACCESO 52
2.2.3 FLUCTUACIÓN EN LA RED 52
2.2.4 FLUCTUACIÓN EN EL ESTABLECIMIENTO DE LA SESIÓN 53
2.3 ANCHO DE BANDA 53

2.3.1 MEDIOS DE TRANSMISIÓN 55
2.3.1.1 Par trenzado 55
2.3.1.2 Cable coaxial 56
2.3.1.3 Fibra óptica 57
2.3.2 ANCHO DE BANDA Y LAS APLICACIONES 58
2.4 CONFIABILIDAD 59
2.5 DIFERENTES TIPOS DE TRÁFICO 61
2.5.1 VOZ 62
2.5.2 VIDEO 66
2.5.2.1 Compresión *Interframe* 67
2.5.2.2 Compresión *Intraframe* 69
2.6 RED CONVERGENTE 70
2.6.1 CARACTERÍSTICAS DE UNA RED CONVERGENTE 70
2.6.2 USOS PRÁCTICOS DE REDES CONVERGENTES 72

CAPÍTULO 3

MECANISMOS DE GESTIÓN DE TRÁFICO

3.1 INTRODUCCIÓN 74
3.2 ADMINISTRACIÓN ACTIVA DE COLAS 75
3.2.1 CARACTERÍSTICAS 75
3.2.2 VENTAJAS 75
3.3 ALGORITMOS DE ENCOLAMIENTO 76
3.3.1 ENCOLAMIENTO FIFO 76
3.3.1.1 Funcionamiento 76
3.3.I.2 Rendimiento 77
3.3.2 ENCOLAMIENTO JUSTO PONDERADO (WFQ, *WEIGHTED FAIR QUEUING*) ... 78
3.3.2.1 Funcionamiento 79
3.3.2.2 Rendimiento 82

3.3.2.3 WFQ bajo el modelo GPS (*Generalized Processor Sharing*) 83

3.3.3 ENCOLAMIENTO BASADO EN CLASES CBQ (*CLASS BASED QUEUING*) 87

3.3.3.1 Funcionamiento 87

3.3.3.2 Rendimiento 90

3.4 ALGORITMOS DE DESCARTE DE PAQUETES 90

3.4.1 ALGORITMO DE DETECCIÓN ALEATORIA TEMPRANA (RED, *RANDOM EARLY DETECTION*) 90

3.4.1.1 Funcionamiento 91

3.4.1.2 Rendimiento 92

3.4.2 MÉTODO DE NOTIFICACIÓN EXPLÍCITA DE CONGESTIÓN (ECN, *EXPLICIT CONGESTION NOTIFICATION*) 94

3.4.2.1 Características del Método de Notificación Explícita de Congestión. 95

3.4.2.2 Cabecera de Notificación Explícita de Congestión en IP 95

3.4.2.3 ECN en TCP 97

3.4.2.4 Regiones no colaboradoras de ECN 101

3.4.2.5 Justificación del bit ECT 102

3.4.2.6 Implementación ECN Alternativa 103

CAPÍTULO 4

ARQUITECTURA DE SERVICIOS DIFERENCIADOS

4.1 ARQUITECTURA DIFFSERV 106

4.1.1 CARACTERÍSTICAS DE LA ARQUITECTURA 106

4.1.2 *DiffServ* vs OTROS MODELOS 107

4.2 MODELO DE LA ARQUITECTURA DE SERVICIOS DIFERENCIADOS 109

4.2.1 DOMINIO DE SERVICIOS DIFERENCIADOS 109
4.2.2 REGIÓN DE SERVICIOS DIFERENCIADOS 111
4.2.3 CLASIFICACIÓN Y ACONDICIONAMIENTO DEL TRÁFICO 112
4.3 MODELO CONCEPTUAL DE UN *ROUTER DIFFSERV* 113
4.3.1 CLASIFICADORES 115
4.3.1.1 Clasificador BA 117
4.3.1.2 Multiclasificador BA 118
4.3.2 MEDIDORES 118
4.3.2.1 Perfiles de tráfico 119
4.3.2.2 Ejemplos de medidores 120
4.3.2.2.1 Medidor de velocidad promedio 120
4.3.2.2.2 Medidor EWMA (Exponential Weighted Moving Average) 121
4.3.2.2.3 Medidor token bucket sencillo 122
4.3.2.2.4 Medidor token bucket multi-etapa 123
4.3.3 ELEMENTOS ACTIVOS 124
4.3.3.1 Marcadores 124
4.3.3.2 Formadores 125
4.3.3.3 Descartadores 125
4.3.3.4 Bloques acondicionadores de tráfico (TCBs, *Traffic Conditioning Block*) 125
4.3.3.5 Localización de acondicionadores de tráfico 131
4.3.3.5.1 Dentro del dominio fuente 132
4.3.3.5.2 En la frontera del dominio DS 132
4.3.4 ASIGNACIÓN DE RECURSOS DE RED 133
4.3.5 INTEROPERABILIDAD CON NODOS NO *DiffServ* 133
4.4 CABECERA IP Y *CODEPOINTS DIFFSERV* 135
4.5 COMPORTAMIENTOS POR SALTO (PHBs, *PER HOP BEHAVIOURS*) 139
4.5.1 PHB BÁSICO 140
4.5.2 PHB DE ENVÍO ACELERADO (EF, *Expedited Forwarding*) 140
4.5.2.1 Descripción del PHB EF 141
4.5.2.2 Mecanismos para implementar PHB EF 142
4.5.3 GRUPO DE PHBs DE ENVÍO ASEGURADO (AF, *Assured Forwarding*) 143
4.6 AGREGACIÓN DE COMPORTAMIENTOS (BA, *BEHAVIOR AGGREGATES*) 146
4.6.1 CARACTERÍSTICAS 146

4.6.2 PROPIEDADES 147
4.6.3 EJEMPLOS DE AGREGACIONES DE COMPORTAMIENTOS 148
4.6.3.1 Agregación de Comportamientos *Best Effort* 148
4.6.3.2 Agregación de Comportamientos de Manipulación Voluminosa 148
4.6.3.3 Agregación de Comportamientos Tolerante a las Pérdidas 149
4.6.3.4 Agregación de Comportamientos Preferencial 149
4.7 CLASIFICACIÓN EN LAS DIFERENTES CAPAS 150
4.7.1 CLASIFICACIÓN EN CAPA 3 150
4.7.2 CLASIFICACIÓN EN CAPA 2 151

CAPÍTULO 5

ARQUITECTURA DE SERVICIOS INTEGRADOS

5.1 ELEMENTOS DE LA ARQUITECTURA *IntServ* 152
5.1.1 MODELO DE SERVICIOS INTEGRADOS 153
5.1.1.1 Requerimiento de QoS para un flujo 154
5.1.1.2 Requerimientos de Compartición de Recursos 155
5.1.1.3 Descarte de Paquetes 156
5.1.1.4 Modelo de Reservación 157
5.1.2 IMPLEMENTACIÓN DE REFERENCIA 157
5.1.2.1 Organizador de Paquetes 158
5.1.2.2 Clasificador 159
5.1.2.3 Control de Admisión 160
5.1.2.4 Aplicación de los mecanismos de Control de Tráfico 160
5.1.2.5 *Router* capacitado con Servicios Integrados. 161
5.2 SERVICIOS DE LA ARQUITECTURA IntServ 163
5.2.1 SERVICIO DE CARGA CONTROLADA 163
5.2.1.1 Conducta extremo a extremo 163
5.2.1.2 Requerimientos de los elementos de red 164
5.2.1.3 Caracterización del tráfico (TSpec) 165
5.2.1.4 Políticas de Funcionamiento 167

5.2.1.5 Implementación del Servicio de Carga Controlada 169
5.2.1.6 Evaluación del Servicio de Carga Controlada 170
5.2.2 SERVICIO GARANTIZADO 172
5.2.2.1 Conducta extremo a extremo 172
5.2.2.2 Requerimientos de los elementos de red 173
5.2.2.3 Caracterización del tráfico (TSpec) y de los recursos (RSpec) 174
5.2.2.4 Información a exportar 176
5.2.2.5 Políticas de Funcionamiento 176
5.2.2.6 Implementación del Servicio Garantizado 177
5.2.2.7 Evaluación del Servicio Garantizado 181
5.3 PROTOCOLO DE RESERVACIÓN DE RECURSOS RSVP 181
5.3.1 INTRODUCCIÓN 181
5.3.2 MODELO DE RESERVACIÓN 183
5.3.2.1 Estilos de Reservación 185
5.3.2.1.1 Estilo Filtro Comodín (WF, Wildcard Filter) 185
5.3.2.1.2 Estilo Filtro Fijo (Fixed Filter) 186
5.3.2.1.3 Estilo Compartición Explícita (SE, Shared-Explicit) 186
5.3.3 MECANISMOS DEL PROTOCOLO RSVP 189
5.3.3.1 Mensajes *Path* y *Resv* 189
5.3.3.1.1 Mensaje Path 189
5.3.3.1.2 Mensaje Resv 189
5.3.3.1.3 Reservación 190
5.3.3.1.4 Estados de la Reservación 191
5.3.3.2 Mensaje Cancelar (*Teardown*) 192
5.3.3.3 Mensajes de errores y Estado de Bloqueo 193
5.3.3.4 Mensaje de Confirmación (*ResvConf*) 195
5.3.3.5 Parámetros de Tiempo 196
5.3.3.6 Control de Políticas 196
5.3.3.7 Seguridad 197
5.3.3.8 Nubes No-RSVP 197
5.3.3.9 Compatibilidad Futura 197
5.3.4 ESPECIFICACIÓN FUNCIONAL 198

5.4 ARQUITECTURA *IntServ* SOBRE TECNOLOGÍAS ESPECÍFICAS DE CAPA ENLACE 202
5.4.1 ARQUITECTURA DE SERVICIOS INTEGRADOS SOBRE ATM 202
5.4.1.1 Parámetros ATM 203
5.4.1.2 Categorías de Servicio de ATM 204
5.4.1.2.1 Velocidad Constante de Bit (CBR, Constant Bit Rate) 204
5.4.1.2.2 Velocidad Variable de Bit para Tiempo Real (rtVBR, real time Variable Bit Rate) 204
5.4.1.2.3 Velocidad Variable de Bit inadecuado para Tiempo Real (nrtVBR, no real time Variable Bit Rate) 205
5.4.1.2.4 Velocidad Disponible de Bit (ABR, Available Bit Rate) 205
5.4.1.2.5 Velocidad No Especificada de Bit (UBR, Unspecified Bit Rate) 205
5.4.1.3 Interpretación de los Servicios Integrados 205
5.4.1.3.1 Servicio Garantizado y ATM 206
5.4.1.3.2 Servicio de Carga Controlada y ATM 207
5.4.1.3.3 Servicio Best effort y ATM 208
5.4.2 SERVICIOS INTEGRADOS SOBRE ENLACES DE BAJA TASA DE TRANSMISIÓN 210
5.4.2.1 Encapsulamiento en tiempo real 210
5.4.2.1.1 Fragmentación IP 211
5.4.2.1.2 Mecanismos de Capa de Enlace 211
5.4.2.2 Compresión de Cabeceras 215

CAPÍTULO 6

ARQUITECTURAS HÍBRIDAS

6.1 SERVICIOS INTEGRADOS SOBRE UNA RED DiffServ 216
6.1.1 BENEFICIOS 217
6.1.2 MODELO DE LA ARQUITECTURA *IntServ/DiffServ* 218
6.1.2.1 Aprovisionamiento estático 219

6.1.2.2 Aprovisionamiento Dinámico 220
6.1.2.3 Correlación entre las regiones *IntServ* y *DiffServ* 224
6.1.2.4 Objeto RSVP DCLASS (*Differentiated CLASS*) 225
6.1.3 CONSIDERACIONES ADICIONALES 226
6.2 MPLS Y *DIFFSERV* 227
6.2.1 MPLS Y CONMUTACIÓN DE HARDWARE ATM 228
6.2.1.1 MPLS topológicamente administrado 229
6.2.1.2 MPLS para modo explícito 231
6.2.2 ATM EN LA DIFERENCIACIÓN DE SERVICIO 231
6.3 RSVP EXTENDIDO PARA MPLS 233
6.3.1 TÚNELES LSP 234
6.3.1.2 Mensaje RSVP *Path* 235
6.3.1.2.1 Objeto LABEL_REQUEST 235
6.3.1.2.2 Objeto EXPLICIT_ROUTE 236
6.3.1.2.3 Objeto RECORD_ROUTE 237
6.3.1.2.4 Objeto SESSION_ATTRIBUTE 237
6.3.1.2.5 Objeto FLOWSPEC 238
6.3.1.2.6 Nuevos Tipos de Objetos 238
6.3.1.3 Mensaje RSVP *Resv* 239
6.4 EJEMPLO DE ESTABLECIMIENTO DE UN TÚNEL LSP MEDIANTE RSVP EXT 240

CAPÍTULO 7

SISTEMAS DE POLÍTICAS DE CALIDAD DE SERVICIO

7.1 SISTEMAS DE POLÍTICAS TRADICIONALES 245
7.2 SISTEMA DE POLÍTICAS GENÉRICO 246
7.3 PROTOCOLOS DE POLÍTICAS 248
7.3.1 PROTOCOLO *RADIUS* 250
7.3.2 PROTOCOLO *DIAMETER* 251

7.3.3 PROTOCOLO COPS 252
7.4 INFRAESTRUCTURAS DE DIRECTORIOS 252
7.4.1 DIRECTORIOS 252
7.4.2 PROTOCOLOS DE ACCESO A DIRECTORIOS 255
7.4.2.1 Protocolo X.500 255
7.4.2.2 Protocolo LDAP (*Lightweight Directory Access Protocol*) 256
7.5 MODELO DEL SISTEMA DE POLÍTICAS 259
7.6 PROTOCOLO COMÚN DE SERVICIO DE POLÍTICAS COPS 261
7.6.1 INTRODUCCIÓN 261
7.6.2 MODELO Y FUNCIONAMIENTO 262
7.6.3 MENSAJES Y OBJETOS 265
7.6.4 MENSAJES 265
7.6.4.1 Requerimiento (REQ, *Request*) 265
7.6.4.2 Decisión (DEC, *Decision*) 265
7.6.4.3 Reporte del Estado (RPT, *Report State*) 266
7.6.4.4 Eliminación del Estado de Requerimiento (DRQ, *Delete Request State*) 266
7.6.4.5 Requerimiento de Estado de Sincronismo (SSQ, *Synchronize State Request*) .. 266
7.6.4.6 Abrir Cliente (OPN, *Client-Open*) 266
7.6.4.7 Aceptar Cliente (CAT, *Client-Accept*) 266
7.6.4.8 Cerrar Cliente (CC, *Client-Close*) 266
7.6.4.9 Mantener Vivo (KA, *Keep Alive*) 267
7.6.5 OBJETOS 267
7.6.5.1 Indicador (*Handle*) 267
7.6.5.2 Contexto (*Context*) 267
7.6.5.3 Interfaz de Entrada (*IN-Int*) 267
7.6.5.4 Interfaz de Salida (*OUT-Int*) 267
7.6.5.5 Razón (*Reason*) 268
7.6.5.6 Decisión (*Decision*) 268
7.6.5.7 Decisión LDPD (LPDP *Decision*) 268
7.6.5.8 Error (*Error*) 268
7.6.5.9 Información Específica del Cliente (*ClientSI*) 268
7.6.5.10 *Timer "Keep Alive"* (KA *Timer*) 268
7.6.5.11 Identificación del PEP (PEPID) 268

7.6.5.12 Tipo de Reporte (*Report-Type*) 269
7.6.5.13 Dirección del PDP Redireccionado (*PDP Redirect*) 269
7.6.5.14 Dirección del último PDP (*LastPDPAddr*) 269
7.6.5.15 Timer de Contabilidad (*AccTimer*) 269
7.6.6 UTILIZACIÓN DEL PROTOCOLO COPS EN UNA RESERVACIÓN RSVP. 269
7.7 CONSIDERACIONES ADICIONALES 271

CAPÍTULO 8

MONITOREO

8.1 INTRODUCCIÓN 273
8.2 PARÁMETROS A MONITOREAR 273
8.2.1 MONITOREO DEL RETARDO O LATENCIA 274
8.2.2 MONITOREO DEL JITTER 275
8.3 MONITOREO EN DIFERENTES PROTOCOLOS 276
8.3.1 PROTOCOLO TCP 276
8.3.2 PROTOCOLOS BASADOS EN RESERVACIONES 276
8.3.3 PROTOCOLO ATM 276
8.4 ACUERDOS DE NIVEL DE SERVICIO 278
8.5 MONITOREO CON HERRAMIENTAS TRADICIONALES 281
8.5.1 PING (PACKET INTERNET GROPER) 281
8.5.1.I Funcionamiento 281
8.5.1.2 Ejemplo de sesiones *PING* 282
8.5.1.3 *PING* Y QoS 290
8.5.1.4 Formato de Mensajes ICMP 291
8.5.1.4.1 Mensaje Eco 291
8.5.1.4.2 Timestamp y Respuesta al Timestamp 293
8.5.1.4.3 Mensaje Destino Inalcanzable 295
8.5.1.4.4 Mensaje Tiempo Cumplido 296
8.5.2 TRAZADO DE RUTA (*TRACEROUTE*) 296

8.5.2.1 Funcionamiento 297
8.5.2.2 Ejemplo de la utilización de *TRACEROUTE* 297
8.5.2.3 *TRACEROUTE* Y QoS 298

CONCLUSIONES Y RECOMENDACIONES **299**

ANEXOS

GLOSARIO

REFERENCIAS BIBLIOGRÁFICAS

ÍNDICE DE FIGURAS

CAPÍTULO 1

Figura 1.1	Proyección del número de usuarios de Internet, medido en millones	2
Figura 1.2	Tendencia del crecimiento de *hosts* en Internet	3
Figura 1.3	Red de negocios multicapa	9

CAPÍTULO 2

Figura 2.1	Capacidad vs. Costo de un enlace	33
Figura 2.2	Retardo teórico en el módem	42
Figura 2.3	Acuses de Recibo en una red X.25	44
Figura 2.4	Acuses de Recibo en redes Frame Relay	45
Figura 2.5	Flujo de datos y acuses de recibo en una sesión TCP	47
Figura 2.6	Funcionamiento del protocolo TCP	49
Figura 2.7	Tiempo de Respuesta de un Servidor perteneciente a un ISP	50
Figura 2.8	Tiempo de Respuesta de la Red	50
Figura 2.9	*Jitter*	51
Figura 2.10	Estructura de un cable coaxial	57
Figura 2.11	Estructura de una fibra óptica	58
Figura 2.12	Calidad de la voz vs. Tasa de pérdida de paquetes	64
Figura 2.13	Tramas Claves y Tramas Intermedias en una red descongestionada	67
Figura 2.14	Distribución del arribo de paquetes y retardo promedio en *buffers*	68
Figura 2.15	Recorte de la Tramas Clave cuando la capacidad es limitada	68

CAPÍTULO 3

Figura 3.1	Encolamiento FIFO	77

Figura 3.2 Funcionamiento del algoritmo de Encolamiento Justo Ponderado 80
Figura 3.3 Característica de Protección del algoritmo WFQ 83
Figura 3.4 Ejemplo de implementación WFQ en base a GPS y función virtual de tiempo 86
Figura 3.5 Representación del mecanismo CBQ 88
Figura 3.6 Campo ECN dentro de la cabecera IPv4 96

CAPÍTULO 4

Figura 4.1 Diagrama de un Dominio DiffServ 111
Figura 4.2 Diagrama de una Región DiffServ 112
Figura 4.3 Bloque Funcional de un *Router DiffServ* 114
Figura 4.4 Jerarquía de Componentes dentro de un *Router DiffServ* 115
Figura 4.5 Diagrama de un clasificador 116
Figura 4.6 Diagrama de un medidor genérico 119
Figura 4.7 Diagrama de un Bloque Acondicionador de Tráfico (TCB) 126
Figura 4.8 Diagrama a bloques de un TCB 128
Figura 4.9 Estructura del campo TOS 135
Figura 4.10 Estructura del Campo DS 136
Figura 4.11 DSCP asignado al EF PHB 143

CAPÍTULO 5

Figura 5.1 Modelo jerárquico de la compartición de recursos en un enlace 156
Figura 5.2 Modelo de implementación de referencia para los *routers* 162
Figura 5.3 Protocolo RSVP en los *hosts* y *routers* 182
Figura 5.4 Modelo de una reservación RSVP 183
Figura 5.5 Configuración del *router* 187
Figura 5.6 Ejemplo del estilo de reservación filtro comodín (WF) 188
Figura 5.7 Ejemplo del estilo de reservación filtro fijo (FF) 188
Figura 5.8 Ejemplo del estilo de reservación compartición explícita (SE) 188

Figura 5.9	Estado de bloqueo	195
Figura 5.10	Formato de la cabecera RSVP	199
Figura 5.11	Formato de los Objetos RSVP	200
Figura 5.12	Arquitectura de red IP/ATM que permite QoS	205
Figura 5.13	Funciones del Dispositivo Frontera	206
Figura 5.14	Formato de un fragmento utilizando PPP multienlace	212
Figura 5.15	Extensión corta al protocolo PPP multienlace	213
Figura 5.16	Extensión larga al protocolo PPP multienlace	213
Figura 5.17	Formato compacto de un fragmento del protocolo PPP orientado a tramas tipo HDLC	214
Figura 5.18	Formato Compacto Extendido de un fragmento del protocolo PPP orientado a tramas tipo HDLC	215

CAPÍTULO 6

Figura 6.1	Arquitectura Híbrida *IntServ/DiffServ*	223
Figura 6.2	Objeto DCLASS utilizado por RSVP	226
Figura 6.3	RSVP Estándar y RSVP Extendido	234
Figura 6.4	Formato del Objeto *SESSION_ATRIBUTE*	237
Figura 6.5	Procesamiento de un objeto LABEL por el LSR	240
Figura 6.6	Topología de la Red para el ejemplo	241
Figura 6.7	Asignación de Etiquetas en el ejemplo	244

CAPÍTULO 7

Figura 7.1	Sistema de Políticas Genérico	248
Figura 7.2	Modelo de Autentificación Tradicional	249
Figura 7.3	Modelo de Autentificación Dinámico	250
Figura 7.4	Estructura de un directorio jerárquico	254
Figura 7.5	Modelo del Sistema de Políticas	260

Figura 7.6 Modelo Básico del Protocolo COPS 262
Figura 7.7 Transacción COPS en una negociación RSVP 270

CAPÍTULO 8

Figura 8.1 Red de Acceso de Ramtelecom 284
Figura 8.2 Red de UUNET en Norte América 285
Figura 8.3 *PING* hacia el servidor de acceso de Ramtelecom 286
Figura 8.4 *PING* hacia un servidor de Ramtelecom en Miami 287
Figura 8.5 Histograma de los Tiempos de Respuesta 289
Figura 8.6 Histograma de la diferencia de los Tiempos de Respuesta 290
Figura 8.7 Formato del paquete *PING* 291
Figura 8.8 Mensaje *Timestamp* 294
Figura 8.9 Formato del mensaje Destino Inalcanzable 295
Figura 8.10 *TRACEROUTE* desde Ramtelecom hacia Net2Phone 298

ÍNDICE DE TABLAS

CAPÍTULO 1

CAPÍTULO 2

Tabla 2.1	Retardo para varios medios y dispositivos Ethernet	36
Tabla 2.2	Velocidades de conexión a Internet vía módem	39
Tabla 2.3	Tipos de acceso doméstico a Internet	39
Tabla 2.4	Tasas de transmisión de varias aplicaciones	59
Tabla 2.5	Tipos de Aplicaciones y Requerimientos de la Red	62
Tabla 2.6	Comparación de varias conexiones de voz	65

CAPÍTULO 3

CAPÍTULO 4

Tabla 4.1	Valores de Precedencia definidos para el campo TOS	135
Tabla 4.2	Definición de los bits D, T y R	136
Tabla 4.3	Valores DSCP para PHBs AF	145

CAPÍTULO 5

Tabla 5.1	Exponente en los números de punto flotante	175
Tabla 5.2	Estilos de Reservación	185
Tabla 5.3	Valor del campo Tipo de Mensaje para los mensajes RSVP	199
Tabla 5.4	Parámetros ATM	203

ANEXOS

ANEXO 1

NIVELES ACEPTABLES E INACEPTABLES DE RETARDO, JITTER Y PÉRDIDA DE PAQUETES PARA TRÁFICO CONVERSACIONAL

ANEXO 2

ENCOLAMIENTO JUSTO PONDERADO: RENDIMIENTO

ANEXO 3

CÁLCULO DE LA LONGITUD PROMEDIO DE LA COLA

ANEXO 4

ALGORÍTMOS DE CUBETA CON GOTEO (*LEAKY BUCKET*) Y DE CUBETA CON FICHAS (*TOKEN BUCKET*)

ANEXO 5

FORMATO DE LOS MENSAJES RSVP

ANEXO 6

MPLS (*MultiProtocol Label Switching*)

RESUMEN

El presente Proyecto de Titulación trata sobre un estudio de la Calidad de Servicio en Internet, un concepto moderno que está en auge en la red de redes, como tecnología para la provisión de diferenciación de niveles de servicio y por ende, diferenciación de costo.

La provisión de una infraestructura de este tipo, implica una optimización efectiva de los recursos de la red, pues la filosofía fundamental de una red con Calidad de Servicio es administrar adecuadamente sus recursos antes que proponer la expansión física de los mismos.

Una administración efectiva de los recursos implica comprender la influencia de éstos en los diferentes tipos de aplicaciones. Para ello, se realiza un estudio de aplicaciones críticas, en tiempo real y de los parámetros técnicos que afectan su rendimiento.

Innumerables estudios y desarrollos, han aparecido en los últimos tiempos, todos ellos basados en dos prominentes arquitecturas: la Arquitectura de Servicios Diferenciados y la Arquitectura de Servicios Integrados, las cuales, a través de mecanismos de gestión de tráfico y sólidos sistemas de administración. implementan una infraestructura de Calidad de Servicio en Internet.

La Arquitectura de Servicios Diferenciados está diseñada en base a un sencillo modelo de provisión de servicios, bajo el cual se clasifica al tráfico en pocos grupos, cada uno de los cuales obtiene un tratamiento diferenciado de la red.

La Arquitectura de Servicios Integrados trabaja bajo un modelo de reservación de recursos en los elementos de red, que le permite ofrecer garantías de servicio extremo a extremo a aplicaciones con estrictos requerimientos de recursos.

Cada arquitectura señalada presenta ventajas y desventajas, motivando la creación de una Arquitectura Híbrida que aproveche las bondades de las Arquitecturas de Servicios Diferenciados e Integrados: escalabilidad y garantía de servicio, respectivamente.

La parte final del Proyecto presenta la utilización de herramientas que permiten verificar los niveles de servicio recibidos por un usuario que accede a Internet por medio de la red de su proveedor de servicios.

PRESENTACIÓN

Uno de los factores de mayor éxito del Internet actual es la aceptación de los protocolos TCP/IP como estándar de *facto* para todo tipo de servicios y aplicaciones. El Internet ha desplazado a las tradicionales redes de datos y ha llegado a ser el modelo de red pública del siglo XXI.

Si bien es cierto que Internet puede llegar a consolidarse como el modelo de red pública de datos a gran escala, también es verdad que actualmente no satisface todos los requerimientos de los usuarios, especialmente de los usuarios corporativos, los cuales necesitan la red para el soporte de aplicaciones críticas.

Una de las principales carencias de Internet es la imposibilidad de seleccionar niveles de servicio adecuados a los distintos tipos de aplicaciones. En la actualidad se trata de salvar esta falencia implementando mecanismos de configuración y manipulación de tráfico, sólidos sistemas de administración y adecuadas técnicas de monitoreo, con el objetivo de implantar una infraestructura de Calidad de Servicio en Internet.

Ésta es la razón que motivó el desarrollo del presente Proyecto de Titulación, en el cual se han planteado los siguientes objetivos:

- La identificación de los parámetros técnicos que definen la Calidad de Servicio, así como un análisis de cada uno de ellos.
- El estudio y conocimiento de modelos y arquitecturas que permitan implementar Calidad de Servicio en Internet.
- Conocer la forma de administrar una red que incorpore Calidad de Servicio.

A fin de conseguir el cumplimiento de estos objetivos se ha organizado el presente estudio en ocho capítulos

- El primer capítulo define la problemática y justifica la necesidad de un cambio de carácter en Internet. En su parte final otorga una idea general del tema, mediante una descripción .de los principales elementos que conforman un ambiente QoS.

- El segundo capítulo estudia los parámetros técnicos que definen la Calidad de Servicio en Internet, y su influencia en el rendimiento de diferentes tipos de aplicaciones. Además, se realiza un estudio de los requerimientos de recursos de aplicaciones de voz, audio y video.

- El tercer capítulo vincula a los mecanismos de gestión de tráfico, como entes primarios en la consecución de Calidad de Servicio en una red. Específicamente se estudia algoritmos de encolamiento, notificación de congestión y descarte de paquetes cuya interacción faculta el desarrollo de una administración activa de colas.

- El cuarto capítulo estudia y analiza la Arquitectura de Servicios Diferenciados como una sencilla y escalable solución de Calidad de Servicio.

- El quinto capítulo estudia y analiza la Arquitectura de Servicios Integrados como segunda solución posible de implementación de Calidad de Servicio. Se analiza el moderno protocolo de reservación de recursos RSVP. Finalmente se describe la implementación de esta arquitectura sobre tecnologías específicas de capa enlace: ATM y enlaces lentos.

- El sexto capítulo estudia varias Arquitecturas Híbridas: una de ellas formada por la combinación de las arquitecturas analizadas en los capítulos 4 y 5; y la otra, en base a la interacción de éstas con el protocolo MPLS.

- El séptimo capítulo tiene que ver con la administración de una red que incorpora Calidad de Servicio. Se presenta el modelo de políticas de QoS y se profundiza en el estudio de su protocolo característico (COPS). Además se

estudia las infraestructuras de directorios y el protocolo de accceso a ellos (LDAP).

- El octavo capítulo versa sobre técnicas que un usuario puede aplicar a fin de probar el estado de la red. Se analizan herramientas tradicionales, ilustrándose su utilización mediante un ejemplo real.

Para un enfoque minucioso de ciertos tópicos se ha incorporado 6 anexos en los cuales se podrá encontrar información útil si se desea profundizar en algún determinado tema. Este trabajo cuenta también con un glosario de términos y sus definiciones para lograr un mejor entendimiento del texto contenido.

Con este esquema se pretende que el lector se forme una idea clara sobre los siguientes aspectos:

- Los requerimientos de aplicaciones en tiempo real, tales como audio y video.
- Cambios a efectuarse en la actual ingenieria de red a fin de implementar la infraestructura de Calidad de Servicio.
- Conocer técnicas que permitan monitorear los niveles de servicio en un ambiente QoS.

Este trabajo ha sido fruto de un gran esfuerzo y se aspira que sea de mucha utilidad.

Capítulo 1

MOTIVACIONES PARA LA PROVISIÓN DE CALIDAD DE SERVICIO

CAPÍTULO 1

MOTIVACIONES PARA LA PROVISIÓN DE CALIDAD DE SERVICIO

La evolución de las redes IP, como el Internet, se puede comparar a la de cualquier sistema u organización, al principio únicamente se piensa en lo básico e imprescindible, pero cuando el sistema se asienta surge la necesidad de ir añadiendo nuevos valores, como la optimización, la eficiencia, y economía en el uso de recursos, la predictibilidad de los sistemas, etc.; es decir la calidad en el servicio.

Así, cuando se empezó a usar el conjunto de protocolos TCP/IP (*Transport Control Protocol / Internet Protocol*), base de Internet y de las redes IP en general, la preocupación fundamental fue encontrar un sistema muy abierto que constituyese un lenguaje común para un entorno heterogéneo de interconexión del que no se vislumbraba la magnitud de su alcance en ninguna de sus vertientes.

Por eso se primó la generalidad por sobre la eficiencia creando un sistema de comunicaciones que no es, desde luego, el más efectivo. Seguridad, calidad, e incluso eficiencia no fueron objetivos contemplados.

1.1 EL MODELO *BEST EFFORT*

El Internet actual consiste de una multitud de redes construidas por varias tecnologías de capa de enlace que confían en el Protocolo Internet (IP) para interconectarse entre ellas. El protocolo IP no se preocupa sobre los protocolos de capas inferiores en el *stack*[1], y ofrece un servicio no confiable, no orientado a conexión en la capa de red, que está sujeto a pérdida, reordenamiento y duplicación de paquetes, todo esto, junto con el retardo de encolamiento en los

[1] Stack – Se refiere a la pila de protocolos de algún modelo de referencia

buffers de los *routers*, incrementan la carga de la red. Debido a todos estos factores, el tradicional modelo de envío IP es a menudo conocido como *best effort*, complementado con un protocolo adicional de una capa más alta como es el Protocolo de Control de Transmisión (TCP), requerido para proveer confiabilidad extremo a extremo. TCP hace esto a través de mecanismos como retransmisión de paquetes, lo cual añade más retardo en la transferencia.

Este modelo permite que la complejidad del Internet se sitúe en los extremos (*hosts*) finales, posibilitando que el núcleo de la red se mantenga relativamente simple, cualidad evidenciada en la habilidad del Internet para soportar un crecimiento fenomenal. Las figuras 1.1. y 1.2 muestran el crecimiento de los usuarios y los *hosts* respectivamente.

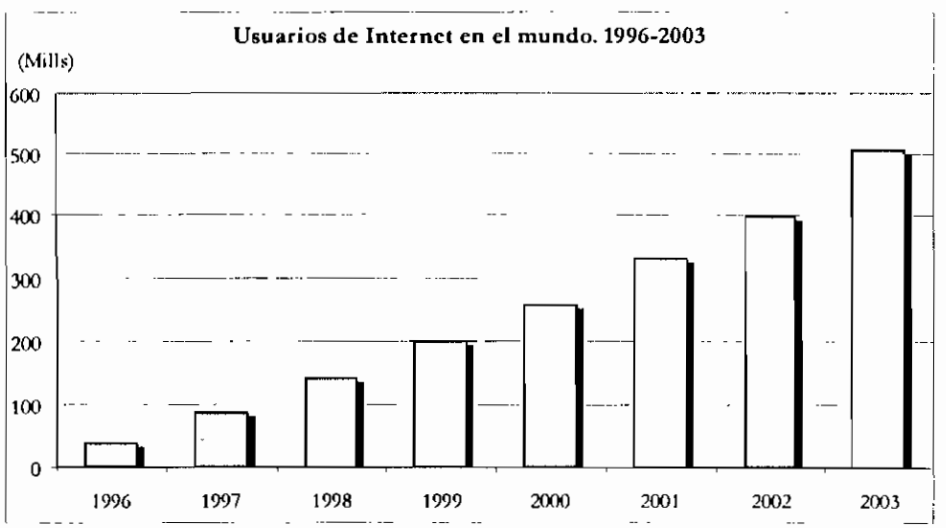

Figura 1.1 Proyección del número de usuarios de Internet, medido en millones. [1]

Se estima que la tasa anual de incremento en la demanda fluctúa entre el 300 y el 700 por ciento. Después de todo, en alrededor de 25 años el Internet se ha expandido de aproximadamente 100 *hosts* o nodos de la Arpanet[1] a cerca de 60 millones de nodos actualmente. [2]

Mientras más *hosts* se conecten, la demanda de servicios de la red eventualmente excede su capacidad. Pero el servicio nunca es negado, en lugar de ello se

[1] Arpanet.- Nombre asignado inicialmente al Internet.

degrada significativamente, presentando variabilidad en el retardo (*Jitter*), y pérdida de paquetes, que no afectan a las aplicaciones tradicionales de Internet –correo electrónico, transferencia de archivos y aplicaciones Web- pero otras aplicaciones no pueden adaptarse a niveles de servicio inconsistentes. Los retardos en la transferencia causan problemas en aplicaciones con requerimientos de tiempo real, como multimedia y telefonía.

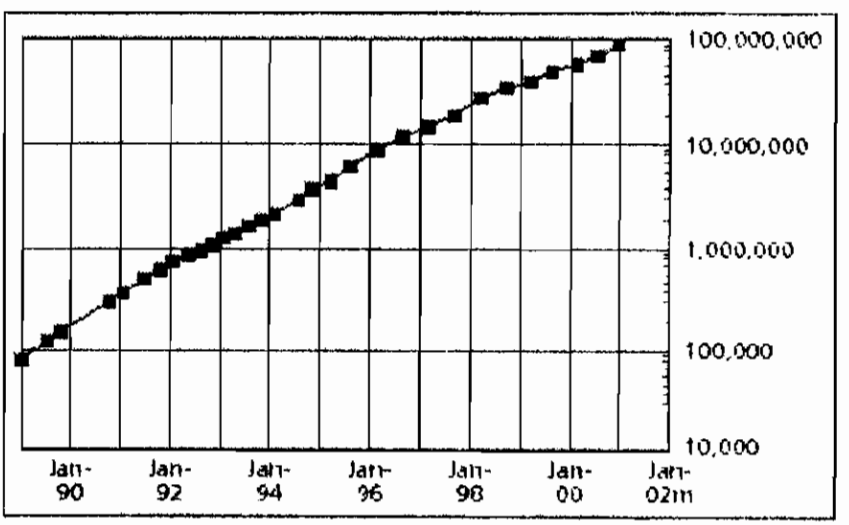

Figura 1.2 Tendencia del crecimiento de *hosts* en Internet. [3]

En casi todos los casos, el Internet actual y todas las redes privadas basadas en la tecnología Internet, tratan los paquetes exactamente igual. Cuando un paquete avanza a través de la red, recibe el mismo tratamiento sin importar si es parte de una transferencia de archivos de respaldo, una solicitud a o una respuesta del servidor Web, un segmento de audio de una llamada telefónica sobre IP, o parte de una videoconferencia. Además, en una red pública compartida, un paquete es tratado igual sin importar cuánto pague el usuario por el servicio de Internet.

Si las aplicaciones en Internet fueran homogéneas, esto no podría tener gran consecuencia. Por ejemplo, si todo el tráfico de la red fuera telefonía a una tasa de transferencia constante, luego el diseñador de la red podría saber que cada llamada requiere 64 kbps de capacidad [2], y el camino dado en la red será diseñado para manejar un máximo número de llamadas. Multiplicando el número de

llamadas *N*, por los 64 kbps requeridos por cada llamada, se determinaría el ancho de banda digital del camino.

Cada llamada podría luego simplemente enviar paquetes sumando hasta 64 kbps en tiempo real. (La ausencia de transmisión de los paquetes de silencio es desde luego una optimización disponible, y la telefonía puede adaptarse a menor velocidad reduciendo la calidad de la voz. Pero esto no cambia el argumento esencial ya que debe haber una máxima tasa de transmisión por llamada en promedio.)

Con un sistema homogéneo como el anteriormente descrito, sería fácil calcular la capacidad necesaria. Además, ya que los paquetes serán transmitidos inmediatamente, en tiempo real, la longitud eléctrica conocida del camino hace fácil calcular el retardo máximo que cualquier paquete de voz podría sufrir. Si los cálculos fueran imprecisos, pocos paquetes de voz podrían perderse; pero esto es aceptable porque las señales de voz contienen una gran cantidad de información de poca relevancia.

La situación sería diferente si todo el tráfico fuera transferencia de archivos. El número total de transferencias, la velocidad a la cual los discos en cada punto final podrían aceptar datos, y el tamaño de los archivos a ser transferidos serían desconocidos. Aún cuando ninguna transferencia individual sería especialmente urgente, algunos paquetes perdidos tendrían que ser retransmitidos para evitar el daño del archivo. En este caso la transferencia del archivo podría correr sobre el Protocolo de Control de Transmisión (TCP), el cual tiene dos características importantes. Primero, algunos paquetes perdidos debido a la congestión son retransmitidos en el curso correcto. Segundo, censando la pérdida de paquetes por la congestión, el protocolo TCP, baja la velocidad de envío de cada paquete en una transferencia de archivo en curso. Estas dos características le permiten adaptarse a las condiciones presentes en la red, aliviando la congestión.

Ahora considerar lo que sucede cuando estos dos tipos de aplicaciones (telefonía y transferencia de archivos) son mezclados en el mismo segmento de la red. Sin importar la congestión del tráfico y la pérdida de paquetes, el sistema telefónico

diseñado preservará la tasa de transmisión a 64 kbps para cada una de las *N* llamadas. Entretanto, el tráfico de transferencia de archivos usando el mismo camino, congestión y pérdida de paquetes, disminuirá su velocidad en un intento de compartir la capacidad disponible. Como el tráfico telefónico no puede disminuir la velocidad, tratará de obtener más de lo que la red puede ofrecer, causando así que se incrementen las pérdidas y por tanto se obtiene una pobre calidad de voz, pues el tráfico de la transferencia de archivo se encuentra todavía en el camino, aunque a una menor velocidad. En general si el tráfico con capacidad de adaptarse a la congestión, es mezclado con el tráfico de tiempo real, ambos lados pierden.

1.2 CAMBIO DE CARÁCTER EN EL INTERNET ACTUAL

A más del uso de la voz sobre circuitos de datos, otros desarrollos recientes de tecnología de redes y telecomunicaciones están conduciendo a la necesidad de aprovisionar calidad de servicio. Estos incluyen aplicaciones en tiempo real como videoconferencias, tecnologías de empaquetamiento, las bondades de equipos inteligentes para redes, el despliegue de aplicaciones críticas sobre el modelo *best effort*, y el deseo de reducción de costos.

1.2.1 NUEVOS SERVICIOS Y APLICACIONES

1.2.1.1 Voz sobre circuitos de datos

Si las actuales tendencias continúan, y al parecer no hay razón para que cambien, el tráfico de datos sobrepasará al tráfico de voz sobre redes de larga distancia alrededor del año 2002 [4]. Esta tendencia también se ve afectada por el surgimiento de servicios de portadores de larga distancia que despliegan soluciones de voz sobre el protocolo IP (VoIP, *Voice over IP*).

El uso de VoIP en lugar del actual sistema telefónico convencional ofrece ciertas ventajas:

- A diferencia de la Red Telefónica Pública Conmutada (PSTN, *Public Switched Telephone Network*), la cual usa tecnología de conmutación de circuitos, la telefonía sobre IP usa tecnología de conmutación de paquetes. En la conmutación de paquetes no se dedica un enlace de comunicación

solo a llamadas de voz, todas las aplicaciones comparten los recursos de la red, disminuyendo significativamente los costos de una llamada de voz.

- La red telefónica convencional PSTN puede soportar una sola calidad de sonido, la misma que utiliza un ancho de banda de 64 kbps. En cambio, en una red IP se pueden manejar diferentes grados de calidad de sonido, tales como estéreo de alta fidelidad o sonido envolvente (*surround*), si existe el suficiente ancho de banda.

- Los servicios de datos pueden fácilmente ser combinados con servicios de voz para crear nuevas aplicaciones las cuales no son posibles con la telefonía convencional.

- Debido a que el tráfico de voz puede ser cursado por redes de datos, empresas con sedes en diferentes lugares pueden integrar su red telefónica existente con su red de comunicación de datos para obtener un significativo ahorro de dinero. El ahorro se debe a que las empresas deben mantener una sola red que soporta voz, datos y fax.

- A pesar de la utilización de dispositivos inteligentes en el núcleo de una red PSTN, la limitada funcionalidad de los aparatos telefónicos no ha permitido la provisión de servicios al usuario final. Por el contrario, con un computador se puede tener acceso a servicios agregados como por ejemplo la información del consumo en tiempo real.

A continuación se detallan algunas aplicaciones que tienen como base el tráfico de voz complementado con servicios de datos.

MENSAJERÍA UNIFICADA

La mayoría de gente tiene dirección electrónica, número de teléfono móvil, número de teléfono fijo y número de fax donde puede ser localizada. Un sistema de conmutación de paquetes ofrece la unificación de todos los servicios de mensajería descritos. Un usuario puede obtener todos sus mensajes enviados a un determinado lugar al instante en que él crea conveniente. Un correo de voz, una llamada telefónica o un e-mail pueden ser desviados a una misma localización.

Esta característica permite en un futuro la asignación de un simple número telefónico para todos los servicios de comunicación.

CALL CENTERS BASADOS EN EL WEB

Este tipo de servicio permite que un usuario que está navegando en Internet establezca una llamada de VoIP desde el sitio web de una organización hacia el *call center* de la misma. El usuario no necesita terminar la navegación, simplemente se establece una sesión adicional. Este servicio además de ayudar a captar la atención de potenciales clientes, permite proveer información suplementaria en caso que el cliente lo desee.

1.2.1.2 Servicios de Videoconferencia

La unión de los ordenadores con las redes de comunicaciones permite crear servicios que facilitan enormemente la colaboración remota. En los últimos años se ha trabajado mucho en técnicas de colaboración asíncronas, alcanzándose un grado de efectividad muy grande y permitiendo automatizar la comunicación de grupos y los flujos de trabajo con gran eficiencia. Su uso se está generalizando, tanto en el mundo académico, en el mundo empresarial e incluso entre la sociedad en general, debido al aumento de productividad que generan.

Esto está creando una continua demanda de nuevos servicios telemáticos, entre los que destacan aquellos que proporcionan capacidades de colaboración interactiva en tiempo real, desde la videoconferencia básica hasta servicios más sofisticados de Tele-educación, Tele-reunión, o trabajo colaborativo. Las aplicaciones básicas de transferencia de datos, audio y video, se complementan con herramientas como pizarras, editores y cualquier otro tipo de aplicación distribuida. También se aprecia la introducción de nuevas facilidades multimedia como las técnicas holográficas, visión en tres dimensiones (3D) o realidad virtual. Aunque las redes actuales no están preparadas para soportar los servicios que se están experimentando, el mercado potencial es enorme.

El gran reto que el mundo de la creación de servicios tiene planteado actualmente es dar soporte efectivo a la colaboración interactiva en tiempo real. Para ello se han desarrollado en los últimos años las técnicas de videoconferencia y la compartición

de espacios de trabajo en tiempo real. Las tecnologías están ahí pero todavía no han madurado suficiente como para soportar colaboraciones remotas interactivas de forma eficaz.

Para el usuario el acceso al servicio debe ser muy simple, sin embargo, desde el punto de vista de la red la situación es muy distinta. Aunque la tecnología de red no depende solo de aspectos funcionales, como el costo de introducción de una nueva infraestructura, o los costos de integración, cableado, interconexión y gestión; los aspectos más determinantes son: ancho de banda, capacidad de comunicación multienvío, retardo, confiabilidad y capacidad de sincronización de canales.

Hace algunos años se aseguraba que los servicios multimedia interactivos tardarían en imponerse. Entonces el objetivo consistía en construir un sistema de video de alta calidad que necesitaba un canal con una capacidad de 100 Mbps [5]. En la actualidad las cosas han cambiado. Por un lado están todos los desarrollos asociados a las normas de videoconferencia llevadas a cabo por ITU-T (*International Telecommunication Union – Telecommunication*) y otros organismos que trabajan de manera coordinada como IMTC (*International Multimedia Teleconferencing Consortium*). Estas normas soportan la multiconferencia de video, audio, y datos.

1.2.1.3 Red de Negocios de Próxima Generación

La estructura de la red, desde la arquitectura y las tecnologías fundamentales a las configuraciones de los dispositivos por si mismos, será adaptada a procesos de negocios que manejen perfiles de usuario. Se dará una migración hacia la convergencia y los usuarios serán libres de desplegar sus propias aplicaciones dentro de guías robustas basándose en el trabajo de grupo.

La figura 1.3 muestra una red donde la capacidad es dividida en múltiples capas discretas, siendo la capa superior reservable para usuarios y aplicaciones calificadas.

El tráfico de negocios vendrá primero, y la red incluirá soporte para usuarios móviles a través de enlaces *dial-up* o accesos de Red Privada Virtual (VPN, *Virtual*

Private Network). Para mantener todo funcionando adecuadamente se deberá tener seguridad basada en certificados digitales, y por tanto la administración de la red se focalizará en aplicaciones y usuarios.

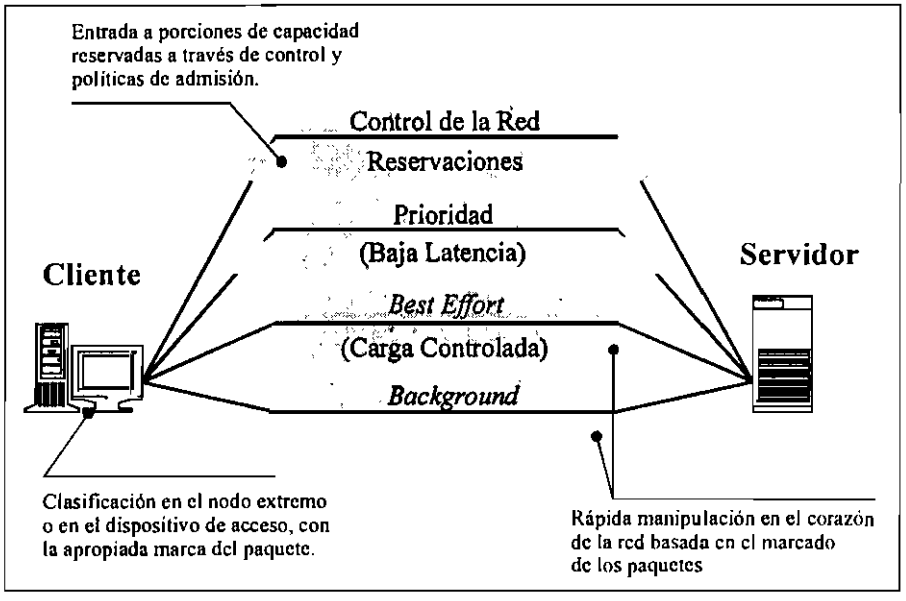

Figura 1.3 Red de negocios multicapa. [6]

Se verá también un mayor cambio de interfaces y direcciones a usuarios y aplicaciones. Este cambio será alimentado por el incremento de sitios de portal[1] como plataformas de aplicación en lugar de únicamente enlaces, así como también por el despliegue de sistemas de configuración dinámicos que hagan posible la movilidad "*plug-and-play*"[2].

1.2.2 DESARROLLO DE HARDWARE Y SOFTWARE

1.2.2.1 Desarrollo de los computadores

Los ordenadores modernos ofrecen gran capacidad de cómputo y periféricos para soportar animación, reproducción de audio y digitalización de voz. Aplicaciones interactivas y conferencias *full-duplex*[3] son parte de muchos sistemas operativos,

[1] Portal.- Un sitio web que ofrece un amplio arreglo de recursos y servicios, la mayoría de los cuales son en línea, como correo electrónico, foros, máquinas de búsquedas y comercio electrónico.

[2] Movilidad Plug and Play.- Facilidad de conectarse a una red desde cualquier lugar, gracias a la tendencia actual de administrar una red en base a usuarios y aplicaciones en lugar de direcciones IP.

[3] Comunicación full-duplex.- Capacidad de un sistema de comunicación que le permite enviar y recibir datos al mismo tiempo.

además ciertas utilidades presentes en los navegadores permiten a los usuarios manejar diferentes formatos multimedia.

Estas nuevas plataformas de hardware, aplicaciones y características de los sistemas operativos incrementan la demanda de capacidad, ya que generan más tráfico y hacen más perceptible a los usuarios el estado de la red. Un retardo en la transferencia de archivos es difícil de notar, en cambio un retardo en una trama de video es fácilmente detectable. El crecimiento de la demanda y la conducta visible de la red están conduciendo al despliegue de equipo inteligente.

1.2.2.2 Desarrollo de la tecnología de conectividad

Los actuales equipos de redes pueden ofrecer servicios de clasificación o reservación de ancho de banda para un rendimiento garantizado en el envío de tráfico. El software puede simular múltiples redes concurrentes a través de un enlace físico simple, y hacerlo en una forma que optimice el ancho de banda disponible asignando la capacidad excedente al tráfico menos importante y otorgando la capacidad apropiada a las aplicaciones sensitivas al retardo. Tráfico que no es adecuado para mezclarse, tales como tráfico en tiempo real y *store-and-forward*[1], tráfico de gran volumen a velocidad constante y ráfagas de paquetes pequeños, coexisten en enlaces lógicos diferenciados que tal vez no lo harían en un entorno *best effort* puro.

Estos equipos también pueden ofrecer diferenciación. Los dispositivos son diferentemente adecuados para tráfico en una red convergente. Un *switch*[2] tradicional de telefonía cursa muy bien varios flujos de datos sincrónicos, pero lo hace a un alto costo por bit. Por otro lado, un *hub*[3] Ethernet acarrea pobremente múltiples flujos de tráfico sensitivo al tiempo, pero es extremadamente económico en términos de precio por la capacidad obtenida.

A pesar de la introducción de equipos inteligentes de conectividad, la demanda de ancho de banda continua siendo enorme y la introducción de nuevos servicios de

---

1 Tráfico store and forward - Tráfico que puede ser almacenado y enviado posteriormente debido a que no es sensible al retardo

2 *Switch* - Dispositivo de interconectividad de redes que opera al nivel de capa enlace o capa 2 del modelo de referencia OSI

3 *Hub* - Dispositivo de red en el cual convergen datos desde, y se envían datos hacia, varias direcciones

colaboración en tiempo real exige muchos más recursos que los servicios de datos tradicionales. Actualmente se están utilizando tecnologías de transmisión de altas velocidades, las cuales están en el orden de los megabits por segundo como ATM (*Asynchronus Transfer Mode*), de los gigabits por segundo como SONET (*Synchronus Optical NETwork*) e incluso de los terabits por segundo como WDM (*Wavelength División Multiplexing*). [10]

Las tecnologías orientadas a conexión como ATM y Frame Relay son más adecuadas para transportar tráfico dependiente del tiempo como la voz; los nodos finales solo necesitan digitalizarlo y conmutarlo a un circuito, operando en una forma más determinística. Por ejemplo, ATM define cómo manejar tráfico de voz en detalle. Varias marcas ofrecen soluciones de voz sobre ATM que posibilitan capacidades de la voz, tales como cancelación de eco y compresión de silencio.

Muchos Dispositivos de Acceso Frame Relay (FRADs, *Frame Relay Access Devices*) también ofrecen características de voz. A nivel mundial, el despliegue de esta tecnología es limitado con relación a ATM, a diferencia de lo que sucede en Ecuador, donde Frame Relay es el protocolo WAN más utilizado.

Por otro lado, nuevas técnicas de acceso tornan aún mas crítica la situación del actual Internet, debido a sus altas velocidades de transmisión. Ejemplos representativos son las líneas digitales de abonado, los cable módems, y la difusión directa desde el satélite. Estos sistemas evitan los tradicionales *switches* PBX (*Private Branch eXchange*) usados para cursar tráfico de voz, superando las limitaciones de la actual infraestructura de voz de conmutación de circuitos.

Las Líneas Digitales de Abonado (DSL, *Digital Suscriber Lines*) toman ventaja de las frecuencias no utilizadas del par de cobre telefónico existente para transportar información a tasas muy superiores a las que permiten los actuales módems. Un módem ADSL (*Asymetric Digital Suscriber Line*) puede manejar tasas de transmisión de 1.554 a 8.448 Mbps para el enlace desde el Internet hacia el abonado y de 16 a 640 kbps para el enlace desde el abonado hacia el Internet. [7]

Los cable módems son una alternativa al par de cobre. Éstos pueden manejar tasas de transmisión de hasta 10 Mbps para el enlace desde el Internet hacia el

abonado y de 200 kbps a 2 Mbps [40] para el enlace desde el abonado hacia el Internet.

Los sistemas de difusión satelitales permiten una alta tasa de transmisión desde el Internet hacia el usuario final. Para ello se utiliza antenas satelitales en el extremo del cliente. El enlace desde el usuario hacia el Internet utiliza enlaces de módems. DirecPC es una empresa que ofrece este servicio, el cual permite alcanzar velocidades de transmisión de 400 kbps en el enlace *downstream*[1]; mientras el enlace upstream[2] utiliza módems *dial-up* o líneas ISDN (*Integrated Services Digital Network*).

El bajo costo de los equipos, la amplia disponibilidad y la gran innovación tecnológica en el mundo de los datos, no son los únicos factores que conducen a la convergencia. El alto costo de mantenimiento de redes paralelas independientes, una para voz, una para video y una para datos, y la administración de las interfaces entre ellas hacen que las corporaciones no puedan mantener por siempre este sistema.

1.3 ADMINISTRACIÓN DEL ANCHO DE BANDA

Tradicionalmente los problemas de congestión se han solucionado incrementando el ancho de banda en los segmentos de red que lo requieran, pues lo normal es pensar que cuando las conexiones "van lentas", la única solución posible es contratar más capacidad para las mismas. Pero cuando se hace esto, las líneas vuelven a quedar escasas tras un breve período de tiempo. Para entender si esto es correcto se puede hacer una analogía fácil de entender: si se amplía la carretera de salida de una ciudad se puede circular de una manera más desahogada, pero debido al coste de las obras primero se analizaría qué hacer para ir mejor sin realizar esta ampliación. La respuesta será racionalizar el tráfico, intentando que todos los vehículos circulen más rápido aprovechando mejor la carretera actual y además que los más importantes lo hagan mejor que los menos importantes.

---

[1] Downstream - Flujo de comunicaciones desde la red (Internet) hacia el cliente

[2] Upstream - Flujo de comunicaciones desde el cliente hacia la red (Internet

Esto mismo se puede hacer con el ancho de banda de la red, administrar mejor este muy caro recurso. La administración del ancho de banda es la asignación de la capacidad de transporte apropiada dentro de una red para cada usuario y aplicación.

"La capacidad apropiada" significa la velocidad de transmisión correcta, el retardo adecuado y el correcto nivel de cambio en el retardo. Ofrecer a cada aplicación un retardo extremo a extremo de 10 milisegundos es una gran idea, pero "la capacidad apropiada" significa también optimizar el costo. Si los respaldos pueden correr en la noche sobre enlaces lentos, usar un enlace de video de alto rendimiento para ellos es gastar el dinero.

"Dentro de la red" significa que una red tiene servicios diferenciados que ofrecer. Si una red va a proveer más de una clase de servicio, debe ser capaz de identificar diferentes tipos de tráfico y manipularlos en diferentes formas. La clasificación y la manipulación pueden resultar en una diferenciación muy general, como el uso de dos circuitos con diferentes latencias o puede ser muy especifica como una negociación dinámica del *throughput*[1] y el retardo en una sesión individual.

"Para cada usuario y aplicación" implica que la red tiene sistemas de identificación basados en usuarios y aplicaciones. Los actuales dispositivos de interconectividad trabajan con direcciones e interfaces, pero una red con un ancho de banda administrado necesita determinar información de usuarios y aplicaciones desde la información de direccionamiento usando una variedad de servicios de red tales como las autentificaciones basadas en usuario. Esto representa un cambio significativo lejos de una perspectiva de direccionamiento en bajo nivel y más bien migra hacia una red basada en usuarios y servicios, empezando la infraestructura de interconectividad a ofrecer Políticas de Administración. [6]

Una red con ancho de banda administrado es capaz de ofrecer distintas características basadas en el tipo de tráfico que maneje. El tráfico puede ser clasificado por aplicación, usuario, o aún por factores externos como hora del día,

---

[1] Throughput - En transmision de datos, el throughput es la cantidad de datos que se mueven de un lugar a otro en un periodo de tiempo dado

fecha o nivel de congestión de la red. Se puede construir una red capaz de ofrecer diferentes grados de servicio, desplegar un sistema de directorios y crear puntos de autentificación que gobernarán la asignación de estos servicios.

La terminología para la administración del ancho de banda es compleja. Hubo inicialmente dos grandes campos para la administración del ancho de banda: Clase de Servicio (COS, *Class Of Service*) y Calidad de Servicio (QoS, *Quality of Service*). COS divide a la red en pocas categorías discretas de servicio; QoS, por otro lado permite la negociación dinámica de servicios de red específicos a través de la reservación de recursos. Los dos campos son mejores que el actual modelo *best effort* pues permiten asignar al tráfico diferentes niveles de servicio.

Una red basada en COS, con pocas clases discretas de servicio es mucho más simple de crear que un sistema completo QoS. Un pequeño número de clases de servicio puede no ser suficiente para las diferentes aplicaciones que aparecerán en los próximos años, cada una de ellas con diferentes necesidades. Por otro lado, incrementando el número de clases se perjudica la simplicidad del modelo COS.

Actualmente el término QoS se utiliza para referirse a los dos campos mencionados por lo que a lo largo del desarrollo del presente proyecto de titulación se mantendrá este criterio.

1.4 CALIDAD DE SERVICIO: CONCEPTOS Y DEFINICIONES

QoS es la habilidad de un elemento de red (por ejemplo, aplicación, *host* o *router*) de tener algún nivel de seguridad para que su tráfico y requerimientos de servicio puedan ser satisfechos. Habilitar QoS requiere la cooperación de todas las capas de red desde la más alta a la más baja, así como también de cada elemento de red de extremo a extremo.

QoS es un conjunto de tecnologías que permite a los administradores de la red manejar los efectos de congestión del tráfico de las aplicaciones usando óptimamente los recursos de red, en lugar de añadir capacidad continuamente.

QoS no crea ancho de banda. No es posible para la red dar lo que no tiene, pero la disponibilidad de ancho de banda es el punto de inicio. QoS solo administra el ancho de banda de acuerdo a la demanda de la aplicación y la configuración de la administración de la red.

Las redes están construidas por la concatenación de dispositivos de red como *routers* y *switches*. Éstos direccionan el tráfico entre ellos mismos usando interfaces. La congestión ocurre si la tasa a la cual el tráfico arriba a una interfaz excede la tasa a la cual la interfaz puede direccionar tráfico al siguiente dispositivo. Por tanto la capacidad de una interfaz para direccionar tráfico es un recurso de red fundamental. Los mecanismos de QoS trabajan distribuyendo este recurso preferentemente a cierto tráfico sobre otros.

Para hacerlo, es necesario identificar primero diferentes clases de tráfico. El tráfico entrante al dispositivo de red es separado en distintos flujos por medio del proceso de clasificación de paquetes. El tráfico de cada flujo es luego dirigido a la correspondiente cola de la interfaz correspondiente. Las colas en cada interfaz se rigen de acuerdo a algún algoritmo. Estos algoritmos determinan la tasa a la cual el tráfico se direcciona a cada cola, determinando los recursos asignados a cada una de ellas para tratar los flujos correspondientes. En consecuencia para proveer Calidad de Servicio a una red es necesario suministrar o configurar en los dispositivos lo siguiente:

- Clasificación de la información, que sirve de base para que los dispositivos separen el tráfico en flujos.

- Colas y Algoritmos de Encolamiento que manejan el tráfico de los distintos flujos.

Estos dos puntos se conocen como Mecanismos de Manipulación de Tráfico. Estos mecanismos por si solos no son útiles, deben ser suministrados o configurados a través de muchos dispositivos en una manera coordinada para proveer un servicio extremo a extremo útil a través de la red. Para proveer servicios útiles se requiere, además de los mecanismos de manipulación de tráfico, Mecanismos de Provisión y

Configuración. Estos mecanismos coordinan los mecanismos de manipulación de tráfico sujetos a Políticas *(Policies)* planeadas por los administradores de la red.

1.4.1 MECANISMOS DE MANIPULACIÓN DE TRÁFICO

A continuación se presenta los mecanismos más significativos de manipulación de tráfico, estos incluyen:

- 802.1p
- *DiffServ*
- *IntServ*
- ATM, ISSLOW y otros

1.4.1.1 802.1p

Muchas redes de área local (LANs, *Local Area Networks*) están basadas en tecnología IEEE 802. Éstas incluyen Ethernet, Token Ring, FDDI (*Fiber Distributed Data Interface*), y otras variaciones de redes con medio compartido. 802.1p es un mecanismo de manipulación de tráfico para soporte de QoS en estas redes. QoS en las redes LAN es de interés debido a que comprenden un alto porcentaje de las redes en uso en *campus* universitarios, *campus* corporativos y oficinas.

802.1p define un campo en la cabecera de los paquetes IEEE 802, de capa 2, que puede manejar de uno a ocho valores de prioridad. Los *hosts* o *routers* envían tráfico en una LAN marcando cada paquete transmitido con el apropiado valor de prioridad. Los dispositivos LAN, tales como *switches*, *bridges*[1] y *hubs* deben estar preparados para tratar los paquetes adecuadamente (haciendo uso de los mecanismos de cola). El campo de prioridad marcado por 802.1p está limitado a la LAN. Una vez que los paquetes son llevados fuera de la LAN, a través de dispositivos de capa 3, la prioridad 802.1p es quitada.

---

[1] Bridge - Dispositivo, que opera en la capa enlace de datos, mediante el cual pueden conectarse varias redes LAN

1.4.1.2 DiffServ

Es un mecanismo de QoS de capa 3 que ha tenido un uso limitado por muchos años, aunque últimamente se ha intentado estandarizarlo. *DiffServ* define un campo en la cabecera de los paquetes IP, en capa 3, denominado DSCP (*DiffServ CodePoint*). Los *hosts* o *routers* envían tráfico dentro de una red *DiffServ* marcando cada paquete transmitido con el DSCP apropiado. Los *routers* dentro de la red *DiffServ* utilizan el DSCP para clasificar los paquetes y aplicarles encolamiento específico o tratamientos preestablecidos, conocidos como PHBs (*Per Hop Behaviors*).

Ejemplo de un PHB es el Envío Acelerado o EF (*Expedited Forwarding*). Este comportamiento está definido para asegurar que los paquetes sean transmitidos desde la entrada hasta la salida, a alguna tasa limitada, con muy baja latencia[1]. Otros tratamientos pueden especificar que unos paquetes tengan cierta prioridad relativa con respecto a otros paquetes, referente a *throughput* promedio o preferencia de descarte, pero no con particular énfasis sobre la latencia. PHBs se implementan usando mecanismos de encolamiento de capas inferiores.

Los PHBs son tratamientos individuales aplicados a cada *router*, por sí solos no garantizan QoS extremo a extremo. Sin embargo, concatenando *routers* con iguales PHBs (y limitando la velocidad a la cual los paquetes son sometidos por algún PHB), es posible usarlos para construir un servicio QoS extremo a extremo. Por ejemplo, una concatenación de PHBs EF, a través de una ruta preestablecida con un cuidadoso control de admisión puede producir un servicio similar al servicio ofrecido por una línea dedicada, el cual es apto para voz interactiva. Otras concatenaciones de PHBs pueden producir un servicio apto para reproducción de video, etc.

1.4.1.3 IntServ

Hay dos servicios definidos dentro de este campo. Estos son los *Servicios Garantizados* y el *Servicio de Carga Controlada*. Los Servicios Garantizados transportan un cierto volumen de tráfico a una latencia cuantificable determinada. El

[1] Latencia - Retardo en la transmisión de datos a través de una red

Servicio de Carga Controlada transporta un cierto volumen de tráfico dando la apariencia de una red ligeramente cargada. Estos son servicios cuantificables en el sentido de que ellos están definidos para proveer QoS cuantificable a una cantidad específica de tráfico.

Los Servicios Integrados están típicamente, pero no necesariamente, asociados al protocolo de señalización RSVP (*Resource ReSerVation Protocol*). La totalidad de los Servicios Integrados definen algoritmos de Control de Admisión, los cuales determinan cuánto tráfico puede ser admitido a una clase específica en un dispositivo de red particular, sin comprometer la calidad del servicio. Los Servicios Integrados no definen los algoritmos de encolamiento de capas inferiores a ser usados para proveer el servicio.

1.4.1.4 ATM, ISSLOW y otros

ATM es una tecnología de capa enlace que ofrece manipulación de tráfico de alta calidad. ATM fragmenta los paquetes en celdas, las cuales son encoladas y luego tratadas por algoritmos de encolamiento apropiados para obtener un servicio ATM particular. El tráfico ATM es cursado sobre circuitos virtuales (VC, *Virtual Circuits*), los cuales soportan uno de los varios servicios ATM. Estos incluyen: Servicio de velocidad constante de bit (CBR, *Constant Bit Rate*), Servicio de velocidad variable de bit (VBR, *Variable Bit Rate*), Servicio de velocidad indeterminada de bit (UBR, *Unknown Bit Rate*), y otros. ATM actualmente va más allá de un estricto mecanismo de manipulación de tráfico, en el sentido que incluye un protocolo de señalización de bajo nivel que puede ser usado para iniciar o finalizar un canal virtual.

Debido a que ATM fragmenta los paquetes en celdas relativamente pequeñas, puede ofrecer un servicio con muy baja latencia. Si es necesario transmitir un paquete urgentemente, la interfaz ATM puede estar siempre descongestionada para la transmisión, tomándose un tiempo de celda. Para comparar, considérese el envío normal de tráfico de datos TCP/IP sobre un enlace lento de módem sin el beneficio de la capa de enlace ATM. Un paquete típico de 1500 bytes, una vez iniciada su transmisión sobre un enlace de módem de 28.8 kbps, ocupará el enlace por alrededor de 400 milisegundos hasta que sea completamente transmitido [8],

sin permitir la transmisión de otros paquetes sobre el mismo enlace. Los Servicios Integrados sobre Capas de Enlace Lentas (ISSLOW, *Integrated Services Over Slow Link Layers*) se ocupan de este problema. ISSLOW es una técnica de fragmentación de paquetes IP en la capa enlace para su transmisión sobre enlaces lentos, tal que los fragmentos nunca ocupen el enlace más tiempo que algún umbral.

Otros mecanismos de manipulación de tráfico han sido definidos para varios medios de transmisión, incluyendo cable módems, plantas híbridas de fibra óptica y coaxial (HFC[1], *Hybrid Fiber Coaxial*), etc. Estos pueden usar mecanismos de señalización de bajo nivel, específicos a la capa de enlace, como por ejemplo la señalización UNI (*User Network Interface*) para ATM.

Se puede notar que existen mecanismos de manipulación de tráfico al nivel de capa red, así como también al nivel de capas inferiores. El mapeo en capas relacionadas con el medio puede tomar ventajas de las características específicas del medio. En un modelo basado en capa 3, las propiedades del medio pueden ser ignoradas cuando el tráfico pasa a través de la capa IP, este modelo clasifica al tráfico en uno de varios grupos discretos para simplificación. Las optimizaciones en la capa enlace son pasadas por alto en modelos que buscan "el mínimo común denominador". Ya que pocas clases discretas deben trabajar para todas las instancias, las eficiencias específicas, cuando los dos modos de transporte se juntan, simplemente no están disponibles.

Si la información IP es el único factor usado en el enlace, algunas bondades específicas del medio que pueden ser asociadas podrían ser pasadas por alto. Por ejemplo, Frame Relay y ATM ofrecen mecanismos explícitos de congestión y señalización, como no sucede con IP. Un *router* que enlaza ATM y Frame Relay no tendría forma de representar la información dentro del paquete IP, por tanto la señalización de congestión podría perderse.

---

[1] HFC - Es una tecnología de telecomunicaciones en la cual se utiliza fibra óptica y cable coaxial en diferentes porciones de una red, para cursar tráfico de gran ancho de banda.

1.4.1.5 Mecanismos de manipulación de tráfico por conversación vs. Agregados

Los Mecanismos de Manipulación de Tráfico Por Conversación son aquellos que pueden manejar cada conversación como un flujo independiente. En este contexto, una conversación incluye todo el tráfico entre una instancia específica de una aplicación específica de un *host* y una instancia específica de una aplicación correspondiente a la anterior en un *host* correspondiente al anterior. En el caso de tráfico IP, la dirección IP de origen/destino, el puerto, y el protocolo son los únicos parámetros que identifican a una conversación. Tradicionalmente, los mecanismos *IntServ* pertenecen a esta clasificación.

En los Mecanismos de Manipulación de Tráfico en Agregados, algún grupo de tráfico, proveniente de múltiples conversaciones, es clasificado en el mismo flujo y manipulado en conjunto. Para clasificar el tráfico se observa los identificadores de agregados ubicados en la cabecera de los paquetes. Ejemplos de esta clasificación son los Servicios Diferenciados y el protocolo 802.1p en capa 3 y capa 2 respectivamente. En ambos casos, los paquetes correspondientes a múltiples conversaciones son marcados con el mismo DSCP o con la misma marca 802.1p.

Cuando el tráfico es manipulado en una base "por conversación", los recursos son asignados en una base "por conversación". Desde una perspectiva de la aplicación, esto significa que el tráfico de la aplicación es dotado completamente de recursos independientemente de los efectos que causen otras conversaciones en la red. Aunque esto tiende a mejorar la Calidad de Servicio experimentada por la aplicación, también impone una carga en el equipamiento de la red. Los dispositivos de la red deben mantener un estado independiente para cada conversación y además aplicar un procedimiento independiente a cada una. En el núcleo de grandes redes, donde es posible soportar millones de conversaciones simultáneas, manipular el tráfico por conversación puede no ser práctico.

Cuando el tráfico es manipulado "en agregados", el mantenimiento del estado y el procesamiento en los dispositivos de los núcleos de grandes redes, representan una carga significativamente menor a la anterior clasificación. Por otro lado, la calidad de servicio percibida por una conversación no es tan independiente de los efectos del tráfico de otras conversaciones en el mismo flujo. Como resultado, en la

manipulación del tráfico por agregados, la calidad de servicio percibida por la aplicación tiende a comprometerse. Asignando un exceso de recursos a este tipo de tráfico se puede contrarrestar este efecto, sin embargo, tiende a reducir la eficiencia en el uso de los recursos de red.

1.4.2 MECANISMOS DE PROVISIÓN Y CONFIGURACIÓN

Para proveer QoS a una red en forma efectiva, es necesario efectuar la provisión y configuración consistente de los mecanismos de manipulación de tráfico descritos, a través de múltiples dispositivos de red.

Los mecanismos de provisión y configuración incluyen:

- Protocolo de Reservación de Recursos (RSVP, *Resource ReSerVation Protocol*) y Administrador del Ancho de Banda de la Subred (SBM, *Subnet Bandwidth Manager*)
- Protocolos y Mecanismos de Políticas (*Policies*)
- Protocolos y Herramientas de Administración.

1.4.2.1 Provisión vs. Configuración

Se usa el término Provisión para referirse a las tareas de administración más estáticas y grandes. Éstas pueden incluir la selección de equipo de red, el reemplazo de este equipo, la adición o supresión de interfaces, modificaciones de velocidad del enlace, cambios de topología, planeación de capacidad, y más. Se usa el término Configuración para referirse a tareas de administración más dinámicas y pequeñas. Estas incluyen, por ejemplo, las modificaciones a los parámetros de manipulación de tráfico en las redes *DiffServ*. La distinción entre estos dos términos no está claramente delineada, y a menudo son intercambiables a menos que se especifique de otra manera.

1.4.2.2 Mecanismos *Top-Down* vs. Señalizados

Es importante notar la distinción entre mecanismos QoS de configuración *Top-Down* y señalizados. Los primeros típicamente "ponen" la información de

configuración desde la consola de administración hacia los dispositivos de red. Los segundos típicamente llevan los requerimientos de QoS (y los requerimientos de configuración implícitos) desde un extremo de la red hacia el otro, a lo largo de la misma ruta cursada por los datos que requieren recursos del sistema. La configuración *top-down* típicamente se inicia en beneficio de una o más aplicaciones por un programa de administración de la red. La configuración señalizada típicamente se inicia por un cambio de la demanda de recursos de una aplicación.

1.4.2.3 Protocolo de reservación de recursos (RSVP)

El Protocolo de Reservación de Recursos (RSVP, *Resource ReSerVation Protocol*) es un mecanismo QoS de configuración señalizado. Es un protocolo por medio del cual las aplicaciones pueden hacer requerimientos extremo a extremo, por conversación, de QoS de una red y puede indicar también los requerimientos de QoS y capacidades de las aplicaciones pares. RSVP es un protocolo de capa 3, adecuado primariamente para su uso con tráfico IP. Ya que éste es un protocolo de capa 3, es bastante independiente de varios medios de red inferiores sobre los cuales opera. Por tanto, puede ser considerado como una capa abstracta entre las aplicaciones (o sistemas operativos de los *host*) y los mecanismos QoS específicos del medio.

Como actualmente está definido, RSVP usa semántica *IntServ* para conducir requerimientos de QoS, por conversación, a la red. Sin embargo, el protocolo RSVP por sí mismo no está limitado ni a su uso por conversación ni a la semántica *IntServ*. De hecho, extensiones de RSVP propuestas actualmente lo habilitan a ser usado para señalizar información estimando tráfico agregado. Otras extensiones lo habilitan a ser usado para señalizar requerimientos más allá de los tradicionales servicios *IntServ* (carga garantizada y controlada).

1.4.2.4 Administrador del ancho de banda de la subred (SBM)

El Administrador del Ancho de Banda de la Subred (SBM, *Subnet Bandwidth Manager*) está basado en una adecuación al protocolo RSVP, el cual extiende su utilidad a redes compartidas. En las subredes compartidas o LANs, las cuales

pueden incluir *hosts* y/o *routers* interconectados por un *switch* o *hub*, no puede actuar el protocolo RSVP estándar. El problema se da porque los mensajes RSVP pueden pasar desapercibidos a través de los dispositivos de capa 2 en una red compartida, admitiendo implícitamente flujos que requieren recursos de red compartidos. Los *hosts* y *routers* que reconocen RSVP admiten o rechazan flujos basados en la disponibilidad de sus recursos privados, mas no en la disponibilidad de recursos compartidos. Como resultado, los requerimientos RSVP destinados a *hosts* dentro de una subred compartida pueden comprometer los recursos dentro de dicha subred.

El Administrador del Ancho de Banda resuelve este problema habilitando dispositivos inteligentes que residen en la red compartida para ofrecer sus servicios como intermediarios de los recursos de dicha red. Estos dispositivos, en orden de aplicabilidad creciente, son:

- *Hosts* añadidos capacidad de SBM
- *Routers* añadidos capacidad de SBM
- *Switches* con capacidad de SBM, los cuales comprenden la red compartida.

Estos dispositivos corren automáticamente un protocolo que elige el o los dispositivos más adecuados, designando un determinado SBM (DSBM, *Designated* SBM). Cuando se elige un *switch*, estos subdividen la red entre ellos mismos basados en la topología de red de capa 2. Los *hosts* y *routers* descubren el dispositivo DSBM más cercano y enrutan los mensajes RSVP a través de éste. Así, el dispositivo DSBM observa todos los mensajes que afectarán los recursos en una subred compartida y provee control de admisión en beneficio de ésta.

1.4.3 MECANISMOS Y PROTOCOLOS DE POLÍTICAS

Los administradores de la red configuran los mecanismos de QoS sujetos a ciertas políticas. Las Políticas determinan cuáles aplicaciones y cuáles usuarios están facultados para variar la cantidad de recursos en diferentes partes de la red.

Los componentes de las Políticas incluyen:

- Una Base de Datos, la cual contiene los datos de las políticas mismas, como nombres de usuarios, aplicaciones, y los recursos de red a los cuales éstos están facultados.

- Puntos de Decisión de Políticas (PDPs, *Policy Decision Points*), estos traducen, a lo largo de toda la red, las políticas de las capas superiores a información de configuraciones específicas para los dispositivos de red individuales. Los PDPs también inspeccionan los requerimientos de recursos transportados en los mensajes RSVP y aceptan o rechazan ellos comparándolos con los datos de políticas.

- Puntos de Ejecución de Políticas (PEPs, *Policy Enforcement Points*), los cuales hacen cumplir las decisiones tomadas por los PDPs. Típicamente son dispositivos de red que conceden, o no, recursos al tráfico entrante.

- Protocolos entre las Bases de Datos, los PDPs y los PEPs.

1.4.3.1 Bases de datos de Políticas – Servicios de Directorio

Los mecanismos de Políticas confían en un conjunto de datos que describen como los recursos, en varias partes de la red, pueden ser asignados a tráfico asociado a determinados usuarios y/o aplicaciones. Los Esquemas de Políticas definen el formato de esta información. Se requieren dos tipos generales de esquemas; uno describe los recursos que deberán ser asignados en una forma de provisión de arriba hacia abajo (*top-down*), y el otro describe los recursos que pueden ser configurados por medio de una señalización extremo a extremo. Esta información tiende a ser relativamente estática y, al menos en una parte, necesita ser distribuida a través de la red. Consecuentemente, los Directorios tienden a ser las Bases de Datos adecuadas.

1.4.3.2 PDPs y PEPs

Los Puntos de Decisión de Políticas (PDPs) interpretan los datos almacenados en el esquema y controlan a los Puntos de Ejecución de Políticas (PEPs). Los Puntos de Ejecución de Políticas son los *switches* y *routers* a través de los cuales cursa el tráfico. Estos dispositivos tienen el control final sobre cuál tráfico recibe recursos y

cuál no. En el caso de la provisión de QoS *top-down*, el PDP impone las políticas a los PEPs en forma de información de clasificación (puertos y direcciones IP) y los recursos a los que están facultados los paquetes clasificados.

En el caso de la provisión de QoS por señalización, los mensajes RSVP transitan a través de la red por la misma ruta de los datos. Cuando un mensaje RSVP llega a un PEP, el dispositivo extrae el Elemento De Política del mensaje, así como también una descripción del tipo de servicio requerido y del perfil del tráfico. El elemento de política contiene generalmente la identificación de los usuarios y/o aplicaciones autenticadas. Luego el *router* pasa la información relevante del mensaje RSVP al PDP para comparar el requerimiento de recursos y los recursos admisibles para un usuario y/o aplicación. El PDP toma la decisión relacionando la admisibilidad y el requerimiento de recursos, retornando una aprobación o una negación hacia el PEP.

En ciertos casos, el PEP y el PDP pueden ser localizados en el mismo dispositivo de red. En otros, el PDP puede estar separado del PEP en forma de un Servidor de Políticas. Un Servidor de Políticas simple puede residir entre el Directorio y múltiples PEPs. Aunque muchas decisiones de políticas pueden ser realizadas trivialmente localizando en el mismo dispositivo al PDP y al PEP, hay ciertas ventajas si son realizadas usando un Servidor de Políticas.

1.4.3.3 Protocolos de Políticas

Cuando los mensajes RSVP transitan por dispositivos de red que entienden RSVP, éstos provocan la configuración de los mecanismos de manipulación de tráfico en los PEPs, incluyendo clasificadores y mecanismos de encolamiento, que proveen servicios integrados. Sin embargo, en muchos casos, el protocolo RSVP no puede ser usado para configurar estos mecanismos. En lugar de eso, se deben usar mecanismos *top-down*.

Éstos incluyen el Protocolo Sencillo de Administración de Red (SNMP. *Simple Network Management Protocol*), la Interfaz de Línea de Comandos (CLI, *Command Line Interface*), los Servicios Comunes de Políticas Abiertas (COPS, *Common Open Policy Services*) y otros. El Protocolo SNMP ha sido utilizado por muchos

años, inicialmente para monitoreo de la funcionalidad de los dispositivos de red desde una consola central, puede ser usado también para configurar la funcionalidad de un dispositivo. CLI es un protocolo usado inicialmente para monitorear y configurar equipo de red Cisco, debido a su popularidad, otras marcas proveen interfaces de configuración CLI a sus equipos. COPS es un protocolo que ha sido desarrollado en años recientes en el contexto de QoS, inicialmente fue planeado como un protocolo de políticas para RSVP, pero ahora es un protocolo general de configuración *DiffServ*. Todos estos protocolos son considerados *top-down*, porque tradicionalmente una consola de administración de alto nivel los usa para colocar información de configuración en un conjunto de dispositivos de red.

En el caso de QoS señalizado, la información de configuración detallada generalmente es transportada al PEP en forma de mensajes de señalización RSVP. Sin embargo, el PEP debe enterar al PDP para que éste efectivice, o no, el requerimiento de configuración. El protocolo COPS fue inicialmente desarrollado para pasar la información relevante contenida en el mensaje RSVP desde el PEP hacia el PDP y para pasar la decisión de política en respuesta. Obviamente si el PDP y el PEP están en el mismo dispositivo, no es necesario protocolo alguno.

También se requiere un protocolo para la comunicación entre el PDP y la base de datos. Ya que la base de datos tiende a tomar la forma de un directorio distribuido, comúnmente se usa el Protocolo Liviano de Acceso a Directorios (LDAP, *Lightweight Directory Access Protocol*) para este propósito.

1.5 BENEFICIOS DE LA IMPLEMENTACIÓN DE QoS

1.5.1 BENEFICIOS ECONÓMICOS

El costo de un circuito de datos depende básicamente del ancho de banda del enlace, pero también se relaciona con la ubicación geográfica entre los puntos a enlazarse, la tecnología que se utilice y el país en el cual se efectúa.

Las soluciones de voz sobre IP pueden disminuir drásticamente el gasto de llamadas de larga distancia. Direccionando las llamadas entre las oficinas mediante un enlace IP en lugar de un tradicional circuito de larga distancia, los sistemas PBX

con funcionalidad IP, pueden influenciar en la capacidad de la Intranet y reducir los costos.

En una red convergente, en la cual aplicaciones de voz sensitiva a la latencia, video con grandes requerimientos de ancho de banda y datos críticos en los negocios, deben coexistir, algún grado de control será necesario para mantener un adecuado funcionamiento de la red, aún en el supuesto de infinito ancho de banda. Optimizando los enlaces WAN puede representar un ahorro debido al precio del mismo.

1.5.2 DESARROLLO DE NUEVA TECNOLOGÍA

La necesidad de implementar calidad de servicio no proviene únicamente de la prioridad relativa de cierta clase de tráfico. Muchos de los protocolos de red, usados hoy en día, fueron diseñados para soportar robustas aplicaciones cliente-servidor, tales como transferencia de archivos.

Nuevos tipos de flujo, *peer-to-peer*, administrador-agente, y la inclusión de latencia, o aplicaciones sensibles al *jitter* en sistemas operativos modernos requieren regulación del ancho de banda y administración a través de la infraestructura tanto LAN como WAN para soportar redes multiservicios.

1.5.3 BENEFICIOS PARA LAS APLICACIONES

Las aplicaciones están consiguiendo mayor demanda. Aplicaciones de misión crítica desplegadas sobre redes IP cada vez requieren contar con mayor certeza en cuanto a calidad, confiabilidad y entrega en el tiempo adecuado. En particular, aplicaciones que usan voz, flujos de video, o multimedia deben ser administradas cuidadosamente dentro de la red IP con el objetivo de preservar su integridad.

La dificultad para manejar QoS comienza a incrementarse a causa de que muchas aplicaciones entregan impredecibles ráfagas de tráfico, por ejemplo, el modelo usado para el web, e-mail y aplicaciones de transferencia de archivos son virtualmente imposibles de predecir, aún los administradores de red necesitan ser hábiles para dar soporte a aplicaciones de misión crítica durante breves períodos. [9]

Las tecnologías QoS permiten a los administradores de red:

- Manejar aplicaciones sensitivas al *jitter*, tales como reproducciones de audio y video
- Manejar tráfico sensitivo al retardo, tal como voz en tiempo real
- Controlar el tiempo perdido en la congestión inevitable de flujos

1.5.4 BENEFICIOS PARA LAS EMPRESAS

El Internet ha llegado a ser parte esencial en los negocios, y las expectativas para asegurar calidad son las mismas tanto para una red privada como para una red pública. El Internet está siendo usado para impulsar tanto intranets como extranets, para habilitar el comercio electrónico entre los socios de negocios. Como los negocios están cada vez utilizando de mayor manera la Web, entonces llega a ser importante que los administradores aseguren que esas redes entreguen apropiados niveles de QoS. Tecnologías de Calidad de Servicio (QoS) proveen las herramientas para el despliegue de negocios de misión crítica sobre redes públicas.

1.5.5 BENEFICIOS PARA LOS PROVEEDORES DE SERVICIO

Claramente, las empresas y corporaciones están exigiendo cada vez mejores condiciones para su tráfico de negocios conducido sobre redes públicas. De igual forma están exigiendo cada vez más servicios de red a los proveedores, lo que les permite a ellos enfocarse más en el negocio interno y en reducir capitales excesivos. Esto significa que aquellos proveedores de servicio, que puedan ofrecer seguridad extremo a extremo al tráfico de negocios captaran mayor mercado.

Tecnologías QoS permiten al proveedor ofrecer más servicios, tales como soporte de tráfico en tiempo real o asignaciones específicas de ancho de banda.[9]

QoS, crea una nueva generación de proveedores de servicio, los cuales están en la capacidad de ofrecer más y mejores servicios a sus clientes.

Aunque las redes actuales todavía no están preparadas para soportar los servicios que se están experimentando, el mercado potencial es enorme. Finlandia, por

mencionar un ejemplo, está poniendo en servicio la primera red multiservicio gestionada sobre red IP (*Sonera* IP *Communicator*), que integra además de servicios de datos, servicios de audio y video. Estas redes van a transformar el modo en que los usuarios de negocios van a acceder y a pagar los servicios de telecomunicaciones. En Alemania se prevee un crecimiento exponencial del mercado de voz sobre IP en el entorno de negocios en los próximos años. [5]

Capítulo 2

PARÁMETROS TÉCNICOS QUE DEFINEN LA CALIDAD DE SERVICIO

CAPÍTULO 2

PARÁMETROS TÉCNICOS QUE DEFINEN LA CALIDAD DE SERVICIO

En términos cualitativos la calidad está directamente relacionada con la respuesta percibida por los usuarios finales cuando acceden a la red y por el grado de satisfacción de los mismos. Si la respuesta de la red no es buena es porque hay deficiencias de ingeniería y de diseño que impiden acomodar elevadas cargas de tráfico. En términos cuantitativos se refleja en una serie de parámetros que se pueden medir y ajustar convenientemente para proporcionar un grado de servicio satisfactorio.

En Internet los parámetros que determinan la calidad del servicio son:

- Retardo o Latencia
- Fluctuación del Retardo o *Jitter*
- Ancho de Banda
- Confiabilidad

2.1 RETARDO o LATENCIA

El Retardo o Latencia es la medida del rendimiento, con la cual los administradores de redes están más familiarizados, que corresponde al tiempo que tarda un paquete en llegar al destino desde su origen, en su tránsito por la red. Las principales fuentes de retardo son: la red misma (enlaces y dispositivos de red), el retardo introducido por el protocolo y el retardo introducido por los servidores y aplicaciones.

2.1.1 RETARDO EN LA INTERRED

El retardo de la Interred está compuesto por el retardo en los enlaces más el retardo producido en los dispositivos de red.

El retardo en los enlaces es inevitable y su valor depende del medio físico por el cual se están propagando las señales; la tecnología de conectividad que se utilice no afecta a este tipo de retardo. En un enlace satelital, la señal electromagnética se propaga por medio del aire a la velocidad de la luz, sin embargo, debido a la gran distancia entre el satélite y las estaciones terrenas, se introduce un retardo sustancial. Si se utilizan satélites geoestacionarios, distantes aproximadamente 36.000 Km de la Tierra, el retardo de propagación extremo a extremo introducido es de 250 a 300 milisegundos, siendo 270 milisegundos un valor común. Este retardo se incrementa a casi el doble, 540 milisegundos, para un sistema VSAT con un eje[1] [10].

Otros sistemas de comunicaciones introducen diferentes latencias, dependiendo del medio de transmisión utilizado. Los enlaces terrestres de microondas tienen un retardo de propagación de casi 3 microsegundos por kilómetro. Los enlaces de cable coaxial o fibra óptica tienen un retardo de aproximadamente 5 microsegundos por kilómetro [41]. Esto se debe básicamente a que las señales electromagnéticas viajan más rápidamente en el aire que en los materiales sólidos.

Se debe considerar que el retardo en la Interred es constante, sin importar el medio de transmisión utilizado, cuando la red está diseñada de tal forma que su capacidad excede la cantidad de información que la cursa en las horas críticas. En éste caso, todo el tráfico puede pasar a través de la red sin un apreciable retardo, el cual es impuesto únicamente por el equipo y el medio de transmisión, éste valor de retardo es relativamente pequeño e intrínseco a la red y no se puede hacer nada para disminuirlo.

El retardo en la Interred crece cuando el tráfico introducido en la red excede su capacidad debido a que el exceso de información se almacena en los *buffers* y se mantiene allí hasta que la red esté en capacidad de retransmitir la información. Una

[1] Eje: Estación terrena especial, con una antena grande de ganancia alta, para retransmitir el tráfico entre estaciones VSAT

red diseñada adecuadamente tiene mayor capacidad que la tasa de datos promedio esperada, de modo que le permita despejar las ráfagas de congestión rápidamente.

El retardo que introduce una red cuando la carga de tráfico excede su capacidad está directamente relacionada con la capacidad de los *buffers* de los dispositivos por los cuales un paquete debe cursar. Sin embargo, para tener una idea completa del retardo, además de conocer el tamaño del *buffer* se debe considerar la velocidad de los diferentes tipos de tráfico que cursan por estos *buffers*.

Cuando una red no está bien dimensionada puede producirse congestión, obteniéndose un retardo mayor al producido únicamente por el medio de transmisión y la capacidad de proceso de los equipos, éste retardo ocurre por varias razones:

- El tráfico es almacenado en los *buffers* debido a que los enlaces de entrada transportan más tráfico del que los enlaces de salida pueden manejar.
- El tráfico es descartado y debe ser retransmitido debido a que los *buffers* están llenos o no existen.
- El tráfico es enviado por una ruta alterna más lenta que la primaria debido a que sus conexiones están saturadas.

En una red congestionada como el Internet, los dispositivos están regularmente saturados, el encolamiento necesario en este caso introduce retardo.

Sobredimensionando la red se evita condiciones de congestión la mayoría de tiempo. Se hace necesario entonces buscar un equilibrio entre el costo del ancho de banda inutilizado y el retardo en el enlace, como se ilustra en la figura 2.1

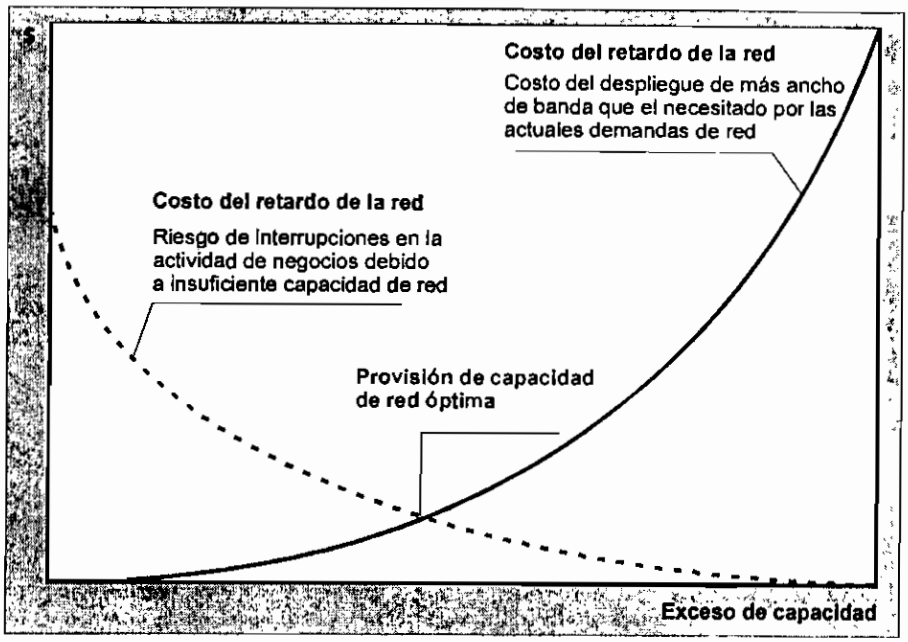

Figura 2.1 Capacidad vs. Costo de un enlace. [6]

En una LAN, los rápidos enlaces reducen drásticamente la duración de la congestión de la red, incluso, se tienen redes como Gigabit Ethernet cuya velocidad de transmisión (1 Gbps) es mayor que la capacidad de transmisión de un moderno computador (4 MBps).[6]

En tales redes (LAN) de alta velocidad, la latencia en los dispositivos es mínima. Pero en redes de área extendida, un retardo de 100 milisegundos en transmitir a través del Atlántico es razonable, pues está limitado simplemente por la velocidad de la luz. [6]

En las redes de área local, así como en las redes de área extendida compartidas, existe a menudo un impredecible retardo en acceder al medio. Este importante componente del retardo total depende del tipo de red y del protocolo que ésta use para acceder al medio. Se analiza a continuación el retardo en el acceso producido en dos redes muy comunes, tales como Ethernet y Token Ring. Luego, se estudiará la latencia introducida en los módems, presente en el acceso a Internet por medio de una línea telefónica.

2.1.1.1 Retardo en el Acceso en las Redes Ethernet

Las redes Ethernet emplean un mecanismo de acceso múltiple con escuche de portadora y detección de colisión (CSMA/CD, *Carrier Sense Multiple Access / Collision Detect*). Si la red está congestionada constantemente se puede obtener un retardo indefinido en el acceso, pues el nodo debe esperar hasta que la red esté disponible.

Podría pensarse que una estación se adueñe del canal produciendo un retardo infinito en el acceso a la red de las demás estaciones, sin embargo este caso es teórico y no debe ser visto si el diseño de la red es adecuado. Una forma de disminuir el acceso en el retardo es la conmutación. Cuando una red Ethernet es conmutada, el emisor nunca detecta otra transmisión y las estaciones finales pueden transmitir inmediatamente. Luego el *switch* Ethernet elegirá entre almacenar en *buffer* los datos o descartarlos si la capacidad no está disponible.

Desplegando *switches* Ethernet a través de la red es posible evitar las limitaciones de CSMA/CD. Sin embargo, en la ausencia de dichos *switches* se necesita mantener un nivel de congestión significativamente menor que la capacidad teórica máxima de una red a fin de asegurar un acceso al medio relativamente rápido, que permita soportar las nuevas aplicaciones y las redes convergentes.

Se debe tener en cuenta que aunque un nodo esté transmitiendo un archivo extremadamente largo no impedirá que otros dispositivos accedan a la red. CSMA/CD es un método de compartición del medio que verifica la disponibilidad del canal antes de transmitir. Un aspecto importante de esta especificación es el llamado Intervalo Entre Paquetes (IPG, *Inter Packet Gap*), el cual hace que los transmisores ocasionalmente efectúen una pausa en la transmisión. La red Ethernet de velocidad de 10 Mbps usa un periodo de silencio de 9.6 microsegundos después de transmitir una trama. Durante este periodo otras estaciones pueden apoderarse del control del enlace para enviar tramas. Este periodo de 9.6 microsegundos corresponde al tiempo que se demora en enviar 96 bits sobre cable coaxial de banda ancha a 10 MHz entre dos nodos separados una distancia definida.[6]

En una red Fast Ethernet de 100 Mbps el mecanismo CSMA/CD usa los mismos 96 bits, sin embargo, debido a que esta red es más veloz, el tiempo de IGP disminuye en un factor de 10, consecuentemente para Fast Ethernet el IGP es de 0.96 microsegundos. [6]

Cuando dos estaciones están conversando en una red *full duplex*, no hay colisiones. Si un tercer nodo intenta enviar tráfico, todavía habrá la probabilidad de que no existan colisiones, pues éste puede transmitir durante el IGP. Si existen más nodos, hay una probabilidad estadística por paquete de que dos estaciones intenten transmitir simultáneamente. Si se presenta la colisión, las dos estaciones esperan un tiempo aleatorio antes de retransmitir. Este tiempo es de carácter aleatorio para que los nodos no empiecen la retransmisión al mismo tiempo. De acuerdo a la especificación de Ethernet, mientras más veces colisionan los paquetes durante la transmisión, más largo será el tiempo aleatorio que deberán esperar las máquinas antes de la retransmisión. Después de 15 retransmisiones fallidas, el transmisor deja que se pierda el paquete, siendo los protocolos de capas superiores los encargados de su retransmisión, así como también se da un indicio de la calidad del enlace.

La detección de colisión está en función del retardo y de la distancia, ya que la distancia afecta al tiempo que les toma a dos transmisores colisionar en el peor de los casos. El retardo en el cable para la mayoría de los tráficos Ethernet es alrededor de 5 microsegundos por kilómetro, sea en *hubs*, receptores, transmisores, mientras en dispositivos similares bordea entre 0.1 a 1.9 microsegundos [6]. La tabla 2.1 muestra el retardo en varios medios de transmisión y dispositivos Ethernet.

Mientras más distantes se encuentren los nodos en el segmento Ethernet, más tiempo les tomará darse cuenta que existe una colisión. Para controlar el número de colisiones, el tiempo que se toma en transmitir la trama más pequeña permitida deberá ser mayor que el tiempo total que le toma en atravesar la red. La razón es simple: el emisor no debe permitir que la trama se transmita completamente si es que hay otra transmitiéndose simultáneamente.

Medio	Retardo
Repetidor Local	0,65 microsegundos
Repetidor de Fibra Óptica	1,55 microsegundos
Repetidor Multipuerto	1,55 microsegundos
Transceiver Multipuerto	0,10 microsegundos
Transceiver Estándar	0,86 microsegundos
Transceiver de Fibra Óptica	0,20 microsegundos
Transceiver de Par Trenzado	0,27 microsegundos
Hub	1,90 microsegundos
Cable Coaxial 10Base5	4.33 microsegundos por kilómetro
Cable Coaxial 10Base2	5.14 microsegundos por kilómetro
Par Trenzado blindado (STP)	5.7 microsegundos por kilómetro
Par Trenzado no blindado (UTP)	5.7 microsegundos por kilómetro
Fibra Óptica	5.14 microsegundos por kilómetro

Tabla 2.1 Retardo para varios medios y dispositivos Ethernet.[11][6]

Si se incrementa la velocidad de la red, se deben transmitir más bits por trama, o disminuir la distancia de la red. Una red 10BaseT tiene un dominio de colisión de 2000 metros. El tiempo que le toma a la electricidad atravesar dos veces esta distancia es el tiempo que se demora en transmitir la trama más pequeña a una tasa de 10 Mbps. Una red 100BaseT tiene un dominio de colisión de 200 metros; y una red Gigabit Ethernet (GE) tiene un dominio de 20 metros.[6]

Actualmente los puertos GE son conmutados y con capacidad de enlaces de datos *full duplex* convirtiendo a las colisiones en un fenómeno hipotético. Sin embargo el estándar GE define dos mecanismos para incrementar el dominio de colisión, en caso de que alguien decida construir *hubs* GE. El primer mecanismo es incrementar la longitud de la trama a 512 bytes, obteniéndose un tiempo de transmisión mayor y por lo tanto una distancia permitida mayor a 20 metros. El segundo mecanismo es permitir al nodo ráfagas de tramas de hasta 9018 bytes, lo cual compensa la penalización del tamaño de trama mínimo en la red. Esta tecnología permite incrementar el dominio de colisión a cerca de 200 metros manteniendo una buena tasa de transmisión.

2.1.1.2 Retardo en el Acceso en las Redes Token Ring

A diferencia de la redes Ethernet, las redes en anillo no son redes de difusión, sino un conjunto de enlaces individuales punto a punto, que forman un circulo. La ingeniería de anillos es casi completamente digital utilizando como medios de transmisión el par trenzado, el cable coaxial o la fibra óptica.

Como se mencionó, este sistema consiste de un conjunto de interfaces conectadas por enlaces punto a punto. Cada bit que llega a una interfaz se almacena en el *buffer* y posteriormente se retransmite al anillo. Mientras está en el *buffer*, el bit puede ser modificado. Este proceso introduce un retardo de un bit en cada interfaz.

Mientras las estaciones de una red Token Ring están inactivas, circula un patrón de bits especial llamado testigo o *token*. Cuando una estación quiere transmitir, debe tomar el testigo y retirarlo del anillo antes de empezar a enviar tramas. Debido a que existe sólo un testigo, sólo una estación podrá transmitir en un instante dado, resolviendo el problema del acceso al canal.

El retardo en este tipo de redes está implícitamente ligado al funcionamiento de las mismas, pues su diseño debe posibilitar en el anillo un retardo suficiente para contener un testigo completo que circule cuando todas las estaciones están inactivas.

El retardo tiene dos componentes: el retardo de propagación de la señal y el retardo de 1 bit introducido en cada estación. Una velocidad de propagación de señal típica es de 200.000 kilómetros por segundo, es decir un retardo 5 de microsegundos por kilómetro.[6]

En la mayoría de los anillos, los diseñadores deben suponer que las estaciones pueden apagarse en diferentes momentos, especialmente durante la noche. Si las interfaces se alimentan de energía desde el anillo, el apagado de la estación no tiene efecto alguno, pero si se energizan externamente deben estar diseñadas para conectar la entrada a la salida al interrumpirse la energía, con lo que se elimina el retardo de un bit. En este caso podría ser necesario la introducción de un retardo artificial en el anillo para asegurar que sea capaz de contener un testigo.

El protocolo de subcapa MAC (*Medium Access Control*) de Token Ring posibilita al primer bit de una trama recorrer el anillo completo y regresar a la interfaz de transmisión antes de que la trama se haya transmitido. En consecuencia, la estación transmisora deberá drenar el anillo mientras continua transmitiendo.

Una estación puede apoderarse del testigo durante un tiempo de retención predeterminado de 10 milisegundos, a menos que se establezca un valor distinto. Si la transmisión de una trama adicional excediese el tiempo de retención del testigo, la estación regenera la trama del testigo de 3 bytes y la pone en el anillo.

El protocolo IEEE 802.5 tiene un elaborado esquema para manejar tramas con diferentes prioridades. La trama del testigo de 3 bytes contiene un campo en el byte intermedio que indica la prioridad del testigo. Cuando una estación quiere transmitir una trama de prioridad *n*, debe esperar hasta que puede capturar un testigo cuya prioridad sea igual o menor a *n*. En Token Ring, una estación que solo tiene tramas de prioridad baja puede sufrir un retardo significativo esperando la aparición de un testigo de prioridad baja. Ciertamente se dará un buen servicio al tráfico de prioridad alta.

2.1.1.3 Retardo en el Acceso Remoto – módems

Mientras los avances en los sistemas operativos y en la capacidad de procesamiento de los PCs han reducido rápidamente el retardo en el sistema a niveles aceptables, los módems siguen siendo fuentes de niveles inaceptables de retardo.

Es importante estudiar el retardo en los módems debido a dos causas. La primera, el alto porcentaje del retardo total que introduce el módem; y, la segunda, el uso masivo de éstos, pues la mayoría de usuarios que acceden al Internet lo hacen a través de módems por medio de una línea telefónica.

Una encuesta que mide la velocidad de conexión para acceso a Internet dentro de los Estados Unidos realizada en 1997 se muestra en la tabla 2.2

VELOCIDADES DE CONEXIÓN A INTERNET	
Velocidad de Conexión	**Porcentaje de Usuarios**
Menor a 14 kbps	0,3 %
14 kbps	5,1 %
28 kbps	21,5 %
33 kbps	27,5 %
56 kbps	5.9 %
128 kbps	3,2 %
1 Mbps o mayor	13,4 %
Inseguro	23,1 %

Tabla 2.2 Velocidades de conexión a Internet. [12]

La encuesta indica que al menos el 61% de los estadounidenses acceden a Internet por medio de módems *dial-up*. La proyección mostrada en la tabla 2.3 indica que los módems *dial-up* han dominado el acceso doméstico a Internet en los últimos tres años.

Tipo de Acceso	1998	1999	2000
Módems *Dial-up*	96,5%	92,1%	87,9%
Cable Módems	2,4%	5,7%	9,5%
ISDN, ADSL, Inalámbrico	1,1%	2,3%	2,6%

Tabla 2.3 Tipos de acceso doméstico[1] a Internet. [13]

Actualmente, la mayoría de usuarios utiliza módems V.34 cuya velocidad de transmisión es 28.8 kbps. A continuación se explicará el comportamiento de este tipo de módems, incluyendo el puerto serial (V.24), y las funciones de compresión (V.42bis) y de detección y corrección de errores (V.42), y su influencia en el retardo.

Los datos a ser transmitidos por el procesador son enviados a un transmisor / receptor asíncrono universal [2] (UART, *Universal Asynchronous Receiver / Transmitter*), por medio del bus ISA[3], cuya capacidad es de 8 Mbps. El UART se

[1] Usuarios domésticos.- Usuarios que acceden a Internet desde su domicilio.

[2] UART.- Chip que controla el envío y recepción de datos desde y hacia un puerto serial.

[3] ISA.- Arquitectura de bus estándar que permite la trasferencia de 16 bits ente la tarjeta madre y una tarjeta de expansión.

comunica con el procesador por medio de una interrupción (IRQ, *Interrupt ReQuest*)[1] cuando su *buffer* de recepción está listo para transmitir (desde el módem) o cuando el *buffer* de transmisión está lleno (hacia el módem). Los valores que se configuran por defecto en el sistema operativo Windows 95/98 son: 8 bytes para el *buffer* de recepción y 16 bytes para el *buffer* de transmisión.

El UART se comunica con el módem mediante la recomendación del UIT-T V.24 (RS-232). Esta interfaz permite un control del flujo de datos a través de hardware (RTS/CTS)[2] o software (Xon/Xoff). La máxima velocidad que se alcanza para un UART 16550 es 115 kbps [13], siendo ésta configurable por el usuario final. La velocidad en la interfaz serial generalmente se establece mayor que la velocidad del módem, con el objetivo de optimizar el enlace tomando en cuenta que el módem efectúa compresión de datos. Por ejemplo, si la relación de compresión es 2:1 se debe establecer en el puerto serial una velocidad de al menos 57.6 kbps para optimizar la conexión de 28.8 kbps que alcanza el módem.

La norma V.42bis de compresión de datos crea un diccionario para cadenas de caracteres, asignando a cada cadena una palabra código. El tamaño del diccionario está entre 512 y 2048 cadenas de caracteres, y la longitud de éstas entre 6 y 250 caracteres, siendo 32 un valor típico [13]. El retardo añadido por el algoritmo de compresión depende del radio de compresión. Por ejemplo, si se usa palabras código de 11 bits y la relación de compresión es 3:1, el *buffer* almacenará en promedio 33 bits antes de que se compriman los datos.

Normalmente las tareas de compresión se pueden realizar durante el flujo de datos en tiempo real, y no tendrán impacto adicional en el retardo. Sin embargo, si hay tráfico que no puede ser comprimido, también existirá un retardo cuando se intente comprimir este.

La norma de detección y corrección de errores V.42 ensambla los datos de usuario en tramas utilizando el Procedimiento de Acceso al Enlace para Módems (LAPM,

---

[1] IRQ.- Líneas de hardware utilizadas por los periféricos de un computador para comunicar al procesador que el dispositivo está listo a enviar o recibir datos.

[2] RTS/CTS.- Señalización entre un DTE y un DCE para comunicar un requerimiento de envío (RTS) e indicar que el dispositivo (DCE) está listo (CTS).

Link Access Procedure for Modems). Este proceso le permite al módem controlar la secuencia de los datos, detectar errores, retransmitir tramas y efectuar control de flujo. El tamaño típico de esta trama es de 128 bytes, sin embargo este valor puede ser negociado entre los módems. El retardo introducido en este procedimiento depende del tamaño de la trama, de la calidad de la conexión y de la dimensión del encabezado.

Para la transmisión de una trama típica, el módem deberá almacenar 128 bytes de datos en el *buffer*, antes de realizar la compresión, o en su defecto activar temporizadores, por ejemplo de 10 milisegundos. En este paquete se añadirá 6 bytes en el encabezado y la inserción de cero (1 bit por cada 160 bits), obteniéndose una trama de cerca de 135 bytes, es decir 5.5% de *overhead*[1].

Una calidad de conexión baja implica mayor número de retransmisiones, los módems típicamente se conectan a velocidades que logran un rendimiento de no más de 1 error en 1000 bloques de datos (0.1% de sobrecarga), sin embargo, existen módems que permiten 1 error en 100 bloques de datos (1% de sobrecarga).

La norma V.34 fue diseñada para maximizar la adaptabilidad a la conexión telefónica por medio de técnicas de modulación complejas como QAM (*Quadrature Amplitude Modulation*) y varias opciones extras disponibles en el momento que se establece la conexión.

Los módems V.34 utilizan modulación TCM (*Trellis Code Modulation*), un conjunto de algoritmos que permiten mejorar la eficiencia respecto del esquema de modulación convencional realizando corrección de errores en el receptor (sin retransmisión), necesitándose el uso de *buffers* de recepción cuyo tamaño típico es 32 símbolos. Esta técnica provee también inmunidad adicional al ruido.

Otra funcionalidad del módem que introduce retardo es conocida como *shell mapping*. Esta técnica considera las posiciones de un número de símbolos (amplitud y fase) sucesivos, ordenándolos de tal forma que minimice la potencia media de la señal. Provee además una ganancia de la relación señal a ruido.

[1] Overhead.- Datos de control anexados a la información.

Influye en el retardo, pues previo a su operación, debe almacenar un número de símbolos consecutivos en el *buffer*. En un típico caso, a una velocidad de conexión del módem de 28.8 kbps con modulación QAM de 9 bits por símbolo y con un tamaño de trama de 8 símbolos consecutivos, se introduce un retardo de 2.5 ms [13]. El tamaño de *buffer* V.34 depende del tamaño de la trama utilizada en la función del *shell mapping*.

Trabajar sobre un bloque de símbolos, en lugar de byte a byte, reduce la carga del procesador DSP (*Digital Signal Processor*). Los módems típicamente tienen procesamiento en bloques en cada interfaz externa de datos. La función de ecualización también actúa sobre bloques de datos y por tanto debe almacenar en *buffers* la información recibida, introduciendo éstos un retardo de 10 ms en el lado del receptor.

A continuación se presenta la localización teórica del retardo en una conexión por módem para una típica sesión de Voz sobre IP (VoIP). Se asume que el módem opera a 28.8 kbps (V.34) y que el flujo de datos de voz es 12.5 kbps, es decir un paquete IP de 94 bytes cada 60 ms.

La figura 2.2 ilustra los componentes del retardo en el módem cuando las funciones V.42 y V.42bis han sido habilitadas.

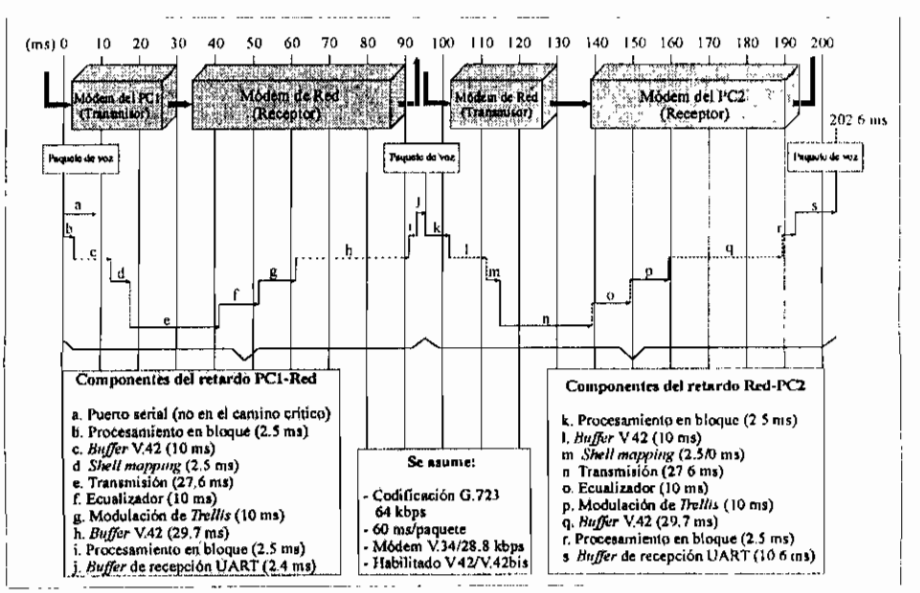

Figura 2.2 Retardo teórico en el módem. [13]

Cada flecha representa el retardo en milisegundos y múltiples flechas en el mismo nivel representan varios pasos a través de la misma función. Las flechas punteadas representan la norma V.42

La figura predice un retardo en el módem, en un viaje de ida y vuelta, de 202.6 ms para una típica sesión VoIP. Se debe considerar lo siguiente:

- Puerto serial: El retardo no es crítico debido al *buffer* V.42.
- Procesamiento en bloque: se asume un tamaño de ocho símbolos a una velocidad de la señal de 3200 baudios (retardo de 2.5 ms), aplicado en cada interfaz V.24.
- V.42: Se asume un bloque V.42 de 128 bytes (tamaño del *buffer*). Como el paquete de 94 bytes no llena el *buffer*, se debe cumplir un tiempo de 10 ms de inactividad en el *timer*. En el lado del receptor, se almacena en el *buffer* 101 bytes, la detección y corrección de errores se efectúan en tiempo real. Se tiene un 5.5% de sobrecarga y retransmisión del 0.1 % de los bloques.
- V.34: La función de *Shell Mapping* utiliza una trama de 8 símbolos, a 3200 baudios, esto es 2.5 ms. La modulación TCM ocurre en tiempo real, sin embargo almacena en *buffer* 32 símbolos, aportando con 10 ms de retardo.

2.1.2 RETARDO EN LOS PROTOCOLOS

Además del retardo introducido en la subcapa de acceso al medio MAC, también existe un retardo producido por los protocolos de capas superiores.

La pérdida o el descarte de paquetes durante la comunicación, puede ser motivo de retransmisión en ciertos protocolos, aumentando aún más el retardo de la red. Los paquetes pueden ser descartados debido a una falla en la suma de verificación (*checksum*) o debido a la incapacidad de los *buffers* en almacenar datos cuando existe congestión.

La retransmisión puede ser efectuada en un ambiente nodo a nodo como en las redes X.25, o en un ambiente emisor – receptor como en una red Frame Relay.

Puede ser comunicada explícitamente (X.25 y Frame Relay) o puede ser deducida a través de información secuencial como ocurre en el protocolo TCP/IP.

2.1.2.1 Retransmisión en redes X.25

Las primeras redes estuvieron expuestas a altas tasas de error, siendo su diseño adaptado a estas circunstancias.

El protocolo de capa red de X.25, básicamente, permite al usuario establecer circuitos virtuales y después enviar paquetes de hasta 128 bytes a través de ellos [10]. Estos paquetes se entregan en forma confiable y en orden. La mayor parte de redes X.25 trabajan a velocidades de hasta 64 kbps, haciéndola obsoleta para muchos propósitos, no obstante su difusión es grande.

Las redes X.25 utilizan un mecanismo de transmisión y asentimiento (*transmit-and-acknowledge*) entre los *switches* individuales de la red. La red por sí misma implementa un control de flujo entre cada *switch*, como se muestra en la figura 2.3.

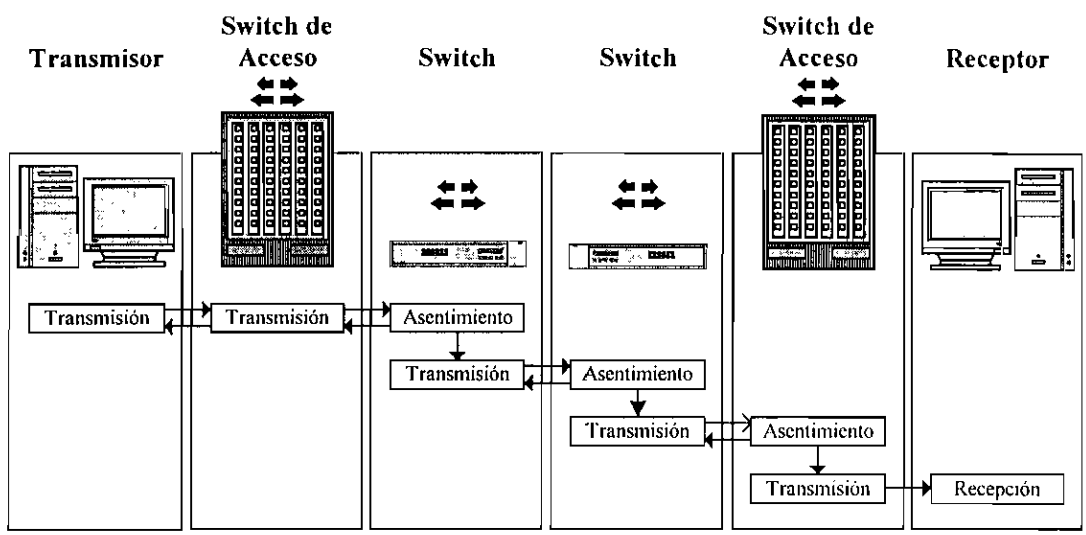

Figura 2.3 Acuses de Recibo en una red X.25. [6]

Este tipo de asentimiento paso a paso conduce a un retardo en cada movimiento de la información, esperando una respuesta de confirmación y verificando que la información ha sido transmitida exitosamente.

Este protocolo es bastante ineficiente en redes que tienen una baja tasa de error, sin embargo es adecuado para redes no confiables, ya que la pérdida es descubierta inmediatamente entre dos *switches*. Por lo tanto el impacto de un error y la posterior retransmisión es menos significante, pero la cantidad de correcciones de errores dentro del sistema es aniquilante para redes modernas.

2.1.2.2 Retransmisión en redes Frame Relay

Frame Relay es esencialmente una red X.25; pero con un rendimiento mayor debido a una tasa de transmisión más rápida y a la reducción en el encabezado del protocolo como una consecuencia de enlaces más confiables. Frame Relay usualmente trabaja a una velocidad de 1.5 Mbps, pero ofrece menos funciones que X.25. [10]

A diferencia de X.25, Frame Relay no proporciona un control de flujo normal, sin embargo, tiene un bit en el encabezado que un extremo de la conexión puede encender para indicar que hay problemas; el uso de este bit es opción de los usuarios.

La figura 2.4 muestra el control de flujo de la información en una red Frame Relay.

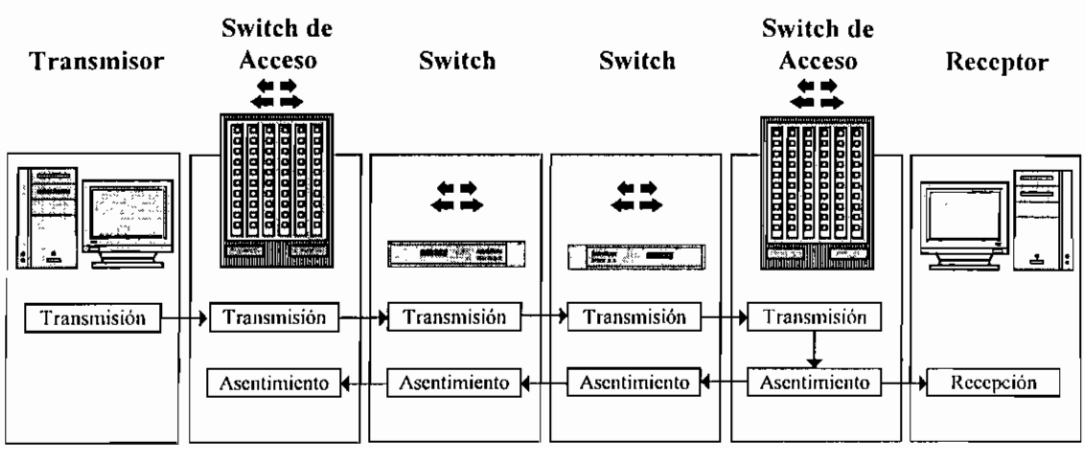

Figura 2.4 Acuses de Recibo en redes Frame Relay. [6]

Como muestra la figura, un circuito Frame Relay usa acuses de recibo sólo en los puntos finales del enlace. Una vez que la trama ha sido exitosamente conmutada a través de la red, el receptor envía un acuse de recibo al emisor.

2.1.2.3 Retransmisión en el Protocolo TCP (*Transport Control Protocol*)

El protocolo TCP es empleado para transferencia de archivos u otras sesiones largas que usan el máximo tamaño posible de paquete, también es usado en aplicaciones pequeñas e interactivas como telnet y http. Las sesiones largas constituyen la mayoría del volumen del tráfico IP público, en cambio, las sesiones cortas constituyen la mayoría de los flujos del tráfico IP.

TCP fue concebido como un protocolo orientado a conexión para interconectar redes. Debido a que trabaja sobre una variedad de medios, con confiabilidad y tasa de error desconocidas, debe ofrecer corrección de errores y enviar servicios garantizados.

El Protocolo de Control de Transmisión garantiza que el flujo de datos transmitidos será reensamblado intacto en el receptor. Mantiene un ordenamiento FIFO (*First Input First Output*), el primero que entra será el primero que sale, a través de una secuencia de números que usan el emisor y el receptor para dividir y reensamblar los paquetes respectivamente, además asegura el envío mediante un sistema de acuse de recibo. TCP también ofrece otras funciones como retransmisión de la información perdida y ajuste de la velocidad a la cual introduce tráfico en la red.

El protocolo no hace reconocimiento de paquetes por los números de los paquetes, en lugar de ello, reconoce que los datos han sido recibidos en la secuencia correcta por la posición del byte en el flujo. Por ejemplo, cuando el receptor ha recibido correctamente todos los datos hasta el byte 2048, enviará un paquete de regreso con el bit de asentimiento ACK[1] encendido, y el número de asentimiento puesto en 2049.

Cuando un receptor de TCP nota que se ha perdido un segmento, no le informa al emisor que una parte de la transmisión se ha perdido, en lugar de esto, almacena en el *buffer* los datos que llegan y le informa al emisor que ha recibido hasta la posición del byte del segmento perdido, por ejemplo n+1. El emisor eventualmente se dará cuenta de lo ocurrido y reenviará el segmento en la posición n+2, como se muestra en la figura. 2.5

[1] ACK - Acrónimo utilizado para reconocer al bit de asentimiento

Debido a que el medio no orientado a conexión sobre el cual corre el protocolo TCP es no confiable e impredecible, el truco real de TCP es saber cuanto tiempo esperar antes de declarar un paquete perdido y cuando reenviarlo. Si el emisor espera un periodo de tiempo demasiado pequeño, puede reenviar datos que han llegado sólo un poco tarde, dando como resultado una retransmisión innecesaria y desperdiciando la capacidad de la red. En cambio, si el emisor espera tiempos demasiado largos en reenviar un paquete, la conexión TCP se tornará lenta porque los receptores almacenan en sus *buffers* una cantidad considerable de tráfico mientras esperan un segmento perdido, retardando de esta forma el envío del flujo de datos.

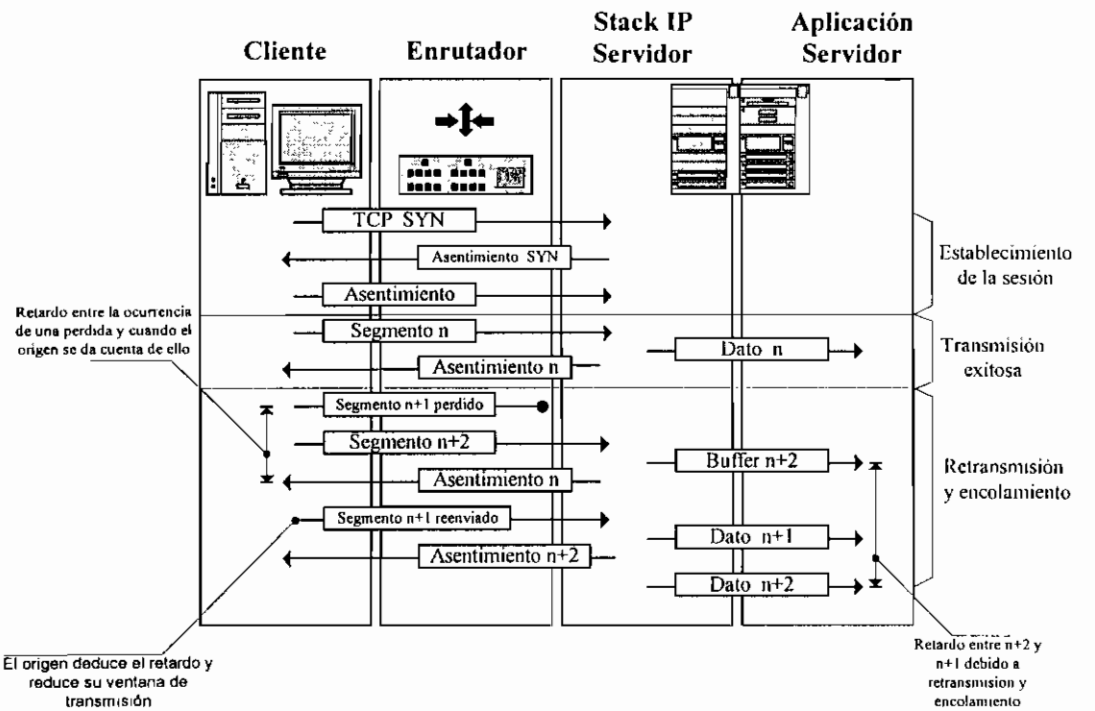

Figura 2.5 Flujo de datos y acuses de recibo en una sesión TCP. [6]

La especificación original del protocolo TCP sugiere que el tiempo de espera antes de la retransmisión sea igual a dos veces el tiempo de ida y vuelta RTT (*Round Trip Time*) [14]. Esta consideración no toma en cuenta el nivel de *Jitter* presente en la red. En una red perfectamente predecible el acuse de recibo debería demorarse exactamente un tiempo igual al RTT contado a partir del envío del paquete, en una red afectada por el *Jitter* este tiempo será mayor. Los modernos *stacks* de protocolos varían este retardo entre una y dos veces el tiempo de un RTT,

dependiendo de las características medidas de la línea. Por ejemplo, si la línea tiene una amplia variación de retardo se esperará un tiempo cercano a dos veces el RTT, pero si la línea tiene una pequeña variación de retardo, el tiempo de espera será cercano al RTT.

Si los datos llegan fuera de secuencia al receptor, éstos serán almacenados en el *buffer* hasta que esté disponible la secuencia completa para efectuar el reensamblaje. Este almacenamiento en *buffers* puede conducir a retardos considerables en casos donde una porción descartada del flujo habría sido aceptable para la aplicación, si ésta necesitara un envío rápido de los últimos datos.

Cuanto mayor sea el retardo, mayor ha de ser el esfuerzo del protocolo de transporte TCP para funcionar eficazmente, ya que la red tiene que "aguantar" más datos en tránsito, lo que afecta a los contadores y temporizadores asociados con el protocolo. TCP es un protocolo que ajusta dinámicamente la velocidad de envío al flujo de información realimentada desde el receptor mediante las notificaciones ACK (las que confirman el éxito de la llegada correcta de paquetes a su destino final). A medida que crece el retardo entre emisor y receptor, más insensible resulta este mecanismo de control de flujo, con lo que el protocolo se hace también menos sensible a los cambios dinámicos en la carga de la red. Para aplicaciones típicas en tiempo real (basadas en UDP, *User Datagram Protocol*, sin mecanismos TCP de control extremo a extremo), tales como voz y vídeo, el aumento del retardo hace que la respuesta de la red sea tan pobre que resulta inservible.

Las redes no confiables influyen en gran manera en el rendimiento de la conexión TCP. El protocolo decide la cantidad de datos que introduce en la red y consecuentemente el rendimiento del enlace, basado en los tiempos de respuesta y los niveles de pérdida de los paquetes que envía. La pérdida de paquetes y la retransmisión en un entorno TCP afecta no sólo al retardo del paquete perdido, sino también a la percepción que se forma el transmisor sobre la capacidad de la red.

El funcionamiento de este protocolo puede ser apreciado en la figura 2.6.

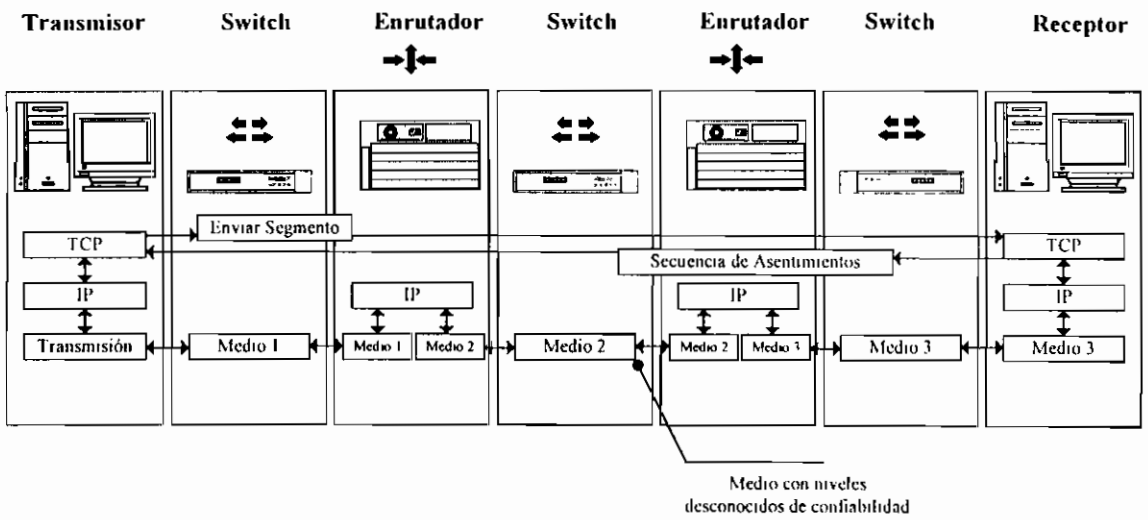

Figura 2.6 Funcionamiento del protocolo TCP. [6]

2.1.3 RETARDO EN LOS SERVIDORES

Corresponde al retardo que ocurre entre la recepción de un requerimiento de la red y la respuesta efectiva, debido a procedimientos internos del servidor.

El retardo introducido en los servidores es importante y no se debe pasar por alto. En muchos casos, éste es mayor que el introducido en la red. Un estudio realizado en uno de los más grandes ISPs (*Internet Service Providers*) de Estados Unidos demuestra que sus servidores son un componente significativo en el retardo total extremo a extremo [15].

Un ejemplo representativo de las medidas efectuadas a la red del ISP mencionado, la cual está muy bien estructurada, se presenta en las figuras 2.7 y 2.8.

En la figura 2.7 se muestra el tiempo de respuesta de un servidor de noticias a requerimientos de artículos de 40 kbytes de tamaño, durante 24 horas. Se observa que éste oscila entre 1 y 2 segundos durante la mayor parte del tiempo, llegando a producirse picos de 7 segundos.

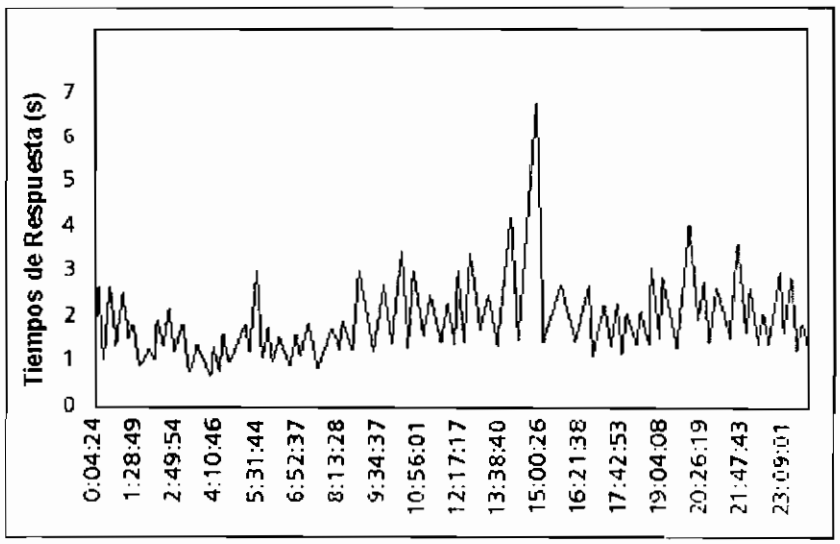

Figura 2.7 Tiempo de Respuesta de un Servidor perteneciente a un ISP. [15]

La figura 2.8 ilustra la respuesta de la red, costa a costa de los EEUU, en la transferencia de 40 kbytes de datos para ochos de los más ocupados puntos de presencia del ISP. El rango de valores medios está entre los 300 y 700 milisegundos. Como se mencionó, el retardo en el servidor es un componente importante en el retardo total extremo a extremo.

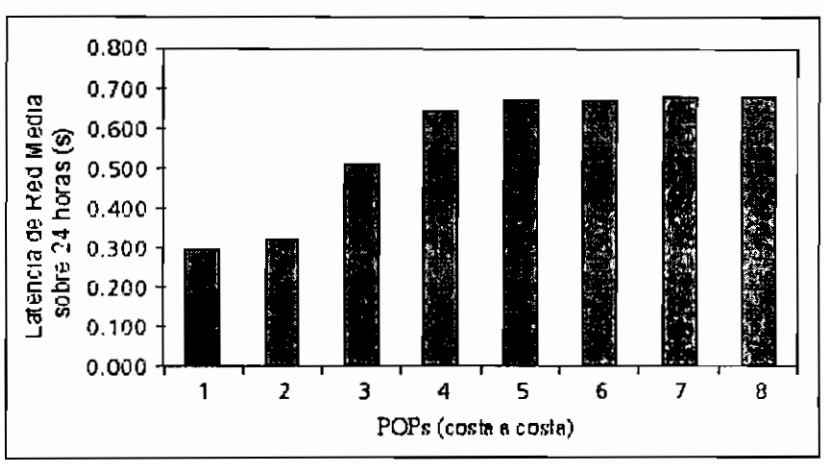

Figura 2.8 Tiempo de Respuesta de la Red. [14]

En general, para cualquier servidor, el retardo está en función de la carga del servidor, la cual proviene de varias fuentes:

- El CPU del servidor puede estar ocupado por otro proceso y es incapaz de responder pronto a los requerimientos.

- El requerimiento puede demandar procesamiento computacional intensivo, tal como ejecutar *scripts*[1] en tiempo real.

- El servidor puede ser capaz de procesar requerimientos rápidamente, pero el sistema que efectúa las acciones de fondo, como la búsqueda en una base de datos, puede tomar mucho tiempo.

Existen sistemas de balanceo de carga que disminuyen el problema del retardo en el lado del servidor. Pueden ser rudimentarios o muy sofisticados, como aquellos que permiten compartir la carga total entre varios servidores. Estos sistemas son esenciales en la provisión de Calidad de Servicio extremo a extremo.

2.2 FLUCTUACIÓN DEL RETARDO O *JITTER*

Es la variación del tiempo entre paquetes consecutivos en el receptor, debido básicamente a la diferencia de velocidad con que los paquetes atraviesan la red. Este concepto se ilustra en la figura 2.9, donde se aprecia que las tramas tercera y quinta arriban tarde, a un tiempo de recepción R2 y R4 respectivamente.

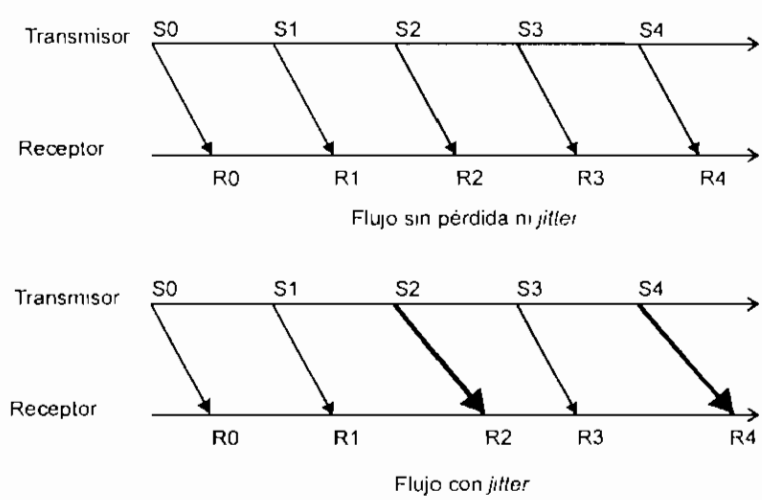

Figura 2.9 *Jitter* [16]

[1] Script - Programa o secuencia de instrucciones interpretadas por algún otro programa en lugar del procesador del computador

El aumento de la fluctuación provoca que el protocolo TCP haga estimaciones muy conservadoras sobre el tiempo de ida y vuelta (RTT), con la consiguiente falta de eficacia para restablecer el correspondiente flujo de datos cuando se superan los umbrales prefijados (*timeouts*). Para aplicaciones en tiempo real, el aumento de esta fluctuación entrega en destino una señal distorsionada, que se podría corregir aumentando el tamaño de las colas en el receptor a fin de reproducir fielmente la señal. Lógicamente, esto originaría un aumento del retardo, lo que no importaría demasiado en señales continuas (como audio o vídeo continuo), pero que dificultaría enormemente el mantenimiento de sesiones interactivas, tales como telefonía sobre IP o teleconferencias. La necesidad de minimizar el Retardo y desterrar el *Jitter* ha incentivado estudios que permiten adaptar dinámicamente el tamaño del *buffer* que se necesita para eliminar el *Jitter*, minimizando su impacto en la latencia.

2.2.1 FLUCTUACIÓN FÍSICA

La mayoría de redes proporcionan un nivel de retardo constante. En los sistemas antiguos tal vez se producía una variación en la velocidad de transmisión sobre distancias muy grandes. En cambio, los sistemas modernos incluyen mejor corrección de error y blindaje, previniendo variación física. No obstante, la dispersión introduce *Jitter* en los enlaces ópticos de alta velocidad.

2.2.2 FLUCTUACIÓN EN EL ACCESO

Es el cambio en el retardo que sufre una aplicación hasta obtener el derecho a transmitir. En una red con protocolo de acceso CSMA/CD la variabilidad del retardo es alta, en cambio, para redes que no comparten el medio, éste será menor. Reducir la variación en el acceso al medio es una razón para utilizar redes LAN conmutadas.

2.2.3 FLUCTUACIÓN EN LA RED

Cuando una red está congestionada, la profundidad del encolamiento y la retransmisión debido al descarte de paquetes, son las mayores fuentes de retardo. El tamaño de la cola cambia con el nivel de congestión, la misma que se

incrementa con la retransmisión de paquetes. Para disminuir este tipo de fluctuación se establecen controles de admisión, ajuste de colas, y además, se da prioridad al tráfico urgente evitando su encolamiento.

2.2.4 FLUCTUACIÓN EN EL ESTABLECIMIENTO DE LA SESIÓN

Cuando una aplicación establece una sesión a través de la red se produce una secuencia de acuses de recibo. Dependiendo de factores como carga del servidor o tasa de descarte, la secuencia puede experimentar un retardo impredecible antes de empezar el servicio.

2.3 ANCHO DE BANDA

El ancho de banda de una señal de comunicación es una medida del rango de frecuencias que la señal ocupa [17]. Todas las señales transmitidas, sean análogas o digitales, tienen un cierto ancho de banda.

El ancho de banda es directamente proporcional a la cantidad de datos transmitidos o recibidos por unidad de tiempo. En sentido cualitativo, el ancho de banda es proporcional a la complejidad de los datos para un nivel dado de desempeño del sistema. Por ejemplo, se requiere mayor ancho de banda para bajar de Internet una fotografía en un segundo, que el requerido para bajar un archivo de texto en un segundo. Archivos de sonido extensos, programas de computadoras, y videos requieren aún más ancho de banda para tener un aceptable desempeño en los sistemas. La realidad virtual[1] (VR, *Virtual Reality*) y presentaciones audiovisuales tienen los mayores requerimientos de ancho de banda.

El medio de transmisión limita mucho las componentes de frecuencia de la señal, ya que permite sólo la transmisión de cierto ancho de banda. En el caso de ondas cuadradas, éstas se pueden simular con ondas senoidales en las que la señal sólo contenga múltiplos impares de la frecuencia fundamental. Cuanto más ancho de banda, más se asemeja la función seno multifrecuencia a la onda cuadrada. Pero generalmente es suficiente con las tres primeras componentes.

---

[1] Realidad Virtual.- Entorno interactivo simulado por computadores, en el cual pueden interactuar los usuarios empleando periféricos tales como guantes electrónicos y pantallas gráficas colocadas en la cabeza

Si se duplica el ancho de banda, puede duplicarse la velocidad de transmisión a la que puede ir la señal. Pero al aumentar el ancho de banda, aumenta el coste de transmisión de la señal aunque disminuye la distorsión y la posibilidad de ocurrencia de errores.

Para un ancho de banda determinado es aconsejable la mayor velocidad de transmisión posible pero de forma que no se supere un umbral de tasa de errores. Para un ancho de banda dado W, la mayor velocidad binaria de transmisión posible es 2W, pero si se permite codificar más de un bit en cada ciclo, es posible transmitir mayor cantidad de información.

La formulación de Nyquist nos dice que aumentado los niveles de tensión diferenciables en la señal (*M*), es posible incrementar la cantidad de información transmitida.

$$C = 2W \log_2 M \quad \text{(bps)} \qquad (2.1)$$

El problema de esta técnica es que el receptor debe ser capaz de diferenciar más niveles de tensión en la señal recibida, cosa que es dificultada por el ruido. De hecho el mayor inconveniente que se presenta en el canal de transmisión es el ruido, el mismo que se torna más crítico cuando se incrementa la tasa de transmisión.

Shannon luego de realizar innumerables estudios propuso la fórmula que relaciona la potencia de la señal (*S*), la potencia del ruido (*N*), la capacidad del canal (*C*) y el ancho de banda (*W*).

$$C = W \log_2\left(1 + \frac{S}{N}\right) \quad \text{(bps)} \qquad (2.2)$$

Esta capacidad es el máximo volumen teórico de cantidad de transmisión, pero en la realidad, este valor es menor debido a que únicamente se ha tomado en cuenta el ruido térmico. La máxima velocidad de transferencia de datos entre dos extremos de la red, no sólo lo impone la infraestructura física de los enlaces, sino también los

flujos procedentes de otros nodos que comparten los enlaces de la ruta en cuestión.

2.3.1 MEDIOS DE TRANSMISIÓN

El medio de transmisión de las señales limita las componentes de frecuencia a las que puede ir la señal, permitiendo sólo la transmisión de cierto ancho de banda. Entre los medios de transmisión más utilizados se encuentran el par de cobre, el cable coaxial, la fibra óptica, y el aire.

2.3.1.1 Par trenzado

El medio más utilizado para circuitos de comunicaciones es el par de cobre, cuyo ancho de banda depende del grosor y de la longitud del cable. Una aplicación típica del par de cobre es la línea telefónica convencional que tiene un ancho de banda de 3.1 kHz [7]. Técnicas de codificación y modulación digitales optimizan este valor de ancho de banda, permitiendo velocidades de transmisión que van desde 28.8 kbps para un enlace *dial-up* con módems V.34, hasta velocidades en el orden de pocos Mbps sobre cortas distancias como en el caso de las líneas ADSL.

Para obtener un mayor aislamiento de ruido (y por tanto, mayor ancho de banda), se trabaja actualmente con alambres de cobre trenzados. Este tipo de cable se lo conoce como par trenzado. Estos pueden ser de dos clases: cables trenzados no blindados (UTP, *Unshielded Twisted Pair*) y cables trenzados blindados (STP, *Shielded Twisted Pair*).

La Asociación de Industrias Electrónicas (EIA, *Electronic Industries Alliance*) ha definido cinco categorías de par de alambre trenzado no blindado para uso en telefonía y transmisión de datos. Estas categorías son:

- Categoría 1: Par básico de alambre trenzado para uso en telefonía convencional (PSTN). No es recomendable su uso para transmisión de datos. [63]
- Categoría 2: Cuatro pares de alambre sólido, no blindados. Se utiliza en transmisión de datos hasta 4 Mbps. [63]

- Categoría 3: Posee una impedancia de 100 ohms y ancho de banda de 10 MHz que le permite soportar velocidades de transmisión de hasta 16 Mbps. Definido es la especificación EIA/TIA 568-A. Generalmente se utiliza sobre redes Ethernet. [18]

- Categoría 4: Posee una impedancia de 100 ohms y ancho de banda de 16 MHz que le permite soportar velocidades de transmisión de hasta 20 Mbps. Definido es la especificación EIA/TIA 568-A. Se utiliza sobre redes Token Ring de 16 Mbps. [18]

- Categoría 5: Posee una impedancia de 100 ohms y ancho de banda de 100 MHz que le permite soportar velocidades de transmisión de hasta 155 Mbps. Se utiliza en redes ATM y Ethernet de altas velocidades. [18]

En el cable blindado STP, el par de cobre trenzado es colocado en el interior de una delgada cubierta metálica, similar a una chapa de aluminio, y luego ésta es protegida por plástico. Este blindaje provee un mejor aislamiento eléctrico de las señales que cursan el par trenzado. El par de alambre trenzado blindado es menos susceptible a interferencia eléctrica causada por equipos o alambres cercanos. Este tipo de alambre puede cursar datos a mayores velocidades que los cables UTP. La desventaja de los cables de par trenzado blindado es que son más costosos y pesados que los UTP, presentando mayor dificultad para su manejo e instalación.

2.3.1.2 Cable coaxial

Está construido de algunas capas de materiales alrededor de un núcleo central, como se ilustra en la figura 2.10. El conductor central es frecuentemente un alambre de cobre, aún cuando en ocasiones es construido en aluminio. Éste es rodeado por un aislamiento, típicamente construido de alguna clase de plástico. A veces se deja espacios en el cable para mantener el conductor central separado de la cubierta o escudo, y en este caso, el material de aislamiento es aire o un gas inerte. Por fuera del aislamiento está el escudo o cubierta, el cual también es un conductor, típicamente es un alambre de cobre fino y trenzado. La cubierta es rodeada por el aislamiento exterior, el mismo que la mayor parte de veces es plástico que provee protección física al cable.

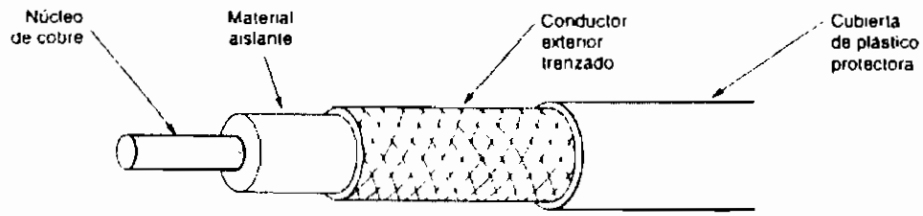

Figura 2.10 Estructura de un cable coaxial [10]

Las clases de cable coaxial más utilizadas son: cable coaxial de banda base y de banda ancha.

- Coaxial de Banda Base: Tiene una impedancia de 50 ohms, se usa comúnmente para transmisión digital, específicamente sobre redes LAN 10Base5 y 10Base2. En cables de 1 km, es posible velocidades de 1 a 2 Gbps. [10]

- Coaxial de Banda Ancha: Tiene una impedancia característica de 75 ohms, se usa comúnmente para transmisión analógica, por ejemplo para transmisión por cable, en la cual pueden ser transmitidos más de 50 canales de televisión sobre un sólo cable coaxial. En enlaces punto a punto, se puede obtener una razón de datos de 500 Mbps con un ancho de banda de 350 MHz, utilizando repetidores separados una distancia de 1 a 10 Km [18]. La industria de la telefonía usa cable coaxial en áreas donde la densidad de la población es muy alta. Un cable coaxial puede cursar sobre 10800 conversaciones de voz, con amplificadores ubicados en cada milla. [63]

2.3.1.3 Fibra óptica

La fibra óptica es una tecnología que ha tenido un gran despliegue en los últimos tiempos, aún cuando ha sido utilizada como medio de transmisión para portadoras comunes durante años.

La fibra óptica es una fibra de vidrio o plástico muy delgado de alta pureza. El corazón de vidrio (o plástico) en el centro (*core*) provee la capacidad de transmitir información. Este centro es recubierto por otro tipo de vidrio (o plástico) conocido como blindaje (*cladding*), el cual es reflectivo y actúa como un espejo hacia el

centro. El *cladding* tiene un recubrimiento de protección que usualmente es de plástico.

Se utilizan dos tipos primarios de fibra óptica: el uno es conocido como fibra óptica monomodo y usa una fibra con un núcleo de aproximadamente 5 μm de diámetro, el otro tipo de fibra es conocido como fibra óptica multimodo y su núcleo tiene aproximadamente 62.5 μm de diámetro.

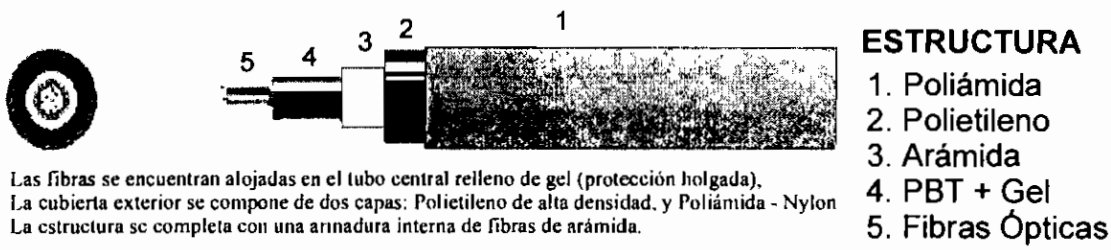

Figura 2.11 Estructura de una fibra óptica. [19]

Una de las más notables características de una fibra óptica es su gran ancho de banda. Una sola fibra óptica monomodo puede cursar datos a velocidades de transmisión de 2 Gbps sobre decenas de kilómetros de distancia bajo condiciones de laboratorio [18]. En aplicaciones prácticas, velocidades en el orden de 135 Mbps son conseguidas en distancias de aproximadamente, más o menos, 40 millas. Para distancias largas, pueden insertarse repetidoras entre las fibras para regenerar la señal. Para distancias cortas, pueden alcanzarse velocidades de 1.7 Gbps o más.

Las fibras ópticas multimodo trabajando en la ventana de 850 nm tienen un ancho de banda de 160 a 600 MHz/km, en tanto que trabajando en la ventana de 1300 nm alcanzan un ancho de banda de 200 a 1000 MHz/km.[20]

2.3.2 ANCHO DE BANDA Y LAS APLICACIONES

A las nuevas aplicaciones multimedia se las conoce como devoradoras de ancho de banda debido a que demandan gran capacidad del canal de comunicación. La tabla 2.4 muestra la tasa de transmisión necesaria para el funcionamiento de varias aplicaciones.

Aplicación	Tasa de Transmisión
Telefonía con calidad de audio	64 Kbps
Aplicaciones simples compartidas	100 Kbps
Videoconferencia	128 Kbps a 1 Mbps
Video MPEG	154 Mbps
Imágenes	8 a 100 Mbps
Realidad Virtual	> 100 Mbps

Tabla 2.4 Tasas de transmisión de varias aplicaciones. [21]

2.4 CONFIABILIDAD

En general se concibe como una propiedad del sistema de transmisión en su conjunto. En el caso aquí analizado, se puede considerar como la "tasa media de error" de la red.

Diversos factores pueden afectar a la confiabilidad: por ejemplo *routers* mal configurados o de bajas prestaciones, que pueden alterar el orden de recepción de los paquetes en destino o provocar pérdidas de aquellos; exceso de tráfico, que ocasiona congestión en la red; insuficiente espacio de almacenamiento en los nodos, etc.

En cualquier caso, TCP corrige estas deficiencias basado en retransmisiones, lo que se traduce en obligar al emisor a disminuir su velocidad de envío, de acuerdo con algoritmos de reducción de la congestión, aunque la causa no fuera la congestión real en la red provocada por el exceso de tráfico.

En el caso de aplicaciones de voz y video basadas en UDP, la falta de fiabilidad causa distorsión en las señales analógicas que se reproducen en destino. En último término, la falta de fiabilidad determina una red de baja calidad, que puede llegar incluso a no estar disponible en determinados momentos.

La disponibilidad general de una red se define como la disponibilidad agregada de todas las posibles rutas entre todos los *routers* del *backbone*. Se suele medir en fallos de rutas durante periodos de tiempo fijos (un mes, tres meses, etc.). Una red bien diseñada debe estar disponible cerca al 100 %, por ejemplo 99.7 %, lo cual

corresponde a un período de fallo de 129.6 minutos en un mes. Aún cuando este valor aparenta ser bajo, podría ser fatal para aplicaciones extremadamente críticas que demandan alta disponibilidad; para este caso existen proveedores de servicio que ofrecen disponibilidad del 99.9999 % conocida como la regla de los "seis nueves", equivalente a un período de fallo de 2.59 segundos en un mes.

Se debe tener en cuenta que todos esos factores no existen de forma aislada, sino que están fuertemente relacionados entre sí, pues cabe recordar que el Internet se compone de un variado conjunto de *routers* y enlaces de transmisión. Los *routers* reciben datagramas IP; según los procedimientos de encaminamiento, determinan el enlace de salida para el siguiente salto (*hop*) y colocan cada paquete en la cola de salida del enlace seleccionado. Los enlaces tienen unas características de retardo, ancho de banda y fiabilidad, inherentes al medio de transmisión. Si el nivel de tráfico excede el ancho de banda correspondiente a su enlace durante un tiempo prolongado, la calidad del servicio se degrada: las colas de salida en el *router* asociadas con el enlace saturado comienzan a crecer, provocando retardos adicionales de tránsito. Cuando los *buffers* del nodo se llenan, entonces el *router* empieza a descartar los nuevos paquetes, con lo que disminuye el rendimiento de la red. A su vez, esto obliga a los mecanismos de control de flujo a que disminuyan la velocidad de entrega de paquetes a la red y se evite la congestión por pérdidas, lo que reduce el ancho de banda efectivo para la aplicación correspondiente. La deficiente calidad de servicio puede deberse a otras causas. Por ejemplo, si los protocolos de encaminamiento son inestables, los *routers* pueden verse obligados a alterar la selección del siguiente salto (modificación de las tablas de encaminamiento), dando lugar a que los flujos extremo a extremo tomen rutas divergentes, lo que origina el aumento de la fluctuación y también una mayor probabilidad de entrega desordenada de paquetes, lo que reduce aún más la fiabilidad.

En el Internet original estos problemas de calidad de servicio eran menos importantes, o al menos así lo percibían sus usuarios. El patrón de tráfico era el típico de datos, con distintas características y requisitos, pero para unas aplicaciones que toleraban bien las posibles deficiencias; tanto las aplicaciones diferidas (correo electrónico, grupos de Noticias) como las interactivas, bien sean

del tipo transferencia masiva (FTP) como del tipo impulsivo o por ráfagas (Telnet). Además, al no haberse desarrollado el WWW, todas aquellas aplicaciones iniciales tenían unos modestos requisitos de ancho de banda, por el contrario, en el actual Internet surgen diariamente nuevas aplicaciones multimedia, devoradoras del ancho de banda y con una tendencia creciente a la inclusión de servicios en tiempo real (refiérase a la tabla 2.4).

2.5 DIFERENTES TIPOS DE TRÁFICO

El tráfico puede ser agrupado en cuatro grandes categorías: [6]

- Tráfico interactivo, el cual necesita sincronización entre el emisor y el receptor, por tanto, la latencia es una consideración importante para estos sistemas.

- Tráfico que demanda gran ancho de banda, el cual puede utilizar grandes *buffers* para disminuir la pérdida de datos o la variación en el retardo.

- Tráfico informativo sensitivo al retardo de la información, proveniente de aplicaciones cuya carga útil no puede ser encolada durante un intervalo de tiempo considerable, tal como la verificación de una tarjeta de crédito o el *stock* de un comercio.

- Tráfico tradicional, correspondiente al resto de aplicaciones que se enmarcan dentro del actual modelo *best effort*.

Se puede hacer alguna afirmación, analizando los requerimientos y conducta de diferentes tipos de tráfico, que permita su clasificación en diferentes clases de servicio. Aquellos que requieren atención especial pueden beneficiarse de algún tipo de reservación, la misma puede ser suministrada en forma administrativa o dinámica.

La tabla 2.5 resume diferentes tipos de aplicaciones y los servicios de red que demandan.

Aplicación	Ancho de Banda	Tiempo de Duración	Ráfagas de datos	Sensitividad al Retardo	Sensitividad al *Jitter*
Voz	Baja	Baja	Baja	Alta	Media
Video continuo	Alta	Alta	Baja	Baja	Alta
Video Interactivo	Alta	Alta	Media	Alta	Alta
Aplicación Compartida	Baja-Media	Media	Alta	Media	Baja
Datos	Baja-Media	Baja-Media	Alta	Baja	Baja

Tabla 2.5 Tipos de Aplicaciones y Requerimientos de la Red [6]

Antes de estudiar la identificación y manipulación de los diferentes tipos de tráfico a través de una red habilitada con Calidad de Servicio, se considerará las necesidades específicas de cada uno.

2.5.1 VOZ

El tráfico de voz requiere relativamente poco ancho de banda, el cual está en función de la frecuencia de muestreo de la señal analógica y de los bits que se utilicen por cada muestra. En cambio, el tráfico de voz es sensitivo al retardo y a las pérdidas, las cuales se manifiestan en forma de vacíos en la conversación.

Al momento del envío, presenta una tasa de transmisión baja y relativamente constante. Consecuentemente el tamaño del mensaje es pequeño, entre 4 y 200 bytes, lo cual permite una rápida transmisión de volúmenes pequeños de tráfico. Si las aplicaciones de voz utilizaran un paquete de 1500 bytes, tal como un datagrama IP, un flujo de voz de 16 kbps significaría que el emisor transmite sólo 1,365 paquetes por segundo, lo cual es absurdo. [6]

Se debe tener en cuenta además, que la voz contiene un 60 % de silencio, lo cual implica un tratamiento especial en el lado del receptor que asegure al oyente una reproducción eficiente.

Una de las principales razones para usar una red de paquetes es optimizar el ancho de banda mediante la compartición del enlace. Esto ocurre también en un circuito telefónico tradicional, a pesar de tener menor flexibilidad. Un enlace telefónico normal tiene una tasa de transmisión de 64 kbps (G.711),

correspondiente a 8000 muestras por segundo y 8 bits por muestra. Los enlaces internacionales a menudo utilizan los 64 kbps para transportar dos circuitos de 32 kbps (G.726). El uso de circuitos de 32 kbps no afecta mucho la calidad del sonido, pero algunos enlaces van más lejos y transportan 4 circuitos sobre el enlace. Estos circuitos de 16 kbps tienen una perceptible, aunque aceptable, degradación de la calidad del sonido.

Los sistemas de voz sobre IP típicamente trabajan entre 4 y 15 kbps por circuito. Los mejores sistemas de compresión de voz sobre IP que se tienen actualmente (G.729 y G.723.1) introducen entre 20 y 40 milisegundos de retardo en el extremo del transmisor; en el lado del receptor se agrega algo menos de 100 milisegundos debido al almacenaje de datos en los *buffers* para contrarrestar el *Jitter*. [6]

En cambio, los tradicionales sistemas telefónicos de conmutación de circuitos introducen tráfico en la red a una tasa de bits constante al mismo instante en que se hace el muestreo.

Retardos menores a 150 ms son aceptables para la mayoría de aplicaciones de voz. Retardos entre 150 y 400 ms son aceptables para comunicaciones de larga distancia. Mientras, retardos superiores a 400 ms son inaceptables para la mayoría de aplicaciones.

Retardos menores a 250 milisegundos son generalmente aceptables para VoIP, pues se encuentran en el límite de la tolerancia del oído humano. Una llamada local presenta un retardo entre 10 y 30 milisegundos, una tasa de transmisión de 64 kbps y además no presenta *Jitter*. Una llamada transatlántica ofrece características similares, con un retardo entre 100 y 200 milisegundos [6]. En el Anexo 1 se da a conocer información sobre niveles aceptables e inaceptables de retardo, *jitter* y pérdida para aplicaciones interactivas.

La compresión es una fuente importante de retardo porque se debe esperar hasta que exista suficiente información en los *buffers*. Los algoritmos de compresión trabajan buscando entre los datos secuencias comprimibles similares. Generalmente, mientras más grande es la tabla usada para la compresión, mayor es el radio de compresión.

Los efectos de la compresión pueden ser resumidos en dos observaciones: la compresión genera un incremento en el retardo debido al procesamiento, y un decremento en el ancho de banda; a la vez, genera un incremento en el ancho de banda y una reducción en el retardo debido al tamaño pequeño que tienen los paquetes. Enlaces lentos y compresión ágil, por tanto, hacen que la compresión valga la pena.

Otro retardo importante que se debe considerar es el retardo producido en el empaquetamiento. Este retardo depende de la carga útil del paquete y de la tasa a la cual se hace el muestreo del tráfico. Por ejemplo, en ATM, el primer byte que arriba tiene que esperar a los 47 restantes de la carga útil antes de que se añadan los 5 bytes de cabecera y ser transmitido. Si se supone que la aplicación hace un muestreo del tráfico de voz a 64 kbps, u 8000 muestras de 8 bits en un segundo, se tiene un retardo de por lo menos 5.8 ms para 47 bytes, tiempo ocurrido entre la llegada de la primera muestra hasta el arribo de la última. [6]

Las pérdidas de paquetes son muy importantes, perder un paquete cada minuto no es causa de alarma, sin embargo perder dos paquetes consecutivos puede indicar un real problema. La figura 2.12 muestra la relación entre la tasa de pérdidas de los paquetes y la calidad de la voz, asumiendo 5 niveles de calidad propuestos por la ITU.

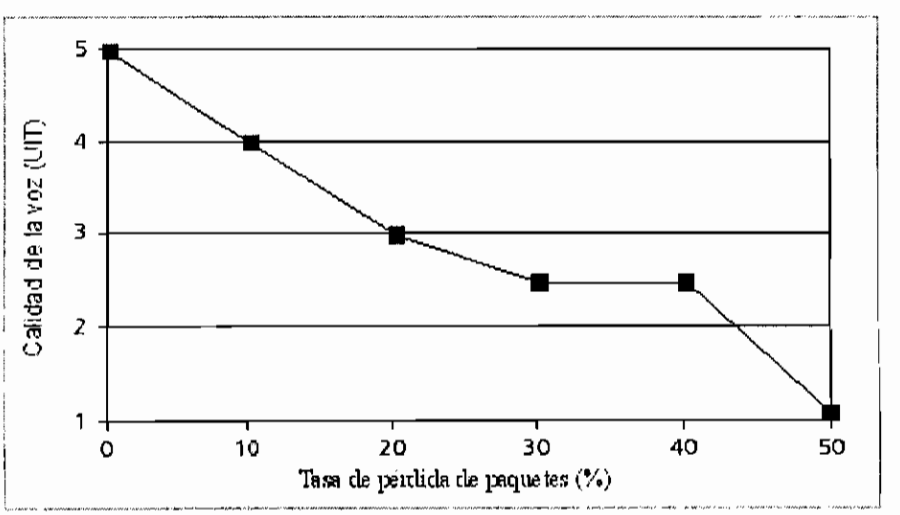

Figura 2.12 Calidad de la voz vs. Tasa de pérdida de paquetes. [22]

Se presentan a continuación algunos ejemplos de conexiones de voz y sus características.

Conexión	Pérdida de Paquetes	Tasa de Transmisión	Latencia	*Jitter*
Línea Telefónica Normal (llamada local)	0%	56 kbps	10-30 ms	0%
Enlace Trasatlántico submarino	0%	56 kbps	100-200 ms	0%
Típica conexión GSM (inalámbrica)	5-15%	13 kbps	100-300 ms	50%
Aplicación Telefónica de Internet (asumiendo un buen ISP y módem *dial-up*)	5-20%	20 kbps	120-240 ms	150%
Aplicación Telefónica de Internet Inaceptable	10-35%	20 kbps	100-300 ms	300%

Tabla 2.6 Comparación de varias conexiones de voz. [6]

La retransmisión de paquetes de voz, efectuada por TCP, retarda el envío de paquetes; dado que los oyentes usualmente no detectan la pérdida de los paquetes de voz si ésta es menor a 5% de los datos descartados, el tráfico de voz típicamente viaja sobre UDP en lugar de TCP; consecuentemente, UDP permite que la aplicación sacrifique envío garantizado a cambio de menor retardo.

CONVERSACIÓN INTERACTIVA

Los factores humanos desempeñan un rol significativo en la tolerancia al retardo. Los usuarios tienen un conjunto de expectativas sobre la conducta de una sesión de voz *full duplex*, tornándose frustrados por la latencia.

Cuando se habla de conversación, se refiere a una interacción entre humanos. Esto es a menudo tráfico de voz, pero muchas aplicaciones de datos están tornándose conversacionales. Sistemas de trabajo concurrentes que permiten a grupos distribuidos correr la misma aplicación debe ofrecer un tiempo de respuesta adecuado. Este caso se da aún cuando existe un circuito separado para tráfico de voz. Tales sistemas son muy sensitivos a la latencia.

Los factores humanos también intervienen en algunas aplicaciones de datos típicas. La latencia puede manifestarse en una forma difícil de ser medida; por ejemplo, un enlace lento puede causar que la productividad de la reunión decaiga cuando los empleados intenten solucionar el problema, o puede desperdiciarse tiempo debido a que los usuarios están distraídos en efectos inusuales producto del mismo retardo.

2.5.2 VIDEO

El video es una aplicación que consume gran ancho de banda. El ancho de banda depende de la resolución de la imagen, la profundidad de colores y la tasa a la cual se presentan las imágenes. Una aplicación de alta calidad puede utilizar 640x480 píxeles de resolución, 24 bits para definición de colores y una velocidad de 30 imágenes por segundo, obteniéndose una tasa de transmisión de 221.184 Mbps.

Con la variación de estos parámetros el ancho de banda necesario puede disminuir considerablemente. Por ejemplo, con una resolución de 320x240 píxeles, 8 bits para la definición de colores y una tasa de 15 imágenes por segundo, la tasa de transmisión necesaria disminuye a 9.216 Mbps. Sin embargo, la demanda de ancho de banda es significativa y si el video debe utilizar enlaces WAN, los costos ascenderían fuertemente.

Para disminuir el consumo de ancho de banda se han desarrollado diversas técnicas de compresión, las cuales pueden tolerar, o no, pérdidas de información. Cuando se utiliza compresión sin pérdidas, se alcanza una relación de compresión máxima de 2:1. En cambio, con técnica de compresión con pérdidas de información se puede obtener relaciones de compresión de hasta 300:1, posibilitando el desarrollo de aplicaciones de videoconferencias para una variedad de tasas de transmisión desde un simple enlace de módem a 28.8 kbps [6].

La compresión del tráfico de video se realiza básicamente de dos maneras:

- *Interframe*
- *Intraframe*

2.5.2.1 Compresión *Interframe*

En *Interframe* el tráfico consiste en la transmisión de tramas claves (*key frames*), que describen la imagen entera, y de tramas intermedias (*delta frames*), que describen los cambios o deltas de la trama clave.

Un flujo despejado de video comprimido muestra ráfagas de tráfico en cada trama clave, y relativamente menor tráfico para las tramas intermedias, como se ilustra en la figura 2.13.

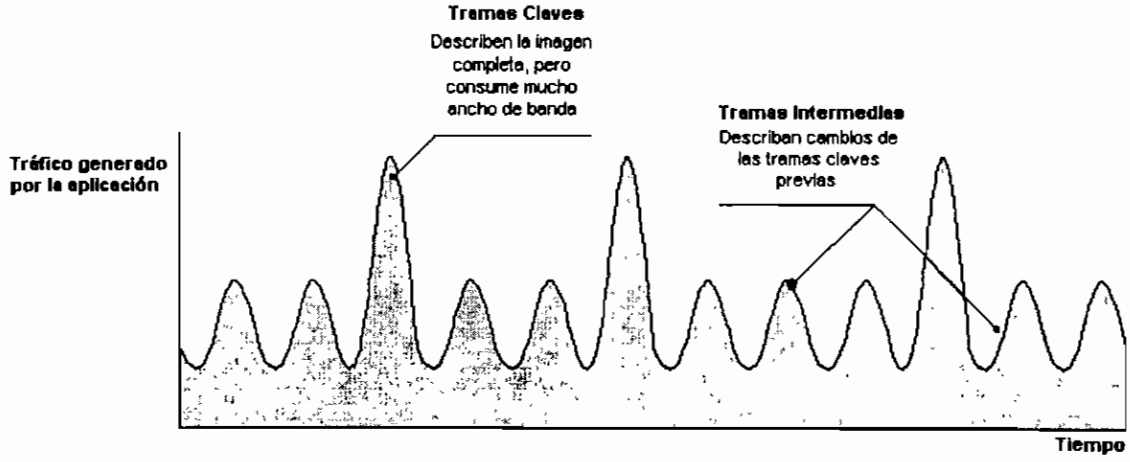

Figura 2.13 Tramas Claves y Tramas Intermedias en una red descongestionada.[6]

Las tramas claves permiten al receptor sincronizarse periódicamente, ya que el emisor y el receptor no pueden estar pasivos por mucho tiempo. Si la red llega a congestionarse, los paquetes pueden ser almacenados en *buffers*, o pueden ser descartados. Diferentes partes de una imagen de video son enviadas a diferente tiempo, y el receptor debe almacenarlas para ensamblar la imagen completa. Algunos paquetes pueden arribar más temprano de lo esperado, mientras otros pueden hacerlo tan tarde que ya no serán útiles, pues la imagen a la que pertenecían ya fue desplegada, como se muestra en la figura 2.14.

Esto significa que, para imágenes de gran volumen en condiciones de congestión, algunos paquetes pueden arribar algo tarde y serán descartados por el receptor. El retardo en el envío de paquetes puede ocasionar un recorte en las tramas clave, degradando la imagen, como se aprecia en la figura 2.15

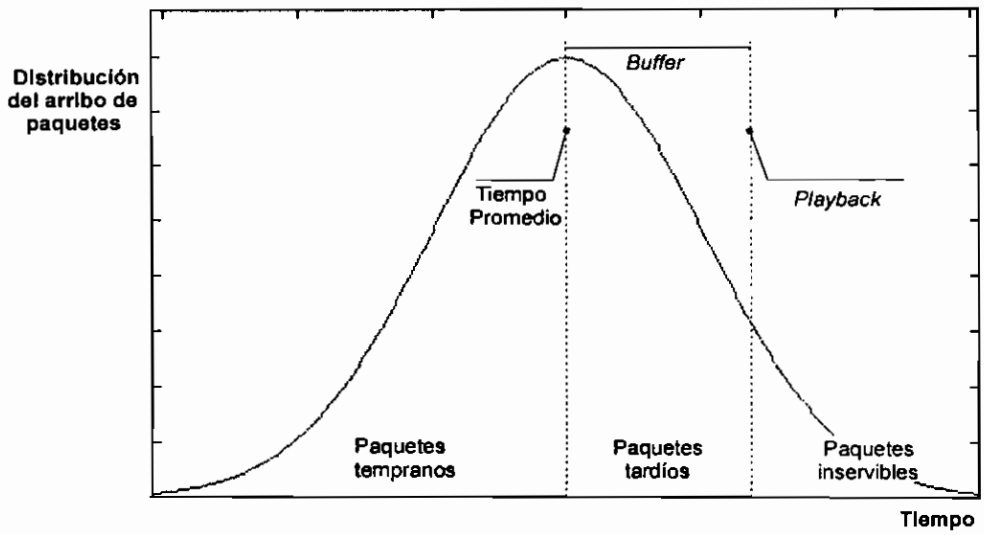

Figura 2.14 Distribución del arribo de paquetes y retardo promedio en *buffers*. [6]

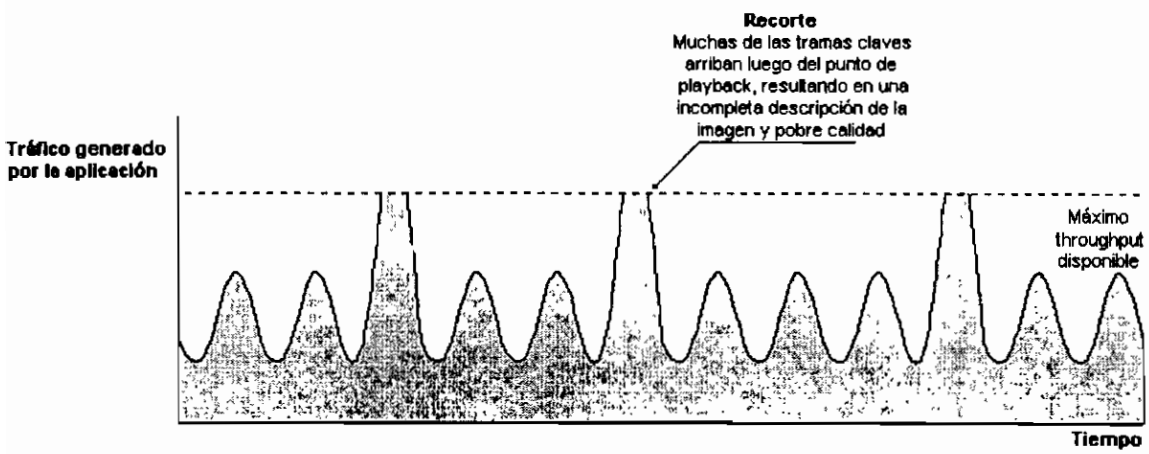

Figura 2.15 Recorte de la Tramas Clave cuando la capacidad es limitada.[6]

Los efectos de este recorte pueden ser reducidos cambiando los parámetros de la aplicación tales como el tamaño de las tramas clave (reduciendo el tamaño de la pantalla o la profundidad del color) o incrementando el retardo entre la recepción del tráfico y su posterior reproducción. Si se altera el tamaño de la trama clave, la imagen tomará más tiempo en estabilizarse cuando se produzca un cambio significativo, pero habrá menos vibración en la transmisión. Por otro lado, incrementando el retardo entre la recepción y la reproducción, se introducirá más latencia en la imagen.

2.5.2.2 Compresión *Intraframe*

La compresión *Intraframe* solamente utiliza la información de una trama particular. Se presenta básicamente en las tramas clave, aunque también actúa sobre los píxeles de las tramas intermedias que mantienen información luego de la compresión *Interframe*.

Las siguientes son las principales técnicas de compresión *Intraframe*: RLE (*Run Length Encoding*), JPEG (*Joint Photographic Experts Group*) y VQ (*Vector Quantization*) [23].

Existe una gran variedad de algoritmos de compresión disponibles a usuarios finales, los cuales pueden requerir hardware dedicado. A continuación se analizan los más importantes:

- MPEG1 utiliza una tasa de 1.5 Mbps, para enviar imágenes de 640x480 píxeles de resolución a una velocidad de 30 tramas por segundo [23]. La compresión en el emisor no se realiza en tiempo real a diferencia de la descompresión en el receptor.

- MPEG2 utiliza una tasa de 4 a 9 Mbps, necesaria para un envío de alta calidad. Tanto la compresión y la descompresión se efectúan en tiempo real. [23]

- MPEG4 utiliza un ancho de banda de 64 kbps. [23]

- H.261, la recomendación de compresión para la norma de videoconferencia H.323, utiliza un ancho de banda de *N* x 64 kbps. [23]

Una típica aplicación de video tal vez utilice un canal de control TCP y dos canales de datos UDP, uno para voz y otro para imagen. Por consiguiente, el establecimiento de una sesión de videoconferencia será complejo, requiriéndose dispositivos que reconozcan la relación entre el canal de control y los canales de datos. El tráfico del canal de control puede ser priorizado para permitir a los dispositivos ajustar su tasa de transmisión cuando el canal de control indique congestión, por mencionar un ejemplo.

El tráfico de video no comprimido trabaja a una velocidad constante de transmisión (CBR, *Constant Bit Rate*). El alto precio de los circuitos WAN y el impacto significativo de la difusión de video han hecho que la mayoría de tráfico de video corra normalmente a velocidades de transmisión variables (VBR, *Variable Bit Rate*), debido a la compresión. Este tipo de video emplea generalmente el máximo tamaño posible de paquete para enviar información, es decir cada datagrama tendrá un tamaño igual al de la unidad de transmisión máxima (MTU, *Maximum Transmission Unit*). Por otro lado, el tráfico de video a una velocidad constante de bit (CBR) tiene una alta tasa de transmisión y baja variabilidad.

En un entorno práctico, la capacidad varía constantemente, tornando difícil la tarea del transmisor y receptor de evaluar la capacidad de la red para seleccionar adecuadamente el tamaño de la trama, la profundidad del color, y la configuración del *buffer* en una red *best effort*. Para resolver este problema se garantiza una porción de ancho de banda para proporcionar algún grado de determinismo a través de "carga controlada", en la cual el retardo de la red se mantiene entre un rango predecible, o "carga garantizada", en la cual el retardo y el *Jitter* de la red se mantienen relativamente estáticos.[1]

2.6 RED CONVERGENTE

Una Red Convergente es aquella que puede satisfacer los requerimientos de latencia, *Jitter*, capacidad, confiabilidad y tiempos de duración de la sesión que imponen varios tipos de aplicaciones. [6]

2.6.1 CARACTERÍSTICAS DE UNA RED CONVERGENTE

Una red de este tipo presenta varias características:

- Medio físico común
- Protocolos comunes en la pila
- Conmutación en el borde

[1] Los servicios best effort, carga controlada y garantizado se analizan en detalle en el capítulo 5

- Núcleo de alto direccionamiento
- Integración de la aplicación

Una red convergente comparte el medio entre varias aplicaciones y tipos de tráfico. Generalmente, éste será par trenzado de categoría 5, el cual permitirá migrar de enlaces de 10 Mbps a enlaces de 100 Mbps. Para efectos operacionales, será excelente la consolidación de voz y datos sobre una sencilla planta de cableado.

Debido a que esta red contiene varios tipos de tráfico, también confiará en un conjunto de protocolos que le sirvan como punto de concentración alrededor de los cuales se desarrollaran servicios y aplicaciones. Debido a su amplia difusión y capacidad para soportar una variedad de aplicaciones, el protocolo elegido es el IP.

Los puertos conmutados reducen los efectos de la congestión en medios de acceso lentos, tornando a la red convergente ideal para aplicaciones sensibles a la latencia tal como la voz. Además con relación a la seguridad, la conmutación reduce el riesgo de que la información importante sea interceptada. Desde una perspectiva de Calidad de Servicio, una arquitectura de red conmutada en el borde, permite una simulación de circuito extremo a extremo a través de la capa de red.

Para obtener los beneficios de un núcleo rápido mientras se hace la capacidad accesible a varios dominios administrativos, un direccionamiento público y un esquema de enrutamiento deben llamar a un mecanismo de direccionamiento de circuito privado. Estos mecanismos incluyen sistemas tales como la Conmutación de Etiquetas de Cisco (*Cisco's Tag Switch*ing) y el Protocolo de Administración de Flujo Ipsilon (IFMP, *Ipsilon Flow Management Protocol*), cuyas herencias pueden ser apreciadas en el Protocolo Múltiple de Conmutación de Etiquetas (MPLS, *MultiProtocolo Label Switching*). Un núcleo de alto rendimiento y públicamente accesible como el MPLS es llamado algunas veces "capa 2.5", ya que posee características de una capa de enlace pero presenta una interfaz de capa 3 para dispositivos aledaños.

El núcleo de una red convergente enfatiza en el direccionamiento del tráfico basado en una clasificación "gruesa", es decir se manejará relativamente pocas

clasificaciones discretas, tales como las cabeceras MPLS, las características de calidad de servicio de ATM, o la precedencia de la información IP.

Una red convergente presenta también integración de la aplicación combinando varios tipos de datos sobre modelos híbridos. Por ejemplo, una aplicación de correo electrónico (*e-mail*) puede incluir la habilidad para enviar mensajes de voz (*voicemail*); un sistema de video continuo (*streamed*) unidireccional puede soportar video interactivo de poco ancho de banda, útil para preguntas.

La mayoría de entornos en los cuales las Redes Convergentes ofrecen beneficios tangibles comparten estas características. Generalmente, habrá más de un tipo de tráfico y el potencial desarrollo de todavía más tipos de tráfico.

2.6.2 USOS PRÁCTICOS DE REDES CONVERGENTES

La convergencia de diferentes tipos de tráfico, como voz y datos, no implica que cada computador sea también un teléfono; en lugar de ello, la convergencia ocurre en una lugar específico donde los beneficios de una estructura común superan los costos de mantenimiento.

Una red de voz sobre IP puede utilizar microteléfonos IP con interfaces Ethernet para reducir el cableado redundante en la red. Esta es una solución de Voz sobre IP de teléfono a teléfono. Por otro lado, el conmutador telefónico de cada oficina puede distinguir entre una llamada interna o externa, utilizando una capacidad reservada de la red de datos para llamadas entre oficinas. Esta es una red PBX-PBX. El punto principal de esta solución es que el *gateway* de Voz sobre IP puede manejar varias conexiones, y además codificar, de circuito a paquete, el tráfico de una llamada para transmitirlo a través de una red de datos.

Un sistema PBX a PBX puede ofrecer características adicionales tales como manipulación de llamadas controladas por computador y conferencias.

Capítulo 3

MECANISMOS DE GESTIÓN DE TRÁFICO

CAPÍTULO 3

MECANISMOS DE GESTIÓN DE TRÁFICO

Los mecanismos de manipulación de tráfico, tanto como los de provisión y configuración de QoS, citados en el capítulo 1, se pueden encajar en dos grandes arquitecturas diferentes y a la vez interdependientes; la primera es la Arquitectura de Servicios Diferenciados, la cual será analizada en detalle en el capítulo 4, y la segunda es la Arquitectura de Servicios Integrados, estudiada en el capítulo 5.

Las dos arquitecturas nombradas utilizan un conjunto de mecanismos de control de tráfico, que mediante la manipulación del tráfico en los dispositivos de red, les permiten cumplir con su objetivo de proveer calidad de servicio. Estos mecanismos incluyen: 1) Clasificador de paquetes, 2) Control de admisión y de políticas, y 3) Organizador (*scheduler*) de paquetes o algunos otros mecanismos dependientes de la capa enlace.

El **control de admisión** determina si un nodo tiene la suficiente cantidad de recursos disponibles para suministrar un servicio de calidad de servicio. El **control de políticas** determina si un usuario tiene permiso administrativo para hacer el requerimiento de calidad de servicio.

El **clasificador de paquetes** y las interfaces de capa enlace, por ejemplo el organizador de paquetes, son los encargados de implementar el QoS deseado, permitiendo obtener del *router* un trato diferenciado con características específicas de retardo y ancho de banda.

Los **mecanismos de organización** y otros mecanismos de capa enlace que permiten administrar las colas, algoritmos de encolamiento y descarte de paquetes, son el objeto de estudio del presente capítulo.

3.1 INTRODUCCIÓN

El actual modelo *best effort* basado en el Protocolo de Internet IP no orientado a conexión ha sido exitoso debido a su alta flexibilidad y robustez, sin embargo, su debilidad principal es la incapacidad de entregar calidad en condiciones de congestión de la red.

El control de congestión se ha dejado a los *hosts* extremos de la red, los cuales mediante el protocolo TCP determinan la cantidad de tráfico que introducen en ésta dependiendo de las condiciones presentes, tal control de congestión ha funcionado bien por algún tiempo, pero hoy no es suficiente y se hace necesaria la incorporación de mecanismos de control en los *routers*. [31]

En la actualidad normalmente se configura el tamaño de la cola en unidades de paquetes y se envían éstos de acuerdo al orden de llegada, el primer paquete en ingresar será el primer paquete en salir, este tipo de encolamiento se denomina FIFO (*First Input First Output*); cuando la cola se ha llenado se procede a descartar todos los paquetes que arriban, esta técnica de descarte se denomina *tail drop*. El descarte de paquetes es el indicativo de congestión en el cual se basa el protocolo TCP para bajar la tasa de transmisión a la cual ingresa datos en la red.

Esta forma de control de congestión ha funcionado bien por mucho tiempo pero tiene dos considerables desventajas que impiden la entrega de calidad de servicio bajo las condiciones actuales. Estas desventajas son la tendencia a un estado permanente de colas llenas y la posibilidad de que un solo flujo monopolice el espacio de la cola.

La tendencia a un estado permanente de colas llenas es un efecto obvio pues no se notifica congestión sino hasta alcanzar el máximo tamaño de la cola. mientras ello ocurre todos los flujos intentan utilizar el máximo espacio posible. Debido a la característica de ráfagas que presenta el tráfico de Internet, cuando éstas se producen y encuentran las colas llenas se presenta una reducción de la tasa de transmisión de los flujos debido a la acción del protocolo TCP. sin embargo cuando la ráfaga ha pasado y no existe congestión se da una subutilización del

ancho de banda, por lo tanto la tendencia a colas llenas a más de introducir retardo puede ocasionar una disminución del ancho de banda.

La monopolización del espacio de la cola es a menudo el resultado de la sincronización y otros efectos de tiempo, este fenómeno no es deseado pues impide que otros flujos tengan cabida en la cola. Se hace entonces necesaria una administración ACTIVA de colas que contrarreste estos efectos.

3.2 ADMINISTRACIÓN ACTIVA DE COLAS

La administración activa de colas es una alternativa al típico encolamiento FIFO y descarte de paquetes *tail drop*.

3.2.1 CARACTERÍSTICAS

- Soluciona los dos problemas, la tendencia a colas llenas y la monopolización de la cola.

- Sirve para tráfico sensitivo a las notificaciones de congestión de la red. La notificación de congestión referida es el descarte de paquetes a la cual es sensible el tráfico TCP.

- Su funcionamiento básico es el descarte de paquetes antes que se sature la cola. Con ello se logra que el *host* extremo baje la velocidad de transmisión sin que se llene la cola posibilitando un estado estacionario de colas despejadas.

3.2.2 VENTAJAS

- Se reduce el número de paquetes descartados en los *routers*. La cola se diseña en función de la capacidad para manejar ráfagas de datos, es decir siempre tendrá espacio disponible que le permita soportar ráfagas de datos evitando la alta pérdida de paquetes que se produce en congestión. Además se debe señalar que un flujo TCP se recupera más fácilmente de la pérdida de un paquete que de la pérdida de una ráfaga de paquetes.

- Provee servicio interactivo de bajo retardo. El mantener un tamaño de cola promedio bajo permite disminuir el retardo.

- Se evita la conducta de monopolización de la cola, pues siempre existirá espacio disponible para otro flujo.

Se ha desarrollado varios algoritmos que permiten una administración activa de colas. Uno de ellos es el algoritmo de Detección Aleatoria Temprana RED (*Random Early Detection*) que trabaja en conjunto con varios algoritmos de encolamiento como WFQ (*Weighted Fair Queuing*), CBQ (*Class Based Queuing*), entre otros.

A continuación se presenta un análisis de estos algoritmos de encolamiento y de descarte de paquetes.

3.3 ALGORITMOS DE ENCOLAMIENTO

3.3.1 ENCOLAMIENTO FIFO

El encolamiento FIFO (*First Input First Output*) es el más utilizado y el más sencillo de implementar lo que implica un bajo costo, sin embargo estas importantes ventajas no son suficientes para administrar el Internet actual, el cual requiere otras características que este algoritmo no las puede ofertar, como la diferenciación del tráfico independientemente del orden de llegada de los paquetes.

3.3.1.1 Funcionamiento

Este algoritmo da servicio a los paquetes de acuerdo, únicamente, al orden de llegada; el primer paquete en arribar al dispositivo de red será el primero en salir del mismo. Cuando la red está congestionada los paquetes que ingresan a la interfaz de entrada del *router* se almacenan en su respectiva cola de entrada para su posterior direccionamiento a la interfaz de salida correspondiente, respetando siempre el orden de llegada. Esta función se efectúa normalmente mientras haya espacio disponible en la cola, cuando ésta se ha llenado se procede al descarte

de todo paquete que llega al dispositivo hasta que la cola esté disponible nuevamente. Este algoritmo está representado en la figura 3.1.

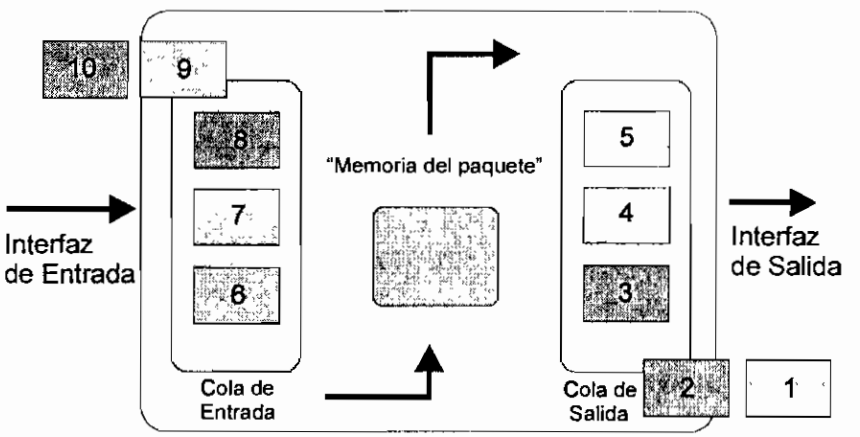

Figura 3.1 Encolamiento FIFO [24]

Mediante este algoritmo, el ancho de banda, la prontitud, y la asignación de espacio en cola para cada flujo de tráfico dependen únicamente del orden de arribo de los paquetes.

El retardo es el mismo para cualquier tipo de tráfico y es proporcional al tamaño de la cola; el retardo para un flujo es igual al tamaño de la cola dividido para la tasa de transmisión. Por ejemplo con un tamaño de cola de 1280 bytes y una tasa de transmisión de 2048 kbps el retardo aproximado será de 5 milisegundos.

3.3.1.2 Rendimiento

Debido a su simplicidad este algoritmo presenta algunas limitaciones:

- Sobredimensionamiento del efecto de congestión que causa grandes variaciones de rendimiento.

- Imposibilidad de asignar prontitud, ancho de banda y espacio en cola independientemente, frustrando una administración efectiva.

- No tiene protección contra flujos provenientes de fuentes que presentan un comportamiento inadecuado modificando el protocolo TCP en afán de obtener mayores recursos.

- Necesita coordinación entre todas las fuentes de tráfico para poder entregar un servicio diferenciado. Esta labor es bastante difícil en un entorno corporativo y prácticamente imposible en una red diversa y distribuida como Internet.

Estas limitaciones han llevado al estudio y desarrollo de nuevos algoritmos que contrarresten estos efectos, como por ejemplo el algoritmo de Encolamiento Justo Ponderado (WFQ, *Weighted Fair Queuing*).

3.3.2 ENCOLAMIENTO JUSTO PONDERADO (WFQ, *Weighted Fair Queuing*)

El objetivo cualitativo de este encolamiento es entregar un servicio justo a todos los flujos que pertenecen a una misma cola; para efectuar un análisis cuantitativo es necesaria la definición técnica del término "justo".

Considérese N usuarios que comparten una cantidad de un recurso independiente μ_{total}, siendo ρ_i el requerimiento de recurso de cada usuario y μ_i la asignación de recurso a cada usuario, existe una asignación justa o *fair* si: [25]

1. La asignación de recurso no es mayor que el requerimiento de recurso de cada usuario. Matemáticamente: $\mu_i \leq \rho_i$.

2. Ninguna otra forma de asignación que cumple la condición 1 tiene asignación mínima mayor. Es decir la asignación mínima a un usuario es la máxima posible.

3. La condición 2 se mantiene recursivamente verdadera, si se quita el usuario mínimo se reduce el recurso total, es decir: $\mu_{total} \leftarrow \mu_{total} - \mu_{min}$. Siendo μ_{fair} la asignación justa, se deduce fácilmente que:

$$\mu_i = MIN(\mu_{fair}, p_i) \qquad (3.1)$$

$$\mu_{total} = \sum_{i=1}^{N} \mu_i \qquad (3.2)$$

Este concepto de justicia se generaliza fácilmente a casos con múltiples recursos, además se debe notar que se asume a todos los usuarios con iguales derechos de obtener el recurso.

Prácticamente, ρ_i puede corresponder a la demanda de ancho de banda, espacio en cola y prontitud; si el recurso es ancho de banda o espacio en cola este valor se deduce claramente a partir del arribo de los paquetes, pero si se refiere a prontitud su valor no es tan claro, analizándose este caso en la siguiente sección.

El usuario asociado al paquete podría referirse a la fuente del paquete, al destino, a un par fuente – destino (conversación), o incluso al proceso dentro de la fuente. Cada tipo de asignación tiene sus desventajas: la asignación basada en la fuente de tráfico puede restringir a fuentes que utilizan considerable ancho de banda como el servidor de archivos; la asignación basada en el destino puede disminuir su ancho de banda efectivo al recibir información no útil procedente de fuentes con mal comportamiento; la asignación basada en procesos individuales de una fuente permite que el usuario inicie varias sesiones simultáneas apoderándose del ancho de banda; y la asignación basada en la conversación puede consumir excesivo ancho de banda al enviar muchos paquetes a diferentes destinos. El Encolamiento Justo Ponderado empleará usuarios o flujos basados en conversaciones (pares fuente – destino) pues presenta mejores características de seguridad y eficiencia con relación a los demás [25].

3.3.2.1 Funcionamiento

A continuación se explica el mecanismo utilizado por el algoritmo de encolamiento justo ponderado para asignar espacio en cola, ancho de banda y prontitud a los paquetes entrantes.

La asignación justa de espacio en los *buffers* se la obtiene fácilmente mediante el descarte del último paquete en ingresar al *router* perteneciente a la conversación con cola más larga.

La asignación justa de ancho de banda se lograría simulando el envío bit a bit de los paquetes de cada conversación en una forma secuencial y cíclica, como se ilustra en la figura 3.2.

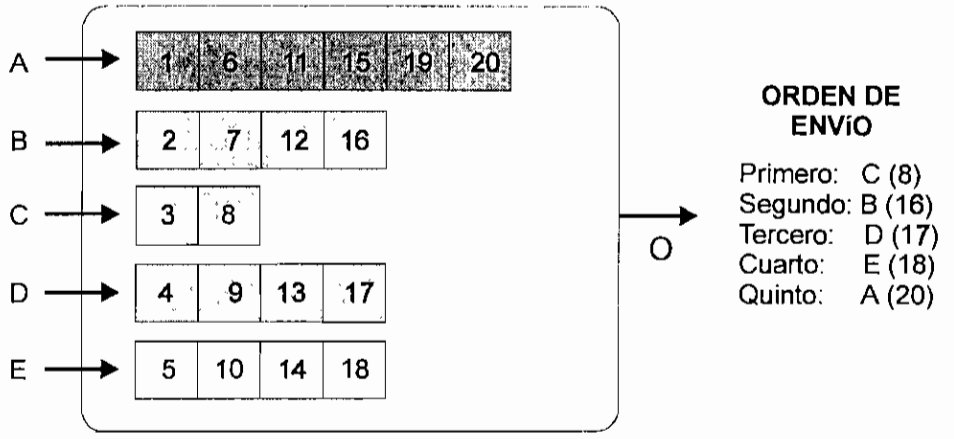

Figura 3.2 Funcionamiento del algoritmo de Encolamiento Justo Ponderado.[10]

Sea *R(t)* el valor correspondiente al número de vueltas completas al tiempo *t*, entendiéndose que una vuelta completa ocurre entre la transmisión del primero y segundo bit de una misma conversación; $N_{ac}(t)$ el número de conversaciones activas al instante *t*; y, μ la tasa de transmisión en la línea de salida del dispositivo de red, se deduce que [25]:

$$\frac{\partial R(t)}{\partial t} = \frac{\mu}{N_{ac}(t)} \qquad (3.3)$$

Si *P* es el tamaño de un paquete cuyo primer bit arriba a un instante t_o, se obtiene [25]:

$$R(t) = R(t_o) + P \qquad (3.4)$$

Sea t_i^α el instante de arribo del paquete *i* de la conversación α, S_i^α el valor de *R(t)* cuando empieza el servicio, F_i^α el valor de R(t) cuando finaliza el servicio, y P_i^α el tamaño del paquete *i*, se deduce a partir de la ecuación 3.4:

$$F_i^\alpha = S_i^\alpha + P_i^\alpha \text{ donde } S_i^\alpha = \text{MAX}(F_{i-1}^\alpha, R(t_i^\alpha)) \qquad (3.5)$$

Nótese que mientras la conversación está activa $R(t)$, $N_{ac}(t)$, F_i^α y S_i^α dependen solamente del tiempo de arribo del paquete y no de la tasa de transmisión. Una conversación está activa si $R(t) \leq F_i^\alpha$ siendo i = MAX(j t.q. $t_j^\alpha \leq t$).

Obviamente es irreal la transmisión bit por bit en dispositivos de capa red que manejan como unidad al paquete, sin embargo se puede adecuar este desarrollo si el orden de transmisión de los paquetes depende del valor que tenga F_i^α, transmitiendo el paquete con menor F_i^α. Aunque la transmisión paquete por paquete no tiene una asignación instantánea de ancho de banda justa, se la puede considerar así en periodos de tiempo relativamente largos.

Existen dos versiones de este algoritmo: preferente y no preferente. La diferencia radica en que la versión preferente considera para los cálculos al paquete que se está transmitiendo, es decir si arriba un paquete con menor F_i^α se detiene la actual transmisión y se transmite el nuevo paquete.

La asignación justa de prontitud no es explícita y para su explicación se debe introducir un parámetro no negativo δ que permite definir el valor B_i^α (3.6); este valor es el que determina el orden de envío de los paquetes, transmitiéndose el paquete con menor B_i^α.

$$B_i^\alpha = P_i^\alpha + \text{MAX}(F_{i-1}^\alpha, R(t_i^\alpha) - \delta) \quad \text{con } \delta \geq 0 \qquad (3.6)$$

En la ecuación 3.6 se debe analizar dos casos:

1. Si $R(t_i^\alpha) \leq F_{i-1}^\alpha$, no interfiere el valor de δ pues la conversación α está activa, habiendo bits encolados.

2. Si $R(t_i^\alpha) > F_{i-1}^\alpha$, la conversación no está activa y para su análisis se considera los dos casos extremos, δ=0 y δ=∞.

- Si δ=0 entonces $B_i^a = P_i^a + R(t_i^a)$, por lo tanto el envío del paquete *i* no depende de la historia.

- Si δ=∞ entonces $B_i^a = P_i^a + F_{i-1}^a$, por lo tanto el envío del paquete *i* depende solo de la finalización del envío del paquete anterior.

Con un valor intermedio de δ el envío del paquete *i* dependerá de la finalización de los paquetes anteriores de la misma conversación pero durante cierta ventana de tiempo.

Como se puede apreciar, el valor otorgado a δ define la prioridad del envío con prontitud de una conversación, presentándose los casos extremos en δ=0 y δ=∞, para indicar prioridad mínima y máxima respectivamente.

3.3.2.2 Rendimiento

La asignación justa de ancho de banda y espacio en cola está claramente definida, sin embargo la asignación de prontitud es un poco más difícil de entender; si se desea una mejor comprensión se sugiere revisar el ANEXO 2, el cual muestra el comportamiento de un flujo sensible al retardo como Telnet que comparte la cola de un *switch* con varios flujos FTP de alta demanda de ancho de banda.

Este tipo de encolamiento presenta las siguientes características:

- Asigna independientemente ancho de banda, prontitud y espacio en cola, posibilitando, por ejemplo, que un flujo FTP otorgue prioridad al ancho de banda y un flujo Telnet dé prioridad a la prontitud, basándose en los parámetros F_i^a y B_i^a respectivamente.

- Crea un *firewall*[1] que protege las fuentes con buen comportamiento de aquellas que intentan apoderarse de todos los recursos; por ejemplo, se da un menor retardo a fuentes Telnet que ocupan menos ancho de banda que una fuente FTP. Esta característica de protección (figura 3.3) habilita una nueva clase de algoritmos de control de congestión mediante los cuales la cantidad de recursos obtenidos de la red depende de la propia conducta de la fuente, la misma que puede optimizar su tasa de transmisión sin importar la conducta de las demás fuentes.

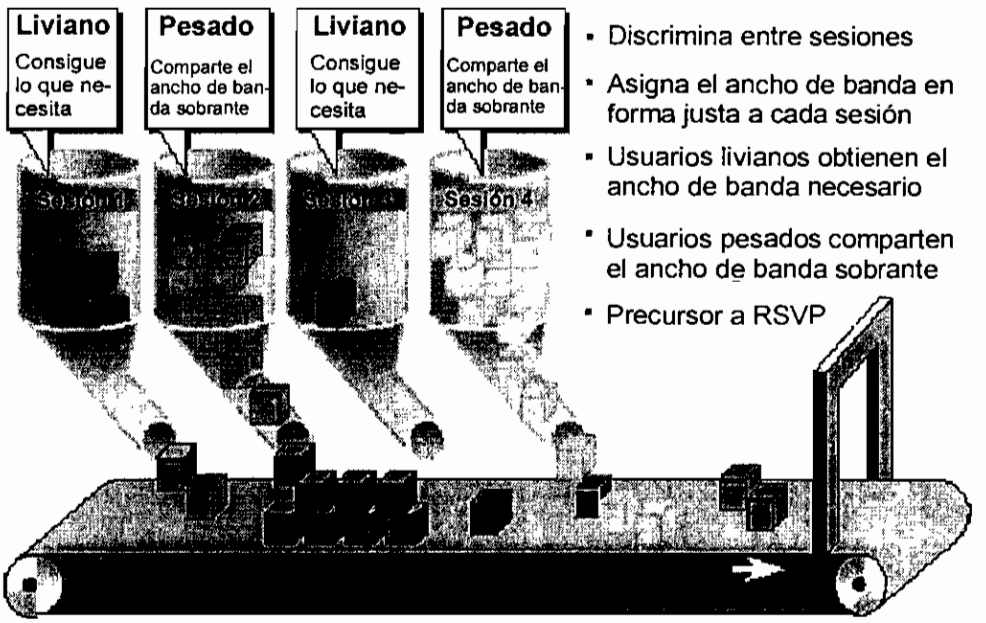

Figura 3.3 Característica de Protección del algoritmo WFQ. [26]

3.3.2.3 WFQ bajo el modelo GPS (*Generalized Processor Sharing*)

Muchos mecanismos de organización (*schedulers*) de limitada prioridad basan su diseño en la disciplina GPS, basándose únicamente en el retardo y en las propiedades de justicia del *scheduler*, GPS es una disciplina de programación ideal. Está definido con respecto a un modelo de flujo de tráfico en el cual se considera que todos los paquetes son infinitamente divisibles. GPS sirve a cada sesión activa (sesión que tiene uno o más paquetes almacenados en un *router* durante algún intervalo de tiempo) a todo instante con una velocidad mínima, cuyo

[1] Firewall.- Sistema diseñado para prevenir el acceso no autorizado a o desde una red privada; en general para intrusos desde el Internet.

valor es por lo menos igual a una velocidad previamente reservada, mientras que el exceso de ancho de banda disponible de sesiones que no están activas es distribuido entre todas las sesiones pendientes en proporción a sus reservaciones individuales. La operación de GPS puede ser explicada como sigue:

En primer lugar se asume que un conjunto de *N* sesiones { *1,2,3,..., N*} comparten un enlace de capacidad *r*. La porción de ancho de banda reservada por cada sesión *i* es representada por el número real ϕ_i. Por lo que el valor de ancho de banda deseado por cada sesión representado por ρ_i, está dado por [27]:

$$\rho_i = \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} \qquad (3.7)$$

Esto es, si μ_i es el ancho de banda reservado por la sesión *i*, de acuerdo a la definición de WFQ, se tendrá:

$$\mu_i \leq \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} \qquad (3.8)$$

Si $N_{ac}(\tau, t)$ representa el conjunto de sesiones que se encuentran pendientes de ser servidas en el intervalo (τ, t). Entonces, bajo GPS, el servicio $W_i(\tau, t)$ ofrecido a cada sesión *i* que forme parte de $N_{ac}(\tau, t)$ es proporcional a ϕ_i.

$$W_i(\tau,t) \geq \frac{\phi_i}{\sum_{j \in N_{ac}(\tau,t)} \phi_j} r(t-\tau) \qquad (3.9)$$

Por lo tanto, el mínimo servicio que una sesión *i* puede recibir en todo intervalo de tiempo es:

$$W_{i_{min}}(\tau,t) = \mu_i \times r(t-\tau) \qquad (3.10)$$

Así GPS resulta en un perfecto aislamiento, una ideal repartición de recursos, y bajos retardos en las sesiones que circulen juntas por un enlace.

En el algoritmo WFQ, un modelo fluido GPS es simulado en paralelo con el actual sistema basado en paquetes, para identificar en primer lugar el conjunto de sesiones que están pendientes a cada instante de tiempo y sus velocidades de servicio. Sobre la base de esta información, se calcula un *timestamp*[1] para cada paquete que arriba, y los paquetes son introducidos dentro de una prioridad de encolamiento de acuerdo al valor de sus *timestamps*; a fin de calcular el valor de éstos, una función virtual de tiempo *v(t)* es definida y mantenida por el *scheduler.* La función *v(t)* es una función lineal en tiempo real *t,* y su pendiente cambia dependiendo del número de sesiones activas y de las velocidades de servicio de cada una de ellas. De una forma mas precisa, si $N_{ac}(\tau, t)$ representa el conjunto de sesiones que esperan ser servidas por el *scheduler* durante el intervalo *(τ, t),* la pendiente *m* de la función virtual de tiempo durante el intervalo *(τ, t)* está dada por:

$$m = \frac{\sum_{i=1}^{N} \phi_i}{\sum_{i \in N_{ac}(\tau,t)} \phi_i} \qquad (3.11)$$

El comportamiento de una función de tiempo virtual en WFQ, es ilustrado en la figura 3.4.

En el ejemplo, se tiene tres sesiones compartiendo el ancho de banda del enlace. Se asume ciertas condiciones iniciales, como por ejemplo que la sesión 1 ha reservado un 50% del ancho de banda del enlace de salida, en tanto que las sesiones 2 y 3 han reservado cada una un 25 %, y que el sistema estuvo desocupado antes del tiempo t = 0. Por facilidad, se asume que todos los paquetes son del mismo tamaño y que el enlace maneja una velocidad de servicio de 100.000 paquetes/seg. Como puede observarse en el gráfico, el primer paquete de la sesión 1 arriba al tiempo t = 0 y comienza a ser servido inmediatamente en el fluido GPS a la velocidad máxima del enlace, ya que es la única sesión activa en ese instante de tiempo.

[1] Timestamp.- Marcación del tiempo en los paquetes, su formato es analizado en el capítulo 8 (8.5.1.4.2).

Al tiempo t = 2 μs, la sesión 2 ha tenido su primer arribo. En este punto el servidor GPS ajusta sus velocidades de servicio instantáneamente tal que la sesión 1 reciba una fracción de la capacidad total del enlace dada por:

$$\rho_1 = \frac{0.5}{(0.5+0.25)} = 0.667 \qquad (3.12)$$

esto es 0,667 x 10^5 paquetes/seg; mientras la sesión 2 recibe, una fracción del enlace total dada por:

$$\rho_2 = \frac{0.25}{(0.5+0.25)} = 0.333 \qquad (3.13)$$

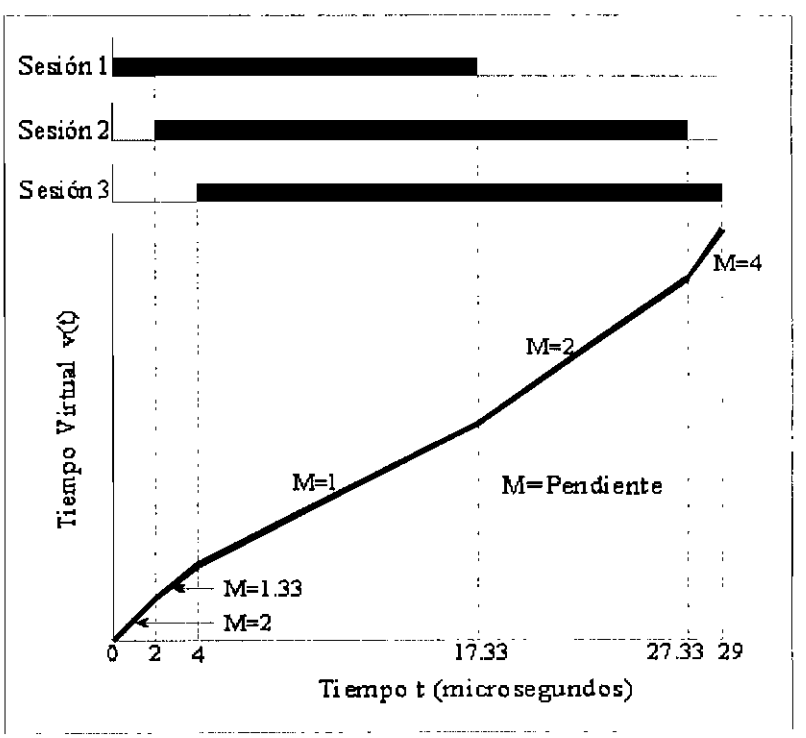

Figura 3.4 Ejemplo de implementación WFQ en base a GPS y función virtual de tiempo [27]

Al tiempo t = 4 μs, el primer paquete de la sesión 3 arriba al sistema, causando por lo tanto un cambio en las velocidades de servicio de las sesiones. Ya que todas las sesiones están activas a ese tiempo, la velocidad de servicio de cada una de ellas llega a ser igual a la velocidad reservada.

En el intervalo de 0 a 2 μs, la pendiente de *v(t)* es 1/0.5 =2, esto significa que la sesión 1 está recibiendo una velocidad de servicio igual al doble de su velocidad reservada. En el intervalo de 2 a 4 μs, la pendiente cambia a 1/(0.5+.25) = 1.33 lo cual indica que las sesiones 1 y 2 (activas en ese intervalo de tiempo) están recibiendo una velocidad de servicio 1.33 veces superior a su velocidad reservada como mínima. Finalmente, cuando la sesión 3 se activa a t = 4 μs, la pendiente cambia a 1.

La transmisión de paquetes de la sesión 1 finaliza a t = 17.33 μs, por lo que en ese instante de velocidad de servicio de las sesiones 2 y 3 se incrementa a: 1/(0.25+0.25) = 2 x velocidad reservada por cada una de ellas, y la pendiente de *v(t)* llega a ser también 2. La transmisión de los paquetes de la sesión 2 termina a t = 27.33 μs. A ese tiempo, la sesión 3 es la única sesión activa y la pendiente de la función virtual de tiempo cambia a 1/.25 = 4, lo que significa que la sesión 3 está recibiendo una velocidad de servicio cuatro veces superior a su velocidad reservada. Las tres sesiones pasan al estado de inactividad a t = 29 μs y por tanto el servidor GPS queda libre.

3.3.3 ENCOLAMIENTO BASADO EN CLASES CBQ (*Class Based Queuing*)

El Encolamiento Basado en Clases (CBQ) es un mecanismo de administración de recursos que puede ser utilizado en las arquitecturas de Servicios Diferenciados e Integrados ya que puede proveer:

- Aislamiento de tráfico "por entidad", con flexibilidad en la definición de la entidad.
- Grados de libertad para proveer un amplio rango de políticas basadas en los servicios o en los protocolos y direcciones de red.
- Compartición del enlace.

3.3.3.1 Funcionamiento

El Encolamiento Basado en Clases es un mecanismo de organización de tráfico que provee la compartición del enlace físico entre varios usuarios o clientes. CBQ

mejora el rendimiento del enlace pues permite compartir el ancho de banda asignado a un usuario si no está completamente utilizado.

CBQ puede compartir el enlace jerárquicamente mediante lo cual cada cliente asigna el ancho de banda apropiado a los diferentes tipos de tráfico, en este caso el ancho de banda no utilizado se comparte primero entre los demás tipos de tráfico del mismo cliente en lugar de compartirlo con otros. Una idealización de este mecanismo se presenta en la figura 3.5.

La operación de CBQ se basa en la interacción entre un organizador general y un organizador de compartición del enlace. El organizador general garantiza el servicio adecuado a cada clase distribuyendo el ancho de banda asignado a ella; el organizador de compartición del enlace distribuye el exceso de ancho de banda de acuerdo a la estructura fijada.

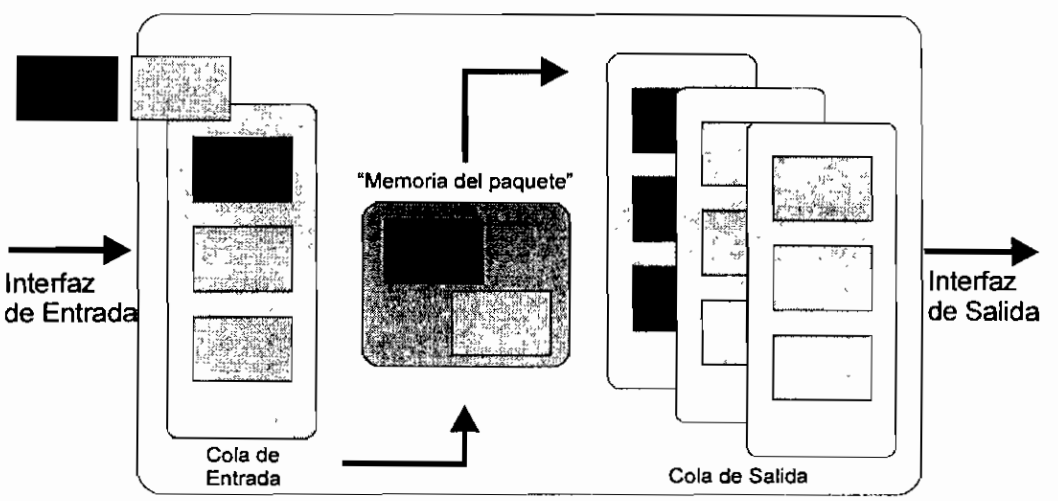

Figura 3.5 Representación del mecanismo CBQ [24]

El organizador general puede utilizar técnicas PRR (*Packet Round Robin*) o WRR (*Weighted Round Robin*). En PRR se selecciona los paquetes de cada tipo de tráfico en una forma secuencial y cíclica, sirviendo un paquete de cada tipo de tráfico en cada ciclo; WRR también selecciona los paquetes en una forma secuencial y cíclica pero se diferencia de PRR en que puede servir en cada ciclo varios paquetes de un determinado tipo de tráfico dependiendo del "peso" que se le haya dado.

El organizador de compartición del enlace es más complejo pues debe tomar en cuenta el ancho de banda que utiliza cada clase de tráfico en un determinado instante. Este organizador estima el ancho de banda utilizado por una clase de tráfico y lo marca para diferenciar si el uso de ancho de banda está bajo el límite, en el límite o sobre el límite. Para estimar el ancho de banda, el organizador chequea el estado de cada clase en cada jerarquía involucrando un gran *overhead*; debido a lo cual se permite aproximaciones de este modelo.

Las características específicas de CBQ dependen de la implementación. A continuación se considera una implementación de CBQ para el sistema operativo FreeBSD[1] conocida como Encolamiento Alternado ALTQ (*ALTernate Queuing*).

La implementación ALTQ/CBQ asume como predeterminado que cada clase de tráfico tiene una asignación fija de ancho de banda y que si no la ocupa totalmente, el ancho de banda en exceso se desperdiciará (pues no será utilizada). Para permitir compartición del enlace se debe configurar explícitamente en cada clase la opción "pedir prestado" que le permite aprovechar los recursos desocupados.

El organizador general WRR establece la asignación de ancho de banda calculando el número de bytes que puede enviar cada clase de tráfico en cada vuelta o ciclo (calculado basándose en el tamaño promedio del paquete), se permite a una clase enviar más de un paquete en cada vuelta si así está configurado. Una clase deja de transmitir paquetes si se termina su ranura de asignación o si ha sido marcada como clase "sobre el límite".

El organizador de compartición del enlace permite a una clase "pedir prestado" ancho de banda hasta un nivel de jerarquía *N*. Si la clase progenitora está sobre el límite, se permite a una clase obtener ancho de banda de sus ancestros hasta un nivel *N*. El valor de *N* se determina de la siguiente forma: [28]

- Si arriba un paquete perteneciente a una clase que no está sobre el límite, *N* toma el valor de la profundidad de la clase (nivel de anidamiento).

[1] FreeBSD.- Versión libre y popular de UNIX que puede ser descargada del Internet.

- Si *N* es *i*, y el paquete que arriba pertenece a una clase que está sobre el límite, la cual tiene un ancestro en un nivel menor que *i* (nivel *j*) que está bajo el límite, *N* toma el valor de *j*.

Si al momento de organizar un paquete no existen clases que estén bajo el límite dentro del nivel *N*, se incrementa el valor de *N* en 1 y se intenta organizar nuevamente.

- Si no se puede enviar un paquete, *N* toma el máximo valor permitido por el sistema (32), maximizando la opción de enviar un paquete en la siguiente vuelta.

- En general, una clase puede "pedir prestado" solo si su progenitora está bajo el límite o si tiene un ancestro que está bajo el límite.

3.3.3.2 Rendimiento

CBQ es un mecanismo de administración de tráfico bastante eficiente que permite diferenciar clases de servicio y entregar los recursos adecuados a las mismas. Sin embargo en la referencia [28] se demuestran ciertas falencias:

- Puede introducir un elevado retardo a algunas clases debido al uso de WRR.

- Presenta problemas cuando se reciben ráfagas de datos pues para sus cálculos utiliza el tamaño promedio de paquete.

- No se puede realizar un minucioso control de velocidad.

3.4 ALGORITMOS DE DESCARTE DE PAQUETES

3.4.1 ALGORITMO DE DETECCIÓN ALEATORIA TEMPRANA (RED, *Random Early Discard*)

El usual mecanismo de descarte de paquetes en Internet denominado *tail drop* descarta todos los paquetes que arriban al dispositivo de red luego que el espacio en la cola se ha saturado; el descarte de paquetes sirve de indicativo de

congestión a protocolos de capa transporte como TCP obligándoles a disminuir la ventana de envío de paquetes.

Esta forma de control de congestión presenta el inconveniente de la sincronización global producida por la disminución del *throughput* de todas las conexiones que atraviesan un nodo cuando éste ha llegado a congestionarse. La sincronización global es la subutilización del enlace luego que se ha producido congestión en un instante dado, debido a la presencia de ráfagas de tráfico en la red, ello provoca que todos los flujos disminuyan su ventana de transmisión.

Para contrarrestar este problema se han desarrollado nuevos algoritmos de notificación de congestión y descarte de paquetes. El algoritmo de detección aleatoria temprana (RED) es un poderoso mecanismo que junto con los algoritmos de encolamiento habilitan una administración activa de colas.

3.4.1.1 Funcionamiento

La principal característica del algoritmo RED es efectuar control de congestión descartando paquetes antes de que el espacio en cola se haya saturado. El control de congestión puede realizarse con o sin la ayuda del protocolo TCP; aunque RED por sí solo está diseñado para hacer control de congestión en el dispositivo de red, efectuar control de congestión con ayuda del protocolo de capa transporte TCP ofrece ciertas ventajas como la escalabilidad.

RED evita congestión controlando el tamaño promedio de la cola de salida, mediante el descarte de todos los paquetes de la cola si ésta supera un umbral máximo y permitiendo el envío de todos los paquetes si la cola es inferior a un umbral mínimo; si la cola tiene un tamaño promedio entre los dos umbrales, se descarta los paquetes aleatoriamente, teniendo mayor probabilidad de descarte aquellos que pertenecen a la conexión con mayor *throughput*.

Sea *avg* el tamaño promedio de la cola, min_{th} el umbral mínimo, max_{th} el umbral máximo y max_p la probabilidad de marcado cuando se alcance el umbral máximo, la probabilidad de marcado p_b está dada por la siguiente ecuación [29]:

$$p_b = \max_p \frac{avg - \min_{th}}{\max_{th} - \min_{th}} \qquad (3.14)$$

La ecuación 3.14 indica que conforme *avg* varía de min_{th} a max_{th}, la probabilidad de marcación de un paquete p_b varía linealmente de 0 a max_p. Para hacerla dependiente del tiempo, RED define la probabilidad final de marcado p_a (ecuación 3.15) la cual se incrementa ligeramente conforme transcurre el tiempo desde que el último paquete fue marcado; para la definición de p_a es necesario introducir un contador *count* que se inicializa con la marcación del último paquete.

$$p_a = \frac{p_b}{1 - count.p_b} \qquad (3.15)$$

El cálculo del tamaño promedio de la cola *avg* es crítico pues de su valor depende el efecto que produzca la sincronización global, dicho en otras palabras el incremento transitorio del tamaño instantáneo de la cola *q* no debe aumentar significativamente el tamaño promedio de ésta. Es notorio que se debe asignar un "peso" al tamaño promedio anterior y otro al tamaño instantáneo, para ello RED utiliza la variable "peso" w_q como se muestra en la ecuación 3.16.

$$avg_t = (1 - w_q).avg_{t-1} + w_q.q \qquad (3.16)$$

Como se aprecia, el tamaño que tome w_q no debe ser demasiado grande tal que impida filtrar los transitorios de congestión, ni muy pequeño pues *avg* respondería demasiado lento a los cambios en tamaño de la cola imposibilitando la detección de los estados iniciales de congestión. Los valores permitidos para w_q están desarrollados en el ANEXO 3.

3.4.1.2 Rendimiento

El algoritmo RED presenta las siguientes características:

- Evita congestión mediante el descarte de paquetes o notificando ésta por medio de un bit en la cabecera de los paquetes. Para obtener un control

$$p_b = \max_p \frac{avg - \min_{th}}{\max_{th} - \min_{th}} \qquad (3.14)$$

La ecuación 3.14 indica que conforme *avg* varía de min_{th} a max_{th}, la probabilidad de marcación de un paquete p_b varía linealmente de 0 a max_p. Para hacerla dependiente del tiempo, RED define la probabilidad final de marcado p_a (ecuación 3.15) la cual se incrementa ligeramente conforme transcurre el tiempo desde que el último paquete fue marcado; para la definición de p_a es necesario introducir un contador *count* que se inicializa con la marcación del último paquete.

$$p_a = \frac{p_b}{1 - count.p_b} \qquad (3.15)$$

El cálculo del tamaño promedio de la cola *avg* es crítico pues de su valor depende el efecto que produzca la sincronización global, dicho en otras palabras el incremento transitorio del tamaño instantáneo de la cola *q* no debe aumentar significativamente el tamaño promedio de ésta. Es notorio que se debe asignar un "peso" al tamaño promedio anterior y otro al tamaño instantáneo, para ello RED utiliza la variable "peso" w_q como se muestra en la ecuación 3.16.

$$avg_t = (1 - w_q).avg_{t-1} + w_q.q \qquad (3.16)$$

Como se aprecia, el tamaño que tome w_q no debe ser demasiado grande tal que impida filtrar los transitorios de congestión, ni muy pequeño pues *avg* respondería demasiado lento a los cambios en tamaño de la cola imposibilitando la detección de los estados iniciales de congestión. Los valores permitidos para w_q están desarrollados en el ANEXO 3.

3.4.1.2 Rendimiento

El algoritmo RED presenta las siguientes características:

- Evita congestión mediante el descarte de paquetes o notificando ésta por medio de un bit en la cabecera de los paquetes. Para obtener un control

eficiente del tamaño promedio de cola se debe configurar correctamente el valor del "peso" w_q.

- Apropiadas escalas de tiempo; el dispositivo de red debe esperar al menos un RTT (*Round Trip Time*) luego de notificar congestión para observar disminución de tráfico, además no disminuyen las ventanas de transmisión debido a transitorios de congestión.

- Evita la sincronización global; la probabilidad de marcado de los paquetes aumenta a medida que aumenta la congestión pero se evita la sincronización global marcando los paquetes a una tasa tan baja como sea posible.

- Maximiza la relación global de *throughput* a retardo; el control explícito del tamaño de cola permite mantener colas despejadas disminuyendo el retardo y aumentando el ancho de banda.

- Justicia, RED no discrimina una conexión en particular pues la fracción de paquetes marcados es aproximadamente proporcional al ancho de banda utilizado por cada una, sin embargo no asegura una fracción igual de ancho de banda para todas las conexiones, ni previene el abuso de usuarios que tienen un mal comportamiento.

- Apropiado para un gran rango de entornos; RED funciona tanto con el protocolo TCP como con otros protocolos de capa transporte, si se descarta un paquete de datos TCP se obliga a la retransmisión del mismo, pero si descarta un paquete ACK o un paquete no TCP es posible que la fuente ni siquiera se dé cuenta de lo sucedido, sin embargo siempre existe control de congestión pues RED descarta todos los paquetes (TCP y no TCP) si el tamaño promedio de cola excede el máximo valor permitido.

3.4.2 MÉTODO DE NOTIFICACIÓN EXPLÍCITA DE CONGESTIÓN (ECN, *Explicit Congestion Notification*)

Los algoritmos utilizados para evitar y controlar la congestión en TCP se basan en la noción o idea de que la red es una caja negra. El estado de congestión de la red es determinada por los sistemas finales, que prueban el estado de la red mediante el incremento de la ventana de carga (aumentando el tamaño de la ventana de transmisión de paquetes) hasta que la red tienda a congestionarse y los paquetes comiencen a perderse. El tratar la red como una caja negra y el tener como una única indicación de congestión la pérdida es apropiado para datos cursados de acuerdo al servicio *best effort*, los cuales son poco o nada sensitivos al retardo o a la pérdida de paquetes individuales, los problemas empiezan a notarse con aplicaciones que no toleran esos inconvenientes.

Debido a que TCP determina la ventana de congestión apropiada, incrementando gradualmente el tamaño de la ventana hasta que los paquetes empiecen a perderse, provoca que las colas lleguen a saturarse formando los conocidos "cuellos de botella" en los *routers.* Utilizando en el *router* alguna política de descarte de paquetes que no sea sensitiva a la carga colocada por cada flujo individual, se descartará paquetes que constituyan algún flujo sensitivo a la latencia. Los mecanismos de administración activa de colas detectan la congestión antes de que la cola llegue a desbordarse y proveen una indicación de ésta a los nodos finales.

Se han desarrollado en la actualidad varios métodos utilizados por los mecanismos de administración activa de colas para indicar congestión a los nodos finales. Uno de ellos es usar el descarte de paquetes y otro, es la Notificación Explicita de Congestión (ECN). La administración activa de colas permite al *router* separar políticas de encolamiento o descarte de paquetes de las políticas para indicar congestión. Así, se permite al *router* usar el bit de Congestión Experimentada (CE, *Congestion Experienced*) en una cabecera del paquete como una indicación de congestión, en lugar de confiar solamente en el descarte de paquetes.

3.4.2.1 Características del Método de Notificación Explícita de Congestión.

El método de Notificación Explícita de Congestión (ECN) cumple ciertos criterios y características, algunos de ellos se analizan a continuación:

- Nuevos mecanismos para controlar y evitar la congestión necesitan coexistir y cooperar con mecanismos existentes, en particular con los métodos de TCP y con prácticas de descarte de paquetes en los *routers*.

- Debido a que ECN sea probablemente adoptada gradualmente, un aspecto fundamental será acomodar la migración. Algunos *routers* podrían únicamente descartar paquetes para indicar congestión, y algunos sistemas finales podrían no ser capaces de soportar ECN. La estrategia más viable es posibilitar un desarrollo gradual sin tener que diferenciar entre regiones ECN y no ECN.

- ECN debe soportar enrutamiento asimétrico, técnica que está ganando espacio en el Internet. El camino (secuencia de enlaces y *routers*) seguido por los paquetes de datos podría ser diferente del camino seguido por los paquetes de acuse de recibo en la dirección contraria.

- Muchos *routers* procesan más eficientemente los campos regulares de la cabecera IP antes que los campos opcionales de ésta, por lo cual ECN mantiene la información de congestión en la cabecera regular de un paquete IP.

- Debe ser reconocido que no todos los sistemas finales cooperarán en los mecanismos para control de congestión, por lo tanto, los nuevos mecanismos tienen que ser fácilmente deshabilitados por aplicaciones TCP que no deseen utilizar tal mecanismo de control de congestión.

3.4.2.2 Cabecera de Notificación Explícita de Congestión en IP

Es deseable que Internet provea una indicación de congestión cuando se tenga problemas de congestión incipiente, en donde la notificación puede ser a través

del marcaje de paquetes en lugar de descartarlos. Para cumplir con este propósito se ha implementado un campo ECN de dos bits en la cabecera IP:

- Bit de transporte ECN capaz (ECT, *ECN-Capable Transport*), el cual será configurado por el origen de datos para indicar que los puntos finales del protocolo de transporte son capaces de soportar ECN.

- Bit de Congestión Experimentada (CE), el cual será colocado por el *router* para indicar congestión a los nodos finales.

Estos bits corresponden a los bits 6 y 7 del octeto TOS[1] en IPv4 (campo ECN); el bit 6 es designado como el bit ECT y el bit 7 es designado como el bit CE. El octeto TOS IPv4 corresponde al octeto de Clase de Tráfico en IPv6.

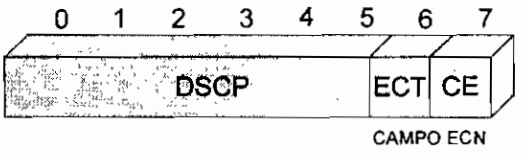

Figura 3.6 Campo ECN dentro de la cabecera IPv4. [30]

Cuando un sistema final recibe la notificación de congestión explícita, las acciones de los algoritmos de control de congestión deben ser esencialmente las mismas que cuando se ha descartado un solo paquete. La razón para requerir esto es acomodar el desarrollo gradual de ECN tanto en los sistemas finales como en los *routers*. Algunas acciones que podrían tomar los *routers* son:

- Descartar los paquetes ECN, por ejemplo, usando las mismas políticas del algoritmo RED para detección de congestión.

- Colocar el bit CE para niveles equivalentes de congestión.

- Descartar un paquete no ECN, y colocar el bit CE en un paquete ECN para niveles equivalentes de congestión.

[1] TOS.- El octeto Tipo de Servicio se detalla en el siguiente capítulo (4.4).

Diferentes respuestas al control de congestión a una indicación CE y al descarte de paquetes resultaría en tratamiento no equitativo para flujos diferentes, por lo que se recomienda configurar adecuadamente la región.

Un requerimiento adicional es que los sistemas finales deberían reaccionar a la congestión máximo una vez en un intervalo de tiempo correspondiente a un RTT para evitar varias reacciones a múltiples indicaciones de congestión dentro de un tiempo de ida y vuelta.

En un *router*, el bit CE de un paquete ECN debe ser únicamente colocado si el *router* hubiera descartado el paquete como una indicación de congestión a los nodos finales. Cuando el *buffer* del *router* todavía no se ha saturado y el *router* está preparado para descartar un paquete como una información a los nodos finales de congestión incipiente, el *router* debe primero chequear si el bit ECT está configurado en la cabecera del paquete IP, si es así, entonces en lugar de descartar el paquete, el *router* podría colocar el bit CE en la cabecera IP.

Un ambiente donde todos los nodos fueren capaces de soportar ECN permitiría el desarrollo de nuevos criterios para configurar el bit CE y nuevos mecanismos de control de congestión para la recepción de paquetes CE en los nodos finales.

Cabe mencionar que mientras ECN está inseparablemente atado con la administración activa de colas en el *router*, la reciprocidad no ocurre; los mecanismos de administración activa de colas han sido desarrollados y desplegados independientemente de ECN, usando descarte de paquetes en su ausencia dentro de la arquitectura IP.

3.4.2.3 ECN en TCP

Además de la funcionalidad dada por el campo ECN dentro de la cabecera del paquete IP, ECN requiere soporte del protocolo de transporte.

En primer lugar el protocolo de transporte requerirá negociar entre los puntos extremos durante el establecimiento de la sesión para determinar si la totalidad de nodos finales en el camino de datos son capaces de soportar ECN, con el objeto de permitir al origen de datos configurar el bit ECT en los paquetes transmitidos.

Segundo, el protocolo de transporte debe ser capaz de reaccionar apropiadamente a la recepción de paquetes CE. Esta reacción podría ser en la forma que el receptor de datos informe al origen de la recepción de los paquetes CE (por ejemplo con TCP) o de alguna otra acción que reduzca la velocidad de arribo de ese flujo al receptor.

En TCP, la implementación de ECN requiere tres nuevos mecanismos:

1. Negociar entre los puntos finales durante el establecimiento de la sesión para determinar si los extremos son capaces de soportar ECN.

2. Implementar una bandera *ECN-Echo* en la cabecera TCP para que el destino pueda informar al origen de datos cuando un paquete CE ha sido recibido.

3. Y finalmente, implementar una bandera de Ventana de Congestión Reducida (CWR, *Congestion Window Reduced*) en la cabecera TCP para que el origen pueda informar al receptor que la ventana de congestión ha sido reducida.

Los mecanismos TCP para negociar las capacidades ECN usan la bandera *ECN-Echo* definida en el bit 9 del campo reservado de la cabecera TCP. Para comunicar al receptor TCP cuando dejar de colocar la bandera *ECN-Echo*, se utiliza la bandera de Ventana de Congestión Reducida (CWR) definida en el bit 8 del campo reservado en la cabecera TCP.

En la fase de establecimiento de la conexión TCP, el origen y el destino intercambian información acerca de sus capacidades o deseos de usar ECN. Una vez establecida esta negociación, el origen TCP coloca el bit ECT en la cabecera IP de los paquetes de datos para indicar a la red que el protocolo de transporte es capaz y participará en una ambiente ECN para esos paquetes. Esto indicará a los *routers* que pueden marcar estos paquetes con el bit CE, si es que ellos usan este método como una notificación de congestión. Si la conexión TCP no desea usar notificación ECN para un paquete en particular, el origen TCP coloca el bit ECT

igual a cero (es decir no lo activa), y el receptor TCP ignora el bit CE en la paquete recibido.

El establecimiento de una sesión TCP que soporta ECN involucra las siguientes acciones:

- El nodo transmisor envía un paquete de solicitud de conexión (SYN) colocando las banderas ECN-Echo y CWR en la cabecera TCP. Para un paquete SYN, las banderas ECN-Echo y CWR son definidas como una indicación de que el envío TCP soporta ECN, en lugar de actuar como una indicación o una respuesta de congestión.

- El nodo receptor del paquete anterior responde con el envío de un paquete SYN-ACK, activando la bandera ECN-Echo y desactivando la bandera CWR en la cabecera TCP, lo cual indica que éste soporta o participa en ECN.

La razón por la cual se tiene que en el camino de envío TCP activa dos banderas para el paquete SYN mientras que en el acuse de recibo, en los paquetes SYN-ACK, únicamente activa una sola bandera es que la asimetría así conseguida provee una robusta negociación de capacidades ECN en implementaciones TCP.

Luego del establecimiento de una conexión TCP que soporta ECN, la transmisión de los paquetes de datos se efectúa de la siguiente manera:

- El origen transmite los paquetes con el bit ECT activado (colocado en 1) en la cabecera IP.

- Si se detecta congestión en la ruta entre el origen y el destino, el nodo congestionado activará el bit CE en la cabecera IP de los paquetes.

- Cuando el nodo destino recibe un paquete CE, activa la bandera ECN-Echo en la cabecera TCP del paquete de acuse de recibo subsiguiente. Si hay alguna acción de retardo implementada en los paquetes ACK, como por ejemplo la implementación TCP "*delayed-ACK*" en la cual el receptor TCP puede enviar un acuse de recibo por cada dos paquetes de datos que

arriban, entonces la bandera ECN-Echo en el paquete ACK será activada en el acuse de recibo subsiguiente a la llegada del primer paquete CE.

- Si el origen recibe un paquete de acuse con la bandera ECN-Echo activada conoce que la red se ha congestionado sobre el camino desde el origen hasta el destino. La indicación de congestión debería ser tratada como una pérdida de paquetes en un ambiente que no soporta ECN, es decir, la fuente TCP reduce a la mitad la ventana de congestión y reduce el punto de partida del umbral. Para proveer seguridad en contra de la posibilidad del descarte de un paquete ACK ECN-Echo, el receptor TCP debe activar la bandera ECN-Echo en una serie de paquetes ACK.

- Cuando el origen reduce la ventana de congestión, activa la bandera CWR en el primer paquete de datos enviado luego de la reducción de la ventana. Si este paquete de datos es descartado en la red, entonces el origen TCP tendrá que reducir la ventana de congestión nuevamente y retransmitir el paquete descartado, por lo que el mensaje de Ventana de Congestión Reducida debe ser entregado confiablemente al destino de datos.

- Cuando el destino recibe un paquete CWR debe terminar la marcación de la bandera ECN-Echo en los paquetes de asentimiento, hasta la recepción de un nuevo paquete CE.

Una condición crítica es que TCP no reaccione a una indicación de congestión más de una vez por ventana de datos (RTT). Esto es, la ventana de congestión del origen TCP debería ser reducida únicamente en respuesta a una serie de paquetes descartados CE de una sola ventana de datos.

Para la actual generación de algoritmos de control de congestión TCP, los paquetes de acuse de recibo puros (es decir, los que no contienen algunos datos especiales) deben ser enviados sin activar el bit ECT, pues los receptores TCP no tienen mecanismos para reducir el tráfico en el camino de los paquetes ACK en respuesta a la notificación de congestión.

Mecanismos que respondan a la congestión de paquetes ACK son actualmente motivos de estudio. Por ejemplo una posibilidad podría ser que el origen reduzca su ventana de congestión cuando reciba un paquete ACK con el bit CE activado. Para actuales implementaciones, un simple descarte de paquetes ACK tiene un muy pequeño efecto sobre la velocidad de envío de TCP.

3.4.2.4 Regiones no colaboradoras de ECN

Esta sección discute aspectos concernientes a la vulnerabilidad de ECN en nodos finales que no colaboran en un ambiente ECN, es decir nodos que activado el bit ECT en los paquetes transmitidos no responden a la recepción de paquetes CE. La adición de ECN a la arquitectura IP no incrementará significativamente su actual vulnerabilidad a flujos insensibles.

Aún para ambientes no capaces de soportar ECN, hay serias inquietudes acerca del peligro que puede ser causado por flujos no colaboradores o insensibles (esto es, flujos que no responden a la indicaciones del control de congestión reduciendo su velocidad de envío al enlace congestionado). Por ejemplo, un nodo final podría en un caso dado desactivar su control de congestión para no reducir su ventana de congestión en respuesta a paquetes descartados. Por lo que se hace necesario que los *routers* utilicen mecanismos, tales como WFQ, para detectar y diferenciar el tratamiento de los paquetes de flujos no colaboradores.

Ha sido argumentado que el descarte de paquetes por si mismo podría ser un adecuado impedimento para no-colaboración, y que el uso de ECN quita este impedimento. La verdad es que, los *routers* capaces de soportar ECN preservan el comportamiento de descartar paquetes en tiempos de alta congestión; y aún en tiempos de alta congestión, el descarte de paquetes por si solo no es un adecuado impedimento para no colaborar. [30]

Inicialmente, los *routers* ECN únicamente marcan los paquetes (en lugar de descartarlos) si el nivel de congestión, y por tanto la tasa de marcaje del paquete, sea razonablemente bajo. Durante períodos donde el promedio de tamaño de la cola excede un umbral superior, y por lo tanto la velocidad de marcaje de los

paquetes sería alta, los *routers* descartarían paquetes en lugar de colocar el bit CE.

Durante los períodos de baja o moderada velocidad de marcaje de paquetes, los flujos no sensibles al retardo usando un protocolo de entrega confiable tendrían un incentivo para incrementar, en lugar de disminuir, su velocidad de envío en presencia de paquetes marcados. Algunos métodos han sido propuestos para identificar y restringir flujos no colaboradores o no sensibles. La adición de ECN a la red en ninguna forma incrementará la dificultad de diseñar y desarrollar tales mecanismos. [30]

El fracaso de control de congestión efectiva podría ser causado no únicamente por un nodo no-colaborador sino también por la pérdida de la indicación de congestión en la red misma. Esto podría suceder debido a un *router* averiado coloca el bit CE en un paquete de un transporte que no soporta ECN, o borra el bit CE en paquetes que arriban. Como un ejemplo, un *router* averiado que borra el bit CE impediría que la indicación de congestión alcance al receptor, lo cual provocará el fracaso del control de congestión para aquel flujo e incrementará la congestión en la red.

La acción de esta clase de *routers* podría también resultar en una innecesaria indicación de congestión a los nodos finales, ésta acción puede provocar descarte de paquetes en ausencia de congestión. Desde un punto de vista del control de congestión, si un *router* no-colaborador activa el bit CE en ausencia de congestión, sería equivalente a que un *router* descarte innecesariamente un paquete.

3.4.2.5 Justificación del bit ECT

La necesidad de la implementación del bit ECT radica en el hecho de que ECN será desplegado gradualmente en un Internet donde algunos protocolos de transporte y *routers* no entienden ECN. Con el bit ECT, el *router* puede descartar paquetes de flujos que no soportan ECN, pero puede en cambio activar el bit CE en flujos que si lo soporten.

Si no existiese la indicación dada por ECT, el *router* tendria que activar el bit CE en paquetes de flujos capaces y no capaces de soportar ECN, lo cual no incentivaría a los nodos finales a desplegar ECN y no podría establecerse un camino viable para el desarrollo gradual hacia un mundo ECN. Al inicio de la congestión, cuando la tasa de marcaje/descarte de paquetes es baja, los *routers* únicamente marcan el bit CE y no descartan los paquetes. Por lo tanto, solamente aquellos flujos que soportan ECN serán capaces de entender y responder a los paquetes CE. De no existir un claro entendimiento, el resultado se reflejará en que los flujos capaces ECN serían retirados, y los flujos no capaces de soportar ECN ignorarán las señales ECN y continuarán ampliando su ventana de congestión.

Así, subconjuntos de flujos capaces de soportar ECN, deben tener mecanismos para indicar este hecho a los *routers*, pues de lo contrario el control de congestión en el Internet sería menos efectivo. Como uno de estos mecanismos se usa el bit ECT en la cabecera IP, y de allí la justificación de su implementación.

3.4.2.6 Implementación ECN Alternativa

Además de la implementación ECN en base a dos bits (ECT y CE), existe una implementación alternativa que utiliza un solo bit con dos valores: el primer valor, "ECT o no CE" representaría un Transporte ECN capaz o la no existencia de congestión, y el otro valor "CE o no ECT" representaría congestión experimentada o un transporte no ECN.

Existen dos diferencias básicas entre las implementaciones ECN de dos y un bit:

- El tratamiento que recibe un paquete que atraviesa múltiples *routers* congestionados. Para entender de mejor manera esta diferencia, considérese un paquete CE que arriba a un segundo nodo congestionado, y es seleccionado por el administrador activo de colas del *router* para marcaje o descarte: en la implementación de un solo bit, el segundo *router* congestionado descarta el paquete CE, debido a que no puede distinguir entre un paquete CE y un paquete no ECT; en la implementación de dos bits, el segundo *router* congestionado tiene la posibilidad de seleccionar el paquete a descartar o de dejarlo solamente con el bit CE activado.

- En la implementación de un solo bit, el origen de los datos ECN tendría ambigüedad para indicar al receptor si cada paquete ha sido configurado como ECN capaz o como no ECN. Una posibilidad para el origen sería indicar en la cabecera de transporte si el paquete fue configurado como ECN capaz. Una segunda posibilidad, que involucraría una limitada funcionalidad para la implementación en un solo bit, sería que el origen indique sin ambigüedades que enviará todos sus paquetes como capaces o no capaces de soportar ECN.

En resumen, aún cuando la implementación en un solo bit es posible, tiene ciertas limitaciones respecto a la de dos bits. La implementación en un bit tiene una funcionalidad mucho más limitada para el tratamiento de paquetes CE en el segundo *router* congestionado, hecho muy común en Internet, y por lo tanto es más propensa a errores.

Capítulo 4

ARQUITECTURA DE SERVICIOS DIFERENCIADOS

CAPÍTULO 4

ARQUITECTURA DE SERVICIOS DIFERENCIADOS

En la actual evolución del Internet, se hace necesaria la clasificación del tráfico a fin de dar el tratamiento adecuado a cada flujo de datos circulante por la red. Sobre la base de esta idea se implantó la Arquitectura de Servicios Diferenciados (*DiffServ*) en Internet cuyo objetivo es jerarquizar el tráfico y otorgarle de acuerdo a su clasificación cierto tipo de servicio.

Un "servicio" define alguna característica significativa de la transmisión unidireccional de paquetes a través de uno o más caminos dentro de una red. Estas características podrían ser especificadas en términos cuantitativos o estadísticos de *throughput*, retardo, *jitter*, y/o pérdida; o podrían ser especificadas en términos de alguna prioridad relativa de acceso a los recursos de red. La diferenciación del servicio es útil para cumplir con requerimientos de aplicaciones y usuarios, además para diferenciar el precio del servicio Internet. [32]

En una manera sencilla y concisa, se puede definir a Servicios Diferenciados como un conjunto de tecnologías que permiten al proveedor de servicio de red ofrecer diferentes clases de Calidad de Servicio a diferentes usuarios y a sus flujos de datos.

La filosofía de *DiffServ* es que un pequeño número de comportamientos distintos de red son suficientes para soportar todo el rango de posibles aplicaciones, mientras los nodos en el centro de la red manejen los paquetes en diferentes flujos de tráfico enviándolos de acuerdo al comportamiento por salto (PHB, *Per Hop Behaviors*) asignado a cada uno de ellos. El PHB a ser aplicado es indicado por el valor del *codepoint* DS (DSCP, *DiffServ CodePoint*) presente en la cabecera de todo paquete IP. [33]

La construcción de una arquitectura en base a este esquema, tiene la ventaja que muchos flujos de tráfico pueden ser agregados a un mismo grupo de paquetes para ser marcados con igual DSCP, los cuales son enviados usando el mismo PHB, simplificando de esta manera el procesamiento y el almacenaje asociados. Por lo tanto, no es necesaria mas señalización que la dada por el DSCP de cada paquete, y no se requiere otro procesamiento en el centro de la red *DiffServ* pues QoS es implementada paquete por paquete.

La Arquitectura *DiffServ* requiere de un cierto grupo de elementos en los nodos, que serán analizados en detalle en el desarrollo del presente capítulo, tales como:

- Definición de un pequeño conjunto de comportamientos por salto (PHBs).
- Funciones de clasificación de paquetes.
- Funciones de acondicionamiento de tráfico, incluyendo medida, marcaje, configuración y administración.

4.1 ARQUITECTURA DiffServ

4.1.1 CARACTERÍSTICAS DE LA ARQUITECTURA

La Arquitectura *DiffServ*, define y cumple los siguientes requerimientos y características:

- Acomoda una amplia variedad de servicios y políticas de aprovisionamiento, extendiéndolas de principio a fin o en una región de la red.
- Permite independizar el servicio de la aplicación particular en uso.
- Presenta una amplia escalabilidad. Esto se logra implementando funciones complejas de clasificación y acondicionamiento de tráfico únicamente en los nodos extremos de la red, mientras en la parte central implementa comportamientos por salto.

- Independiza las funciones de acondicionamiento de tráfico y aprovisionamiento de servicio de los comportamientos por salto implementados en los nodos centrales de la red.

- Requiere únicamente un pequeño grupo de PHBs, cuya implementación no domina el costo de un dispositivo de red, y los cuales no introducen cuellos de botella para implementaciones futuras de sistemas de alta velocidad.

- Permite la implementación de una simple clasificación de paquetes en los nodos centrales de red a través de los clasificadores.

4.1.2 *DiffServ* vs OTROS MODELOS

Existen en la actualidad algunos modelos, que en algún caso podrían confundir o considerarse una especie de *DiffServ*, motivo por el cual a continuación se amplía las diferencias entre ellos. Entre estos "modelos alternativos" se encuentran:

- Marcaje de prioridad relativa

- Marcaje de servicio

- Conmutación de etiquetas

- Servicios Integrados (RSVP)

Ejemplos del **modelo de marcaje de prioridad** relativa incluyen el marcaje de precedencia en IPv4, la prioridad Token Ring en 802.5 y la interpretación por definición de clases de tráfico en 802.1p [32]. En este modelo la aplicación, el host o el nodo proxy[1] selecciona una prioridad relativa o precedencia para un paquete (por ejemplo, retardo o prioridad de descarte), y los nodos de red a lo largo del camino aplican la apropiada prioridad de comportamiento de envío de acuerdo al valor de precedencia especificada en la cabecera del paquete.

DiffServ puede ser considerado como un refinamiento a este modelo, permitiendo especificar de mejor manera el rol e importancia de los nodos extremos y de los

[1] Proxy.- Servidor que se sitúa entre una aplicación del cliente y un servidor real permitiendo la interacción entre ambos.

acondicionadores de tráfico, además el modelo de comportamiento por salto permite crear más y mejores reglas de comportamientos de envío que las dadas únicamente por el retardo o prioridad de descarte.

En el **modelo de marcaje de servicio**, cada paquete es marcado con una solicitud de "tipo de servicio", la cual podría incluir retardo mínimo, máximo *throughput*, máxima confiabilidad o mínimo costo. Los nodos de red seleccionarían caminos o comportamientos de envío que se hallen apropiadamente construidos para satisfacer el servicio requerido. Este modelo no describe el uso del campo DSCP, en lugar de ello, se refiere al uso de los bits D, T y R del campo TOS (observar la figura 4.9), el cual es muy genérico y no abarca el rango de posibles servicios

En el **modelo de conmutación de etiquetas** (o circuitos virtuales) se incluyen Frame Relay, ATM y MPLS. En éste, el estado del camino de envío y el estado de QoS es establecido por flujos de tráfico a lo largo del camino de red.

A cada conjunto de tráfico con ciertas características se le asigna una etiqueta en el nodo de ingreso, la cual determina el correspondiente PHB y la etiqueta reemplazante en cada salto. Este modelo permite una mayor facilidad de administración de los recursos, pues las etiquetas tienen significado local y afectarán solo al dominio administrativo, sin embargo, no se afecta la asignación de recursos extremo a extremo configurada cuando se establece el circuito virtual.

Una desventaja de este modelo es el costo, pues se necesita requerimientos adicionales de administración y configuración para el establecimiento y mantenimiento de los caminos de etiquetas conmutadas. Otra desventaja es el *overhead* introducido para mantener el estado de envío en cada nodo.

El **modelo de Servicio Integrados** permite a las fuentes y receptores intercambiar mensajes de señalización, a lo largo del camino entre ellos, para determinar la forma de proceder ante un flujo determinado. Normalmente, este modelo requiere añadir información en todos los nodos para mantener el estado de la reservación de cada flujo, introduciendo un *overhead* significativo que lo haría sumamente inadecuado en redes capaces de manejar millones de flujos a la

vez. Este modelo también requiere soporte del protocolo de señalización RSVP (*Resource Reservation Protocol*).

Una vez que se ha contrastado los modelos alternativos con el de *DiffServ*, es el momento de hablar sobre el modelo de la Arquitectura de Servicios Diferenciados.

4.2 MODELO DE LA ARQUITECTURA DE SERVICIOS DIFERENCIADOS

La arquitectura *DiffServ* está basada en un modelo relativamente simple en el cual el tráfico entrante a una red es clasificado, y posiblemente acondicionado, en un grupo con comportamientos específicos (BAs, *Behavior Aggregates*[1]). Cada BA es identificado por un solo valor DSCP, el cual define el comportamiento por salto (PHB).

En esta sección se da los conceptos y características de los componentes más importantes de un sistema *DiffServ*, tales como: dominio y región de servicios diferenciados, funciones de clasificación de tráfico y de acondicionamiento de paquetes, además se presenta un ejemplo sobre la manera en la cual los Servicios Diferenciados son alcanzados mediante la combinación de funciones de acondicionamiento de tráfico y PHBs.

4.2.1 DOMINIO DE SERVICIOS DIFERENCIADOS

Un dominio *DiffServ* es un conjunto de nodos DS[2] los cuales operan con políticas comunes de aprovisionamiento de servicio y un mismo grupo de PHBs implementados en cada uno de ellos.

Un dominio DiffServ consiste de nodos DS frontera y nodos DS internos. Los nodos DS frontera interconectan el dominio DS con otro dominio sea este DS o no DS, en tanto que los nodos DS internos únicamente se conectan a un nodo DS interno o a un nodo DS frontera, dentro del mismo dominio.

---

[1] Behavior Aggregate.- es un conjunto de paquetes que han sido marcados con el mismo valor DSCP, y por tanto seleccionan un mismo PHB.

[2] Nodo DS.- es un nodo capaz de soportar funciones y comportamientos de Servicios Diferenciados, el término es usualmente usado en referencia a un nodo o dispositivo de un dominio *DiffServ*.

Los nodos DS frontera son los encargados de clasificar y marcar el tráfico que ingresa al dominio con el objetivo de seleccionar el PHB adecuado. Los nodos DS internos seleccionan el comportamiento de envío para los paquetes de acuerdo al DSCP marcado por el nodo frontera. La presencia de nodos que no soporten los Servicios Diferenciados dentro de un dominio DiffServ produciría funcionamientos impredecibles impidiendo el cumplimiento de Acuerdos de Nivel de Servicio[1] (SLAs), sin embargo puede existir este tipo de implementaciones.

Un dominio DS normalmente consiste de una o más redes bajo la misma administración; por ejemplo, la Intranet de una organización o un ISP. El administrador del dominio es responsable de asegurar que los recursos sean aprovisionados o reservados adecuadamente para cumplir los SLAs acordados.

Tanto los nodos frontera como los internos, deben tener la capacidad de aplicar los PHBs apropiados a los paquetes basándose en el DSCP. En forma adicional, los nodos DS frontera requieren ejecutar funciones de acondicionamiento de tráfico entre el dominio al cual pertenecen y su parte par del dominio al cual se conectan; en cambio los nodos internos deben tener la habilidad para ejecutar limitadas funciones de acondicionamiento de tráfico tales como la remarcación del DSCP, razón por la cual la implementación de nodos frontera es mucho más compleja que la de nodos internos.

Un host dentro de un dominio *DiffServ* podría actuar como un nodo DS frontera para el tráfico de aplicaciones que corren sobre él, si el host no actúa como un nodo frontera, entonces el nodo topológicamente más cercano actúa como nodo DS frontera para el tráfico del mencionado host.

[1] SLA.- Acuerdos de nivel de Servicio; una definición más detallada se encuentra en el Capítulo 8 (8.4)

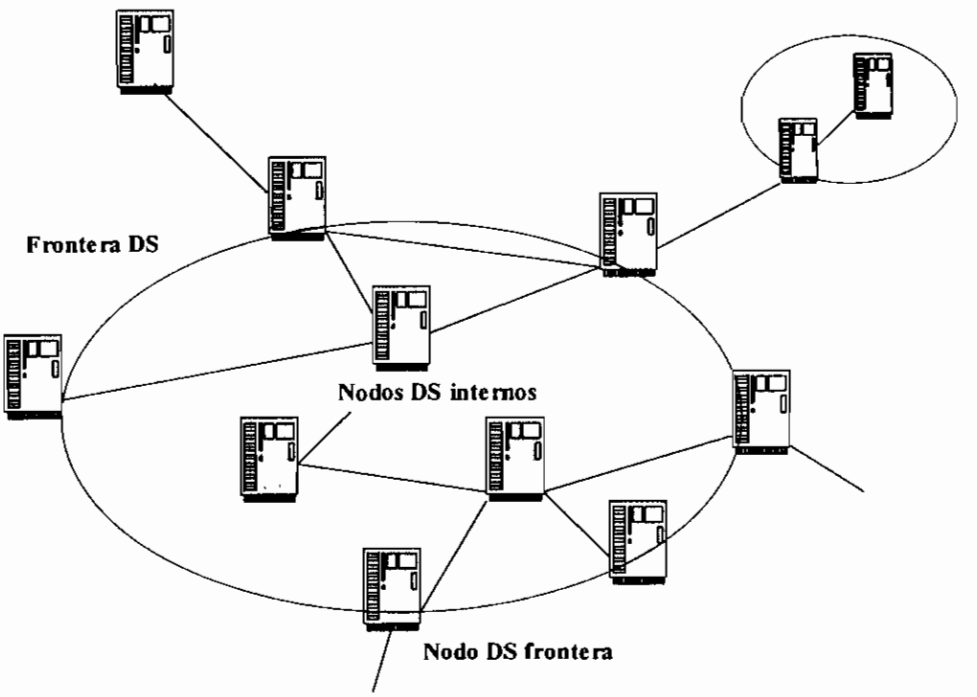

Figura 4.1 Diagrama de un Dominio DiffServ. [34]

4.2.2 REGIÓN DE SERVICIOS DIFERENCIADOS

Una región *DiffServ* es un conjunto de uno o más dominios *DiffServ* contiguos, y por tanto capaz de soportar Servicios Diferenciados.

Los dominios DS en una región podrían soportar internamente diferentes grupos de PHBs, razón por la cual es importante establecer, entre los dominios, un TCA (*Traffic Conditioning Agreement*) en el cual se especifique cómo el tránsito de un dominio DS se acondiciona a otro dominio.

Es muy posible que algunos dominios DS dentro de una región, posean políticas de aprovisionamiento de servicio comunes y soporten iguales grupos de PHBs y DSCPs, eliminando la necesidad de acondicionar el tráfico entre esos dominios.

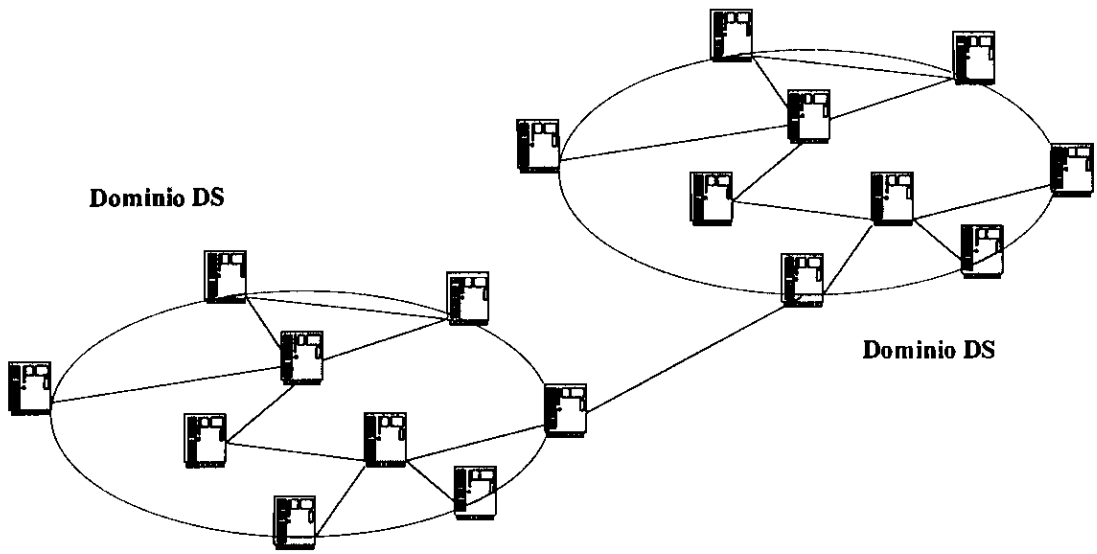

Figura 4.2 Diagrama de una Región DiffServ. [34]

4.2.3 CLASIFICACIÓN Y ACONDICIONAMIENTO DEL TRÁFICO

Los SLAs especifican perfiles de tráfico y acciones a llevarse a cabo con tráfico que se halle dentro o fuera de este perfil, pero además pueden indicar la forma de remarcar el tráfico dentro de una región DiffServ.

Mediante políticas de clasificación de paquetes se identifica el subconjunto de tráfico que debe recibir un servicio diferenciado, acondicionándolo a uno o más BAs dentro del dominio.

El acondicionamiento de tráfico se lo realiza mediante acciones de medida, configuración, administración y remarcación para asegurar que el tráfico entrante al dominio DS se adapte a las reglas especificadas en el TCA en concordancia con las políticas del dominio. Para la realización de estas acciones se requiere implementar en los *routers* ciertas funcionalidades que en la actualidad no existen, razón por la cual se presenta a continuación el modelo de un *Router DiffServ* propuesto por el IETF.

4.3 MODELO CONCEPTUAL DE UN *ROUTER DiffServ*

Este modelo conceptual de un *router DiffServ* define los elementos funcionales del camino de datos, sus posibles parámetros de configuración y la manera en la que ellos pueden ser interconectados a fin de alcanzar las funciones de clasificación, acondicionamiento de tráfico y de funcionalidad de PHBs. El modelo que se presentará más adelante comprende tanto los *routers* DS frontera como los internos.

Un *router* bajo este modelo incluye elementos que cumplen las siguientes funciones:

- Clasificación de tráfico.
- Funciones de medición del tráfico.
- Acciones de Acondicionamiento de Tráfico tales como marcación, descarte, enumeración y multiplexación.
- Funciones de encolamiento.

Los diversos elementos y las funciones arriba mencionadas suelen combinarse para formar bloques que son manejados por herramientas de administración y configuración *DiffServ*. Se puede representar los elementos y funciones de un *router* DiffServ mediante el diagrama de bloques que se presenta en la figura 4.3.

En la parte central del gráfico se puede notar la presencia de una interfaz de entrada, una de salida y el bloque de enrutamiento central. Aún cuando en los *routers* existen una serie de interfaces de entrada y salida, el gráfico trata únicamente de ejemplarizar la estructura de un nodo DiffServ, sin embargo el modelo es completamente válido cuando se tiene más de una interfaz.

Los componentes de interés dentro de cada interfaz son los clasificadores, los componentes de acondicionamiento y los componentes de encolamiento que soportan el tráfico *DiffServ* y hacen cumplir los distintos PHBs. Los mencionados

componentes son importantes y fundamentales dentro del modelo, por lo que más adelante se los analizarán en detalle.

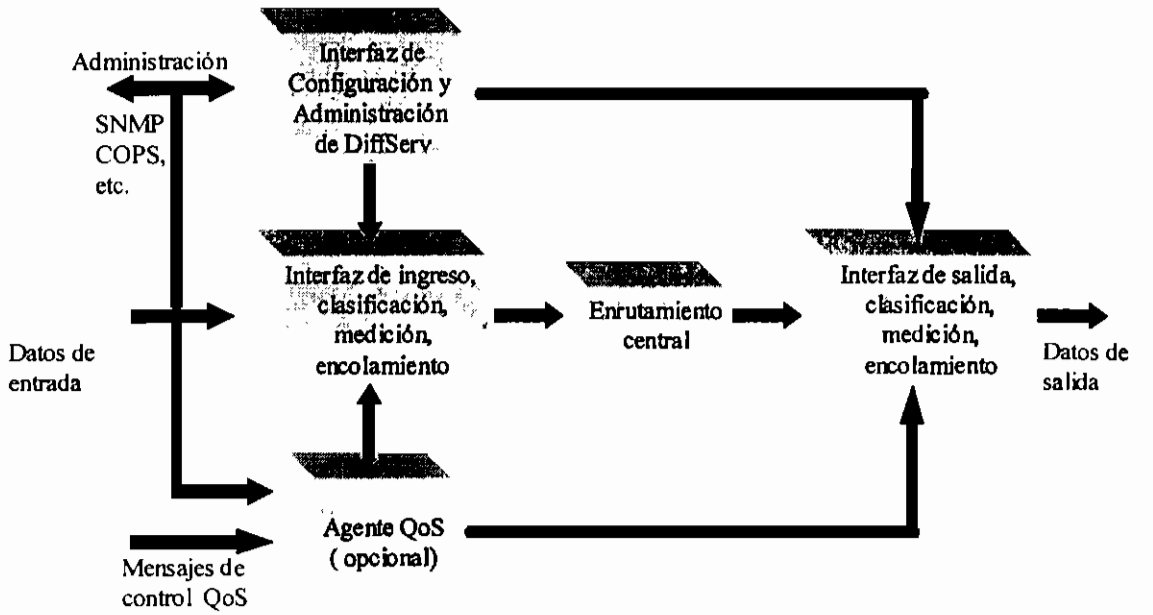

Figura 4.3 Bloque Funcional de un *Router DiffServ.* [33]

La interfaz de configuración y administración es la encargada de proveer y monitorear los parámetros de operación *DiffServ* elaborando estadísticas que le permitan estimar el estado del tráfico cursado. El administrador de la red interactúa con esta interfaz a través de uno o más protocolos de administración tales como SNMP, COPS o de herramientas de configuración de *routers* como terminales seriales o consolas telnet.

El módulo de Agente QoS es necesario cuando existen implementaciones de *DiffServ* junto con Servicios Integrados (Arquitecturas Híbridas), tal como se verá en el Capítulo 6, con el objeto de dar soporte a las reservaciones de estado que tendrá que hacer RSVP. Como puede observarse en la figura 4.3, este módulo actúa únicamente en el plano de control y no en el de datos, es decir en el camino de datos continuarán haciéndose las "reservas" mediante básicamente el valor del DSCP de cada paquete IP.

Esta forma de representar esquemáticamente un *router DiffServ*, permite construir la siguiente jerarquía de componentes :

1. El nivel superior, en el cual el administrador maneja las interfaces. Cada interfaz consiste de un componente de entrada y uno de salida. Cada componente puede contener elementos de clasificación, acción, medida y encolamiento.

2. En el siguiente nivel, el administrador de red maneja grupos de elementos funcionales organizándolos en Bloques de Acondicionamiento de Tráfico (TCBs, *Traffic Conditioning Block*s), los cuales son usados para implementar alguna política de red.

3. En el tercer y más bajo nivel se encuentran los elementos funcionales en forma individual, cada uno de ellos con sus propios parámetros de configuración.

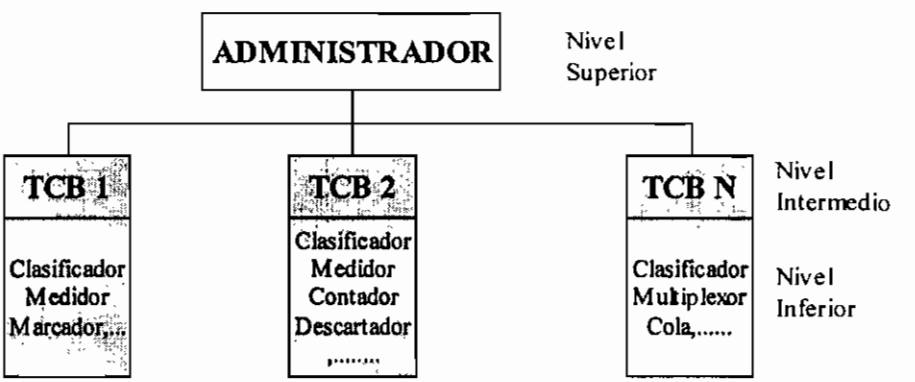

Figura 4.4 Jerarquía de Componentes dentro de un *Router DiffServ.* [33]

4.3.1 CLASIFICADORES

Son dispositivos 1 : N (1 entrada : N salidas), toman como entrada un flujo de tráfico y, basándose en el contenido de alguna porción de la cabecera de los paquetes constituyentes de ese flujo, generan como salida N flujos de tráfico separados lógicamente.

La clasificación de los paquetes se la hace mediante la utilización de filtros, los cuales examinan el contenido de uno o varios atributos de clasificación asociados con los paquetes.

En la actualidad se han definido dos tipos de clasificadores:

- El Clasificador BA (*Behavior Aggregate*), el cual clasifica los paquetes basándose únicamente en el valor del DSCP.
- El Multiclasificador MF (*Multi-Field*), el cual clasifica los paquetes basándose en el valor de la combinación de uno o más campos de la cabecera, tales como dirección de origen, dirección de destino, DSCP, protocolo ID, número de puerto de origen y destino.

Figura 4.5 Diagrama de un clasificador. [34]

A fin de utilizar un clasificador de una manera más eficiente, es posible implementar un multiplexor antes de él, permitiendo de esta forma la entrada de múltiples flujos de tráfico. Por ejemplo, si varias subinterfaces de ingreso se alimentan a través de un solo clasificador entonces el número de interfaz puede ser considerado por el clasificador como un atributo de clasificación asociado al paquete.

El clasificador de la figura 4.5 opera en base a tres filtros, y separa un tráfico entrante en tres flujos de salida. El Filtro1 y el Filtro2 analizan el valor del DSCP y lo comparan con valores estandarizados, a fin de clasificar correctamente el flujo entrante, en tanto que el Filtro 3 actúa como una especie de filtro predeterminado, es decir filtrará todos aquellos paquetes que no sean asociados ni al Filtro 1 ni al Filtro 2.

Filtro	DSCP
Filtro1	101010
Filtro2	111111
Filtro3	****** (comodín)

Un filtro está constituido por un conjunto de condiciones aplicadas a cierta(s) parte(s) de un paquete IP que sean útiles para clasificar. Así por ejemplo en el clasificador BA de la figura 4.5, la condición de clasificación esta dada por el campo DSCP; en tanto que en un Multiclasificador, la clasificación se la realiza analizando varios campos de la cabecera de los paquetes, por ejemplo de la siguiente manera:

Filtro	Dir. IP Origen	Dir. IP Destino	Puerto TCP Origen	Puerto TCP Destino
Filtro4	172.31.10.2/32	172.31.5.X	X	5003

En este ejemplo de un clasificador MF, se analizan cuatro valores todos ellos presentes en la cabecera de los paquetes; el campo de dirección IP del origen, el campo de dirección IP del destino (sin importar el cuarto octeto de la dirección), el campo del puerto origen TCP y finalmente el campo del destino TCP.

A fin de llevar a cabo su labor, los clasificadores deben ser configurados con algunos procedimientos de administración en concordancia con el apropiado TCA. De igual manera, la información que usan los clasificadores para seleccionar los paquetes debe ser previamente autentificada.

4.3.1.1 Clasificador BA

El clasificador *DiffServ* más simple es el clasificador BA, el cual usa únicamente el valor del campo DSCP de la cabecera del paquete IP para determinar la salida lógica a la cual el paquete debe ser direccionado. Un posible filtro BA se encuentra definido como:

Filtro5	
Tipo :	BA
Valor:	111000

4.3.1.2 Multiclasificador BA

Este clasifica los paquetes basándose en uno o más campos del paquete (incluyendo muy posiblemente el valor del DSCP), un tipo muy común de clasificador MF es un clasificador séxtuplo el cual selecciona los paquetes de acuerdo a seis campos de las cabeceras IP, TCP o UDP:

- Dirección IP de origen
- Dirección IP de destino
- Protocolo (especificado en la cabecera IP)
- Puerto (TCP / UDP) origen
- Puerto (TCP / UDP) destino
- DSCP

Otros clasificadores MF podrían clasificar paquetes en base a otros campos tales como dirección MAC (*Medium Access Control*), etc. Un filtro MF séxtuplo puede ser definido como sigue:

Filtro 6:	
Tipo :	IPv4 séxtuplo
IPv4DestAddrValue:	0.0.0.0
IPv4DestAddrMask:	0.0.0.0
IPv4SrcAddrValue:	172.31.8.0
IPv4SrcAddrMask:	255.255.255.0
IPv4DSCP:	28
IPv4Protocol:	4
IPv4DestL4PortMin:	0
IPv4DestL4PortMax:	65535
IPv4SrcL4PortMin:	20
IPv4SrcL4PortMax:	20

4.3.2 MEDIDORES

Los medidores al igual que los clasificadores son dispositivos lógicos 1: N (1 entrada : N salidas), y éstos miden las propiedades temporales de un flujo de

paquetes seleccionado por un clasificador comparándolas con algún perfil de tráfico especificado en el TCA. Los proveedores de redes *DiffServ* ofrecen diferentes servicios a sus clientes de acuerdo al perfil temporal del tráfico que el cliente entrega a la red. Un medidor pasa la información de sus medidas a otras funciones de acondicionamiento, a fin de que éstas lleven a cabo acciones particulares para cada paquete dependiendo si éstos se encuentran dentro o fuera del perfil.

Los medidores son caracterizados por un perfil temporal y por niveles de conformidad, cada uno de los cuales están asociados con una salida del medidor, mismas que pueden ser conectadas a otro elemento funcional del *router*.

La figura 4.6 ilustra un medidor con tres niveles de conformación de tráfico:

Figura 4.6 Diagrama de un medidor genérico. [32]

Un medidor bajo este modelo, mide la velocidad a la cual los paquetes que forman un flujo de tráfico pasan a través de él, y la compara con algún conjunto de velocidades umbrales emitiendo uno o más resultados de esta comparación.

4.3.2.1 Perfiles de tráfico

Un perfil de tráfico especifica las propiedades temporales de un flujo de tráfico seleccionado por un clasificador. Éste provee reglas para determinar si un paquete en particular está dentro del perfil o fuera de él. Por ejemplo, un perfil basado en un *token bucket*[1] podría verse como sigue:

$$\text{DSCP} = X \text{, utiliza } \textit{token-bucket}\ (r,\ b)$$

[1] *Token Bucket*.- Una descripción detallada de este algoritmo se encuentra en el ANEXO 4 (sección A4.2).

El perfil arriba mencionado indica que todos los paquetes marcados con DSCP igual a *X* serían evaluados por un medidor de cubeta con fichas (*token bucket*) de velocidad *r* y tamaño límite *b*. En este ejemplo, los paquetes fuera del perfil son aquellos paquetes que arriban cuando no hay suficientes fichas (*tokens*) en la cubeta (*bucket*). El concepto de dentro y fuera de perfil puede ser extendido a más de dos niveles, pudiéndose implementar múltiples niveles de parámetros dentro de un perfil.

Diferentes acciones de acondicionamiento serán aplicadas a los paquetes que se hallen dentro y fuera del perfil. Así, a los paquetes que se hallen dentro del perfil se les permitiría entrar al dominio *DiffServ* sin acondicionamientos adicionales; o alternativamente, podrían cambiarse los valores de sus DSCP. En forma contraria, los paquetes fuera de perfil serán encolados hasta que sean encasillados dentro de algún perfil (acción de formado), descartados o marcados con un nuevo valor del DSCP (acción de remarcaje). Los paquetes fuera de perfil serán asignados a uno o más BAs cuyo rendimiento sea inferior a los BAs que son asignados a los paquetes que se hallen dentro del perfil, es decir se les da un tratamiento de menor calidad.

4.3.2.2 Ejemplos de medidores

4.3.2.2.1 Medidor de velocidad promedio

Mide la velocidad promedio a la cual los paquetes cursan la red en un promedio de tiempo específico.

Un medidor de velocidad promedio puede tener la siguiente forma:

Medidor1:	
Tipo:	VelocidadPromedio
Perfil:	Perfil1
SalidaConformante:	Cola1
SalidaNoConformante:	Contador1
Perfil1:	
Tipo:	VelocidadPromedio
VelocidadPromedio:	140 kbps
Delta[1]:	100 msec

[1] Delta.- Es la longitud del intervalo de tiempo en el cual se realiza una medida de tráfico.

Un medidor que utilice este perfil podría continuamente mantener una cuenta que indique el número total de paquetes que arriben entre el tiempo T y el tiempo T+100 ms. Siempre que un paquete que arribe no lleve a la cuenta sobre los 14 kb en al menos 100 ms, entonces el paquete se puede considerar dentro del perfil. Algún paquete que lleve a la cuenta sobre los 14 kb, será considerado fuera del perfil.

Este medidor considera únicamente dos niveles de conformación del tráfico: dentro o fuera del perfil, enviándolo luego al tratamiento subsiguiente apropiado.

4.3.2.2.2 Medidor EWMA (Exponential Weighted Moving Average)

El medidor EWMA puede ser parametrizado como sigue:

$$vel\_pro(t) = (1 - w_q) * vel\_pro(t') + w_q * vel(t) \qquad (4.1)$$

$$t = t' + Delta \qquad (4.2)$$

Para un paquete que arriba al tiempo t:

Si (vel_pro(t) > VelocidadPromedio)

FUERA DE PERFIL

Caso contrario

DENTRO DE PERFIL

vel(t) mide el número de bytes entrantes en un pequeño intervalo de muestreo previamente fijado por el valor de delta. Todo paquete que arribe y lleve la velocidad promedio sobre una velocidad predefinida (*VelocidadPromedio*) es considerado fuera de perfil. Como se observa en la ecuación 4.1, la velocidad promedio depende de los valores anteriores; esta dependencia la marca la constante peso (w_q), definida más profundamente en el capítulo 3 (sección 3.4.1.1).

El perfil de un medidor EWMA podría ser el siguiente:

Medidor2:

Tipo:	EWMA
Perfil:	Perfil2
SalidaConformante:	Cola1
SalidaNoConformante:	DescartadorAbsoluto1

Perfil2:

Tipo:	EWMA
VelocidadPromedio:	25 kbps
Delta:	10 μs.
w_q:	1/16

4.3.2.2.3 Medidor token bucket sencillo

Este tipo de medidor es más sofisticado que los analizados hasta el momento; confronta alguna característica del tráfico con un perfil *token bucket*. Dicho perfil generalmente posee dos parámetros que son: una velocidad promedio (*r*) y un tamaño límite (*b*). (Refiérase al ANEXO 4, sección A4.2).

La forma en la que un medidor de este tipo actúa es comparando la velocidad de arribo de los paquetes con una velocidad promedio (*r*) especificada en el perfil *token bucket*. Lógicamente, las fichas (*tokens*) se acumulan en una cubeta (*bucket*) a la velocidad promedio hasta alcanzar máximo el tamaño límite (*b*). Paquetes de longitud L bytes son considerados dentro del perfil si existen *tokens* disponibles en el *bucket,* al tiempo que éstos arriban; caso contrario, se los considera no conformantes o fuera del perfil.

Un medidor *token bucket* puede aparecer como sigue:

Medidor3:

Tipo:	TokenBucketSencillo
Perfil:	Perfil3
SalidaConformante:	Cola1
SalidaNoConformante:	DescartadorAbsoluto1

Perfil3:

Tipo:	TokenBucketSencillo
VelocidadPromedio:	250 kbps
TamañoDeRáfaga:	100 kbytes

Así, un medidor de este tipo compara el tráfico entrante con un Perfil 3 cuyas características son velocidad máxima 250 kbps y tamaño límite 100 kB, de esta

forma todo tráfico que esté dentro del perfil será enviado a la Cola 1 para su tratamiento posterior, en tanto que el tráfico que se halle fuera del perfil será enviada a un Descartador Absoluto 1, en donde seguramente el tráfico será descartado.

4.3.2.2.4 Medidor token bucket multi-etapa

Este medidor define dos tamaños límites y tres niveles de conformación del tráfico, por lo que su implementación suele ser más complicada que un medidor *token bucket* sencillo. Dentro de los dos tamaños límites se define un límite inferior y un límite superior. Los paquetes que exceden el tamaño límite superior son considerados fuera del perfil o no conformantes, los paquetes que exceden el límite inferior son considerados parcialmente conformantes y finalmente aquellos paquetes que no exceden el límite inferior son considerados conformantes del perfil.

Un perfil para un medidor *token bucket* multi-etapa con tres niveles de conformación se define como sigue:

Medidor4:

Tipo:	MultiTokenBucket
PerfilA:	Perfil4
SalidaConformanteA:	Cola1
PerfilB:	Perfil5
SalidaConformanteB:	Marcador1
SalidaNoConformante:	DescartadorAbsoluto1

Perfil4:

Tipo:	TokenBucket
VelocidadPromedio:	100 kbps
TamañoDeRáfaga:	20 kB

Perfil5:

Tipo:	TokenBucketSencillo
VelocidadPromedio:	100 kbps
TamañoDeRáfaga:	100 kB

En el ejemplo de implementación del medidor se tiene dos tamaños límites de ráfagas, el inferior es de 20 kB y el superior de 100 kB. De acuerdo a esto todo paquete que se encuentre dentro del perfil A será considerado "conformante" y será enviado a la Cola1 para su tratamiento posterior. Si en cambio, el paquete ha

sido seleccionado dentro del perfil B significa que el límite inferior ha sido superado pero el superior aún no, por tanto los paquetes son considerados "parcialmente conformantes" y enviados a un marcador a fin de indicar esta característica a los elementos funcionales posteriores al medidor. Finalmente si los paquetes han superado el tamaño límite superior, serán descartados completamente, pues son considerados "no-conformantes" del perfil.

Como puede notarse, un medidor de este tipo suele ser implementado mediante una configuración de múltiples medidores *token bucket* sencillos.

4.3.3 ELEMENTOS ACTIVOS

Una vez que se ha clasificado el tráfico, y luego se han medido sus características se deben llevar a cabo ciertas acciones tales como:

- Marcaje
- Descarte Absoluto
- Multiplexación
- Conteo
- Ninguna acción

A fin de conseguir la puesta en marcha de estas acciones se definen los elementos activos, tales como marcadores, descartadores, multiplexores, contadores y otros que a continuación se los detalla.

4.3.3.1 Marcadores

Éstos, son elementos 1:1 (1 entrada : 1 salida), los cuales configuran el valor del DSCP en la cabecera de los paquetes IP.

Los marcadores de paquetes colocan el valor DSCP en el campo *DiffServ* del paquete, adicionándolo a un BA (*Behaviour Aggregate*) particular. La marcación

se la efectúa acorde a la información del medidor. La acción de remarcación se produce cuando el marcador cambia el DSCP en un paquete.

Los marcadores DSCP para redes *DiffServ* son normalmente caracterizados por un solo parámetro: el campo DSCP de seis bits en la cabecera del paquete, como se muestra a continuación:

Marcador1:

Tipo: MarcadorDSCP
Marca: 010010

4.3.3.2 Formadores

Los formadores son los elementos activos encargados de retardar algunos o todos los paquetes de un flujo de tráfico no conformante, con el objetivo de encasillarlos en algún perfil de tráfico.

Un formador usualmente posee un *buffer* de tamaño finito. Los paquetes serán descartados cuando se haya agotado el espacio en el *buffer*.

4.3.3.3 Descartadores

Son elementos activos que descartan algunos o todos los paquetes de un flujo de tráfico no conformante, con el objetivo de conseguir que el flujo cumpla con algún perfil de tráfico. Este procedimiento se lo conoce como "administrar" el flujo.

Un descartador puede ser implementado como un caso especial de un formador, únicamente configurando el tamaño del *buffer* de éste a cero.

4.3.3.4 Bloques acondicionadores de tráfico (TCBs, *Traffic Conditioning Block*)

Los elementos funcionales hasta el momento analizados se los puede construir dentro de un Bloque Acondicionador de Tráfico, es decir, un TCB contiene los siguientes elementos: clasificador, medidor, marcador, formador y descartador.

Un flujo de tráfico es seleccionado por un clasificador, el cual dirige los paquetes a la parte lógica del acondicionador de tráfico. Dentro de éste, se utiliza un medidor para comparar el flujo de tráfico seleccionado con un perfil, el resultado de tal

comparación, es decir si el flujo se encuentra dentro o fuera del perfil, se usa para tomar acciones de marcaje, descarte o formación.

Cuando los paquetes dejan el acondicionador de tráfico del nodo DiffServ frontera, el DSCP de cada paquete debe colocarse en el valor apropiado.

La Figura 4.7 muestra un diagrama de bloques de un clasificador y un acondicionador de tráfico, partes constituyentes de un TCB.

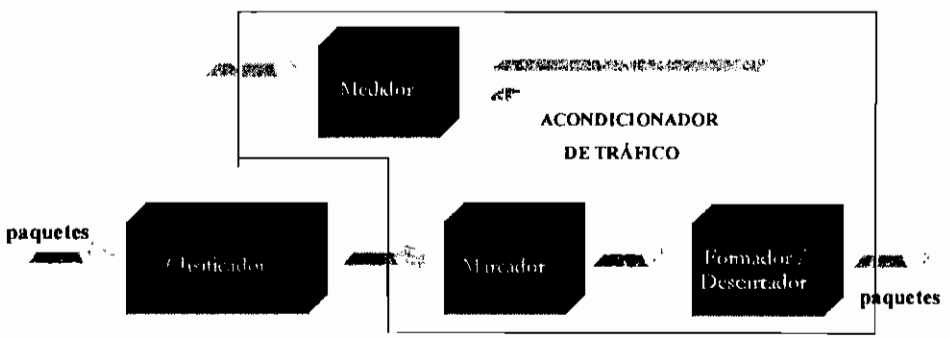

Figura 4.7 Diagrama de un Bloque Acondicionador de Tráfico (TCB). [33]

EJEMPLO DE UN TCB

Un acuerdo de nivel de servicio es un contrato establecido entre el cliente y el proveedor de servicio, mediante el cual este último se compromete a cursar el tráfico del primero bajo ciertos parámetros establecidos previamente. El acuerdo podría ser establecido de la siguiente forma[1]:

DSCP	PHB	PERFIL	TRATAMIENTO
001001	EF	Perfil1	Descartar tráfico "no conformante"
001100	AF11	Perfil2	Conformar al perfil, descartar cuando la cola se llene.
001101	AF21	Perfil3	Remarcar tráfico no-conformante con el DSCP 001000, y descartar cuando la cola se llene.
Otro	BE	ninguno	Aplicar descarte, por ejemplo bajo RED.

Bajo este acuerdo de nivel de servicio el cliente podría enviar paquetes marcados con el DSCP 001001, mismos que serán cursados a través de un PHB EF siempre que conformen el Perfil 1 y cualquier paquete "no conformante" será

[1] En el ejemplo se consideran PHBs como: EF, AF1, AF2 y BE, los cuales se los analiza en la sección 4.5.

descartado completamente. Los paquetes marcados con el DSCP igual a 001100 serán tratados a fin de que conformen el Perfil 2 luego de lo cual serán enviados bajo el tratamiento de el PHB AF11. Los paquetes cuyo DSCP sea igual a 001101 serán comparados con un Perfil 3, remarcando los paquetes que no conformen el perfil con un valor de DSCP igual a 001000. Finalmente para cualquier otro valor de DSCP, el tráfico será cursado en base al PHB BE (*best effort*).

La figura 4.8 ilustra un TCB que podría ser usado para implementar este acuerdo de nivel de servicio en la interfaz de ingreso de la frontera cliente/proveedor.

La clasificación en este ejemplo es realizada por un solo clasificador BA, el cual es usado para separar el tráfico en base al acuerdo de servicio *DiffServ* solicitado por el cliente e indicado por el campo DSCP. El clasificador se implementa con tres filtros DSCP: A, B y C, mientras el filtro X es un filtro comodín que selecciona todo paquete que no haya sido seleccionado por los tres anteriores.

Como puede observarse el camino para los paquetes marcados con los DSCP 001001 y 001101 incluye un estado de medición. Existe un medidor independiente para cada conjunto de paquetes que circulen por las salidas A y C del clasificador, basándose cada uno de ellos en un perfil específico para el correspondiente nivel de servicio *DiffServ*. En este ejemplo de TCB los medidores únicamente tienen dos niveles de discriminación de tráfico: conformante o no conformante.

A continuación del estado de medición se encuentra el estado de acción, el cual emplea contadores y marcadores. Los paquetes con DSCP 001001 que sean considerados fuera de perfil son contabilizados y descartados, mientras los paquetes que se encuentran dentro del perfil son enviados a la Cola 1. Los paquetes cuyo DSCP sea igual a 001101 y considerados no conformantes son remarcados, luego de lo cual son enviados a un multiplexor, los paquetes conformantes no se remarcan y pasan directamente al multiplexor; finalmente, tanto el tráfico conformante y no conformante es direccionado al Descartador2 y a la Cola3. Los paquetes marcados con DSCP 001100 (salida B) son enviados directamente al Descartador1 y a la Cola2.

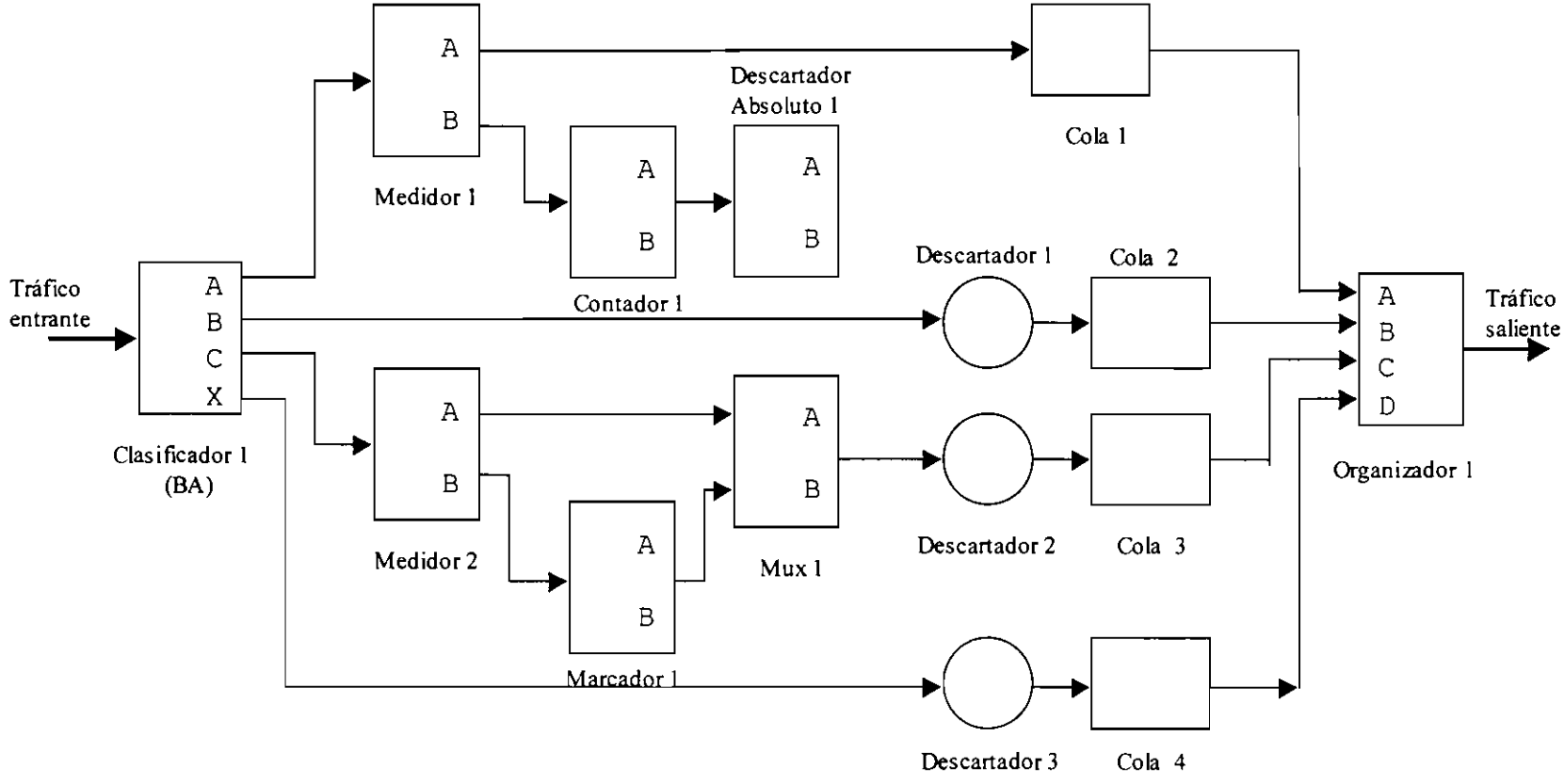

Figura 4.8 Diagrama a bloques de un TCB. [34]

El estado de encolamiento se realiza de la siguiente forma:

- Los paquetes conformantes marcados con DSCP 001001 son pasados después del Medidor1 a la Cola1, en la cual no se producirán sobre flujos que saturen la cola y obliguen a descartar paquetes.

- Los paquetes cuyos DSCP sean 001100 deben ser dirigidos directamente desde el clasificador hacia un descartador de cola, Descartador 1, el cual sirve para limitar la profundidad de la cola siguiente (Cola 2).

- En forma similar los paquetes que posean DSCP iguales a 001101 son direccionados, luego de pasar por el Multiplexor 1, al Descartador 2 y posteriormente a la Cola 3.

- Finalmente, todos los paquetes que posean otros valores de DSCP son enviados a un Descartador 3, el cual utiliza un algoritmo de descarte tipo RED basándose en la información proporcionada por la Cola 4, acerca de su profundidad y posible saturación. De esta forma el Descartador 3 probablemente descarte suficientes paquetes de su flujo de entrada para mantener bajo control el tamaño de la cola.

Esas cuatro colas son luego servidas por un algoritmo organizador (Organizador 1) el cual ha sido configurado para entregar a cada cola una prioridad y/o una compartición apropiada del ancho de banda. Las entradas A y C entregan garantías de ancho de banda, especificadas en el perfil contratado. La entrada B tiene un límite en el ancho de banda que puede utilizar. La entrada D no entrega límites o garantías de ancho de banda y posee la prioridad más baja (servicio *best effort*).

La interconexión de los elementos del TCB ilustrado en la figura 4.7 puede ser representado como sigue:

TCB1:

Clasificador1:

FiltroA:	Medidor1
FiltroB:	Descartador1
FiltroC:	Medidor2

Predeterminado: Descartador3

Medidor1:

Tipo: VelocidadPromedio
Perfil: Perfil1
SalidaConformante: Cola1
SalidaNoConformante: Contador1

Contador1:

Salida: DecartadorAbsoluto1

Medidor2:

Tipo: VelocidadPromedio
Perfil: Perfil3
SalidaConformante: Mux1. Entrada A
SalidaNoConformante: Marcador1

Marcador1:

Tipo: MarcadorDSCP
Marcador: 001000
Salida: Mux1.Entrada B

Mux1:
Salida: Descartador2

Descartador1:

Tipo: AlgoritmoDescartador
Disciplina: DescarteSobreUmbral
Disparador: Cola2-Profundidad > 10kB
Salida: Cola2

Descartador2:

Tipo: AlgoritmoDescartador
Disciplina: DescarteSobreUmbral
Disparador: Cola3-Profundidad > 20kB
Salida: Cola3

Descartador3:

Tipo: AlgoritmoDescartador
Disciplina: RED
Disparador: Interno
Salida: Cola3
MinUmbral: Cola3. Profundidad > 20 kB
MaxUmbral: Cola3. Profundidad > 40 kB

Cola1:

Tipo: FIFO
Salida: Organizador1. Entrada A

Cola2:

Tipo:	FIFO
Salida:	Organizador1. Entrada B

Cola3:

Tipo:	FIFO
Salida:	Organizador1. Entrada C

Cola4:

Tipo:	FIFO
Salida:	Organizador1. Entrada D

Organizador1:

Tipo:	Organizador 4 Entradas

Entrada A:

MaxVelPerfil:	ninguna
MinVelPerfil:	Perfil4
Prioridad:	20

Entrada B:

MaxVelPerfil:	Perfil5
MinVelPerfil:	ninguna
Prioridad:	40

Entrada C:

MaxVelPerfil:	ninguna
MinVelPerfil:	Perfil3
Prioridad:	20

Entrada D:

MaxVelPerfil:	ninguna
MinVelPerfil:	ninguna
Prioridad:	10

4.3.3.5 Localización de acondicionadores de tráfico

Los acondicionadores de tráfico se hallan comúnmente localizados en los nodos *DiffServ* frontera, tanto de entrada como de salida, pero pueden estar localizados en nodos internos del dominio *DiffServ*.

A continuación se analiza la localización de acondicionadores y clasificadores en varias partes del dominio, estableciéndose el lugar más adecuado para un mejor desempeño.

4.3.3.5.1 Dentro del dominio fuente

Se define el dominio fuente como el dominio bajo el cual se hallan contenidos el o los nodos que originan el tráfico al cual se le otorga un servicio en particular. Las fuentes de tráfico y los nodos intermedios dentro de un dominio fuente pueden ejecutar funciones de clasificación y acondicionamiento de tráfico. El tráfico originado en el dominio fuente puede ser marcado por la fuente directamente o por un nodo intermedio antes de cruzar la frontera, esta acción es conocida como marcaje inicial.

Por ejemplo, considerar una compañía cuya política es que sus paquetes LPs deben tener alta prioridad. El host LP marcará el campo DS de todos los paquetes salientes con un valor de DSCP que indique alta prioridad. Alternativamente, el *router* que se halla directamente conectado al host LP clasificará el tráfico y marcará los paquetes LPs con el valor correcto de DSCP. Es importante condicionar el tráfico de alta prioridad cerca de la fuente, a fin de establecer un límite en la cantidad de tráfico de alta prioridad enviado por una fuente en particular.

Existen algunas ventajas para marcar los paquetes cerca de la fuente de tráfico. En primer lugar, una fuente puede soportar más fácilmente las preferencias de una aplicación cuando decide cuáles paquetes deben recibir mejor tratamiento. En segundo lugar, la clasificación de los paquetes es mucho más simple antes que el tráfico sea juntado con paquetes de otras fuentes, ya que el número de reglas de clasificación dentro de un solo nodo es reducido.

4.3.3.5.2 En la frontera del dominio DS

Los flujos de tráfico pueden ser clasificados, marcados y acondicionados en cualquiera de los dos extremos de la frontera de un enlace, es decir en el nodo de salida de tráfico de un dominio o en el nodo de ingreso del tráfico del dominio subsiguiente. El Acuerdo de Nivel de Servicio entre los dominios debe especificar cuál de ellos tiene la responsabilidad de asignar los flujos de tráfico a grupos de comportamientos agregados (BAs) y acondicionar esos grupos de acuerdo con el TCA apropiado. Por lo tanto, un nodo de ingreso debe asumir que el tráfico

entrante podría no estar dentro de los parámetros exigidos por el TCA y debe estar preparado para ejecutar el TCA en concordancia con las políticas locales.

Cuando la remarcación y el acondicionamiento de los paquetes se realiza en el nodo de salida, unas pocas reglas de clasificación y acondicionamiento de tráfico necesitan ser soportadas en el dominio subsiguiente. En estas circunstancias el dominio destino del flujo podría únicamente remarcar o descartar los BAs entrantes para ejecutar el TCA.

4.3.4 ASIGNACIÓN DE RECURSOS DE RED

La implementación, configuración, operación y administración de los grupos PHBs soportados en los nodos de un Dominio *DiffServ* deben compartir efectivamente los recursos de esos nodos y de los enlaces, entre los BAs presentes, en concordancia con las políticas de aprovisionamiento de servicio. Los acondicionadores de tráfico pueden por lo tanto controlar el uso de estos recursos a través de la ejecución de TCAs y posiblemente a través de realimentación de los nodos. Aún cuando un rango de servicios pueden ser desarrollados en la ausencia de funciones de acondicionamiento de tráfico complejas (por ejemplo, usando únicamente políticas estáticas de marcación), funciones tales como administración, formado y remarcación dinámica habilitan el despliegue de mejores TCAs. [10]

4.3.5 INTEROPERABILIDAD CON NODOS NO *DiffServ*

Un nodo no *DiffServ* es aquel que no tiene la capacidad o no está configurado para interpretar el campo DS y por tanto no implementa alguno o todos los PHBs estándares. Existe una clase de nodos que implementan la clasificación de precedencia IPv4 y envío de tráfico como lo definen los RFCs 791 y 1812, pero no soportan *DiffServ*, los nodos de esta clase son conocidos como Nodos Legado (*Legacy Node*). Una diferencia clave entre un nodo Legado y un nodo no *DiffServ* es que el primero puede interpretar los bits 0, 1 y 2 del octeto TOS, mientras un nodo no DS definitivamente no hace esa interpretación.

Los Servicios Diferenciados dependen de los mecanismos de asignación de recursos provistos por los PHBs implementados en los nodos. La calidad del servicio podría bajar cuando el tráfico transite un nodo o un dominio que no es capaz de soportar DS.

Se analiza en primer lugar el caso que involucra el uso de un nodo que no soporta DS localizado en un dominio DS. Un PHB es usado para asignar recursos en el nodo y en el enlace de una manera controlada. Cuando se trabaja sobre enlaces ligeramente sobrecargados (en los cuales el retardo de los paquetes, el *jitter* y la pérdida podrían ser insignificantes) el uso de un nodo no DS no provocaría una degradación en el servicio. En circunstancias más realistas, la ausencia de un PHB en un nodo haría imposible ofrecer bajo retardo, baja pérdida, o asignar ancho de banda a través de los caminos que atraviesan el nodo. En este caso, el uso de un Nodo Legado es una alternativa aceptable, asumiendo que el dominio se restrinja a sí mismo a utilizar únicamente el DSCP Selector de Clases[1] y que la implementación particular de precedencia en el nodo Legado provea comportamientos de envío que sean compatibles con los servicios ofrecidos.

El segundo caso involucra un dominio que no es capaz de soportar *DiffServ*, es decir, no desarrolla funciones de acondicionamiento de tráfico en los nodos frontera; por lo tanto, aún en el evento que el dominio contenga nodos interiores Legados o *DiffServ*, la ausencia de tratamiento del tráfico en la frontera limitará la habilidad para entregar algunos tipos de servicios a través del dominio. Un dominio DS y un dominio no DS deben establecer un acuerdo que determine la forma de marcar el tráfico que sale de un dominio DS y que ingresa a un dominio no DS. Alternativamente, en dominios no DS que contengan nodos Legados, el dominio DS (que trata el tráfico antes) podría remarcar el tráfico de servicios diferenciados a una o más de las clases del DSCP Selector de clases. Si no existen conocimientos sobre las capacidades de administración de tráfico en el dominio siguiente, y no exista un acuerdo de por medio, un nodo de egreso de un dominio remarcará el DSCP a cero, asumiendo que el dominio no DS tratará al tráfico con servicio *best effort*.

[1] DSCP Selector de Clase.- Refiérase a la sección 4.4.

En el caso que un tráfico fluya desde un dominio no DS hacia un dominio DS, éste debe ser acondicionado en el nodo de ingreso al dominio DS, de conformidad con el Acuerdo de Nivel de Servicio apropiado.

4.4 CABECERA IP Y *CODEPOINTS DiffServ*

Históricamente el campo TOS (1 byte) presente en la cabecera de todo paquete IP, ha sido definido de la siguiente manera:

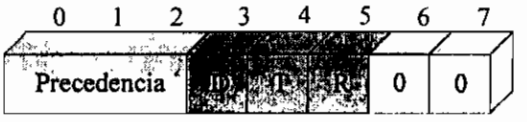

Figura 4.9 Estructura del campo TOS. [35]

Se definió el campo TOS con el objeto de indicar la calidad de servicio deseada, para un cierto tráfico. Pero no ha sido utilizado para este propósito, ya que al no haber una clasificación de tráfico y una correcta interpretación de este campo en los dispositivos de red, no se ha logrado establecer una calidad de servicio en Internet.

En la definición del Campo TOS, los bits de precedencia pueden tomar los siguientes valores:

Valor de Precedencia	Definición
111	Control de Red
110	Control Entre Redes
101	Crítico/ECP
100	Muy Urgente
011	Urgente
010	Inmediato
001	Prioridad
000	Rutina

Tabla 4.1 Valores de Precedencia definidos para el campo TOS. [35]

Los bits D (*Delay*), T (*Throughput*) y R (*Reliability*) representan lo siguiente:

Valor	0	1
Bit 3 (D)	Retardo Normal	Bajo Retardo
Bit 4 (T)	Throughput Normal	Throughput Alto
Bit 5 (R)	Confiabilidad Normal	Confiabilidad Alta

Tabla 4.2 Definición de los bits D, T y R. [35]

Los bits 6 y 7 del campo TOS no han sido definidos.

Con el objeto de implementar *DiffServ* en Internet, se ha definido actualmente el campo DS en la cabecera de los paquetes IP para reemplazar al octeto TOS IPv4 y al octeto Clase de Tráfico presente en IPv6.

En el modelo *DiffServ*, los dispositivos en la periferia de la red marcan el DSCP de acuerdo a las necesidades de un flujo de tráfico. El DSCP marcado puede ser interpretado por un proveedor de servicio para asegurar ciertas características de servicio de principio a fin.

El campo DS está formado por ocho bits divididos en dos subcampos: DSCP y CU. El subcampo DSCP lo constituyen seis bits (del 0 al 5), el cual es útil para seleccionar el PHB bajo cuyos tratamientos circulará un paquete en cada nodo. Los otros dos bits restantes del campo DS (6 y 7) forman parte de un subcampo actualmente no utilizado, llamado subcampo CU (*Current Unused*). El valor de los bits CU son ignorados por los nodos capaces de soportar servicios diferenciados en el momento de determinar el PHB a utilizarse. La estructura del campo DS se la puede apreciar en la figura 4.10 que se muestra a continuación.

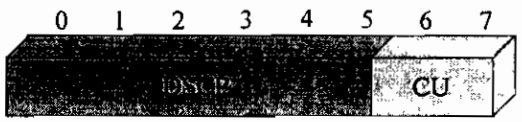

Figura 4.10 Estructura del Campo DS. [36]

Dentro del subcampo DSCP, de seis bits, el bit más significativo corresponde al bit 5, en tanto que el menos significativo corresponde al bit 0.

Como se analizó en la parte final del capítulo anterior, los dos bits del subcampo CU podrían utilizarse para una notificación explicita de congestión (ECN), a fin de "alertar" a los algoritmos de control sobre problemas de congestión en la red para que tomen las acciones adecuadas.

Inicialmente, se propuso dos métodos para la utilización de los bits del DSCP. En el primer método, los cinco primeros bits indican el PHB al cual será sujeto el paquete, en tanto que el sexto bit indica al dispositivo de red si el paquete excede las especificaciones de algún perfil de tráfico. Los dispositivos descartarán todo tráfico que exceda aquellas especificaciones, con el objeto de disminuir la velocidad de envío del origen y evitar efectos adversos en el curso de los paquetes dentro de la red.

En el segundo método, los 6 bits del DSCP indican el comportamiento por salto que será asignado a los paquetes cursantes por la red.

Se escogió y ratificó el uso del segundo método, pues el primer método solo puede diferenciar dos tipos de tráfico: conformante y no conformante, y no permite identificar niveles intermedios de conformación de tráfico que podría ser acondicionado hacia algún perfil.

A fin de mantener la compatibilidad, que debe existir entre el campo DS y el TOS en IPv4, se han conservado valores de DSCP que seleccionan comportamientos por salto que han tenido ya una definición histórica. Así por ejemplo se ha definido un valor de DSCP = 000000, que selecciona un PHB que trata los paquetes al puro estilo de envío *best effort*, es decir envía los paquetes tan pronto como sea posible sin ninguna regla especial de tratamiento. Este comportamiento por salto se lo ha denominado PHB Básico, y debe ser seleccionado para todo valor de DSCP que no se halle estandarizado. Como se aprecia el valor DSCP = 000000, mantiene total compatibilidad con la definición del campo TOS en IPv4 (RFC 791), de la siguiente forma:

- Los bits de precedencia indican un tratamiento habitual.
- El bit D, colocado en 0 indica que los paquetes serán enviados con un retardo normal
- El bit T, colocado en 0 indica envío con *throughput* normal
- Y el bit R, colocado en 0 indica que los paquetes no serán entregados con una alta confiabilidad.

Con el mismo objetivo de mantener la compatibilidad con usos actuales del Campo de Precedencia IP, sin tener problemas futuros en cuanto a flexibilidad, se definen varios PHBs que son compatibles con tratamientos de envío seleccionados por el mencionado campo. Por lo tanto, se hace necesario definir un grupo de valores de DSCP que deban ser asociados con cada uno de esos PHBs. Este grupo de DSCPs toma el nombre de Selectores de Clases y son asignados a los valores de DSCP = xxx000. Los PHBs seleccionados por estos DSCPs "especiales" se les denomina PHBs Selectores de Clase.

Por otro lado se definen un grupo de valores DSCP, que seleccionan PHBs estandarizados, analizados en la siguiente sección (4.5).

Los 6 bits del DSCP permiten 64 posibles valores, que son asignados en tres tipos de grupos: [10]

- Un grupo de 32 DSCPs son definidos para grupos de PHBs estandarizados, y tienen el siguiente formato: xxxxx0.
- Un grupo de 16 DSCPs para ser usados localmente o en forma experimental, los cuales obedecen al siguiente formato: xxxx11.
- Un segundo grupo de 16 DSCPs experimentales, definidos para usarse si el grupo de 32 DSCPs definidos para los grupos PHBs estandarizados llegan a agotarse. El formato establecido para este grupo es: xxxx01.

La implementación de un nuevo esquema de prioridad que utilice dispositivos que soportan *DiffServ* y dispositivos que trabajan en base a Precedencia IP, debe basar su trabajo en los valores de DSCPs cuyo formato es xxx000, ya que este tipo de formato será muy bien entendido por los dos tipos de dispositivos.

4.5 COMPORTAMIENTOS POR SALTO (PHBs, *Per Hop Behaviours*)

Un PHB es una descripción de un tratamiento de envío externamente observable de un nodo DS aplicado a un grupo de paquetes *DiffServ* que posean el mismo valor del DSCP. Es decir un PHB es el instrumento en el cual un nodo se basa para asignar recursos a un BA.

El ejemplo más simplista de un PHB es uno que garantice a un BA la asignación de un porcentaje mínimo de ancho de banda de un enlace, durante un intervalo razonable de tiempo. Un PHB ligeramente más complejo podría garantizar al menos una asignación de un porcentaje del ancho de banda de un enlace, con una proporcional compartición de algún exceso de capacidad transitoria, es decir será más dinámico que el anterior.

Los PHBs podrían ser especificados en términos de varias características;

- sus recursos (por ejemplo, espacio en *buffer* y ancho de banda),
- prioridad relativa sobre otros PHBs,
- o en términos de sus características de tráfico observables (por ejemplo, retardo, pérdida).

Se puede formar grupos PHB mediante la unión de PHBs individuales. Los grupos PHB usualmente comparten tratamientos comunes aplicados a cada PHB dentro del grupo, tales como organización de paquetes o políticas de administración del *buffer*. La forma en la cual se relacionan los distintos PHBs dentro de un grupo es en términos de prioridad absoluta o relativa, como se muestra en la sección 4.5.3.

Los PHBs son implementados en los nodos como instrumentos de acción de algunos mecanismos de administración del *buffer* y organización de paquetes, y

son definidos en términos de características de comportamiento relevantes a las políticas de aprovisionamiento de servicio, y no en términos de mecanismos de implementación particular. En general, dentro de un dominio se pueden definir varios grupos.

Tal como se analizó en la definición del campo DS, un PHB es seleccionado en un nodo analizando el valor del DSCP del paquete recibido. Los PHBs estandarizados tienen valores de DSCP preestablecidos. Todos los DSCP deben ser asociados a algún PHB, en la ausencia de alguna política local, los DSCP que no sean asociados a un PHB estandarizado deben ser tratados con el PHB Básico.

4.5.1 PHB BÁSICO

Como su nombre lo indica es un PHB que otorga características básicas al tráfico que cursa una red, debe estar definido en todos y cada uno de los nodos que implementen servicios diferenciados. Todas las clases de tráfico con un valor de DSCP que no defina algún PHB recibirán este tipo de tratamiento [32].

El valor de DSCP recomendado para este PHB es 000000. El tráfico que es sometido a este comportamiento por salto será enviado tan confiable y rápidamente como sea posible, priorizando algún otro PHB de clase más alta. Los algoritmos de manejo de tráfico deben implementar algún grado de repartición equitativa de la cola para asegurar que las aplicaciones *best effort* nunca queden completamente sin ancho de banda.

4.5.2 PHB DE ENVÍO ACELERADO (EF, *Expedited Forwarding*)

El PHB EF puede ser usado para construir un servicio de principio a fin de baja pérdida, baja latencia, bajo *jitter* y de un ancho de banda asegurado. [37]

Un servicio de tales características aparenta en los extremos finales (usuario) una conexión punto a punto o una línea virtual arrendada. Pérdida, latencia y *jitter* son siempre relativas a los encolamientos que sufre el tráfico mientras transita por la red, por lo tanto para lograr establecer un servicio con las características del PHB

EF se debe asegurar que el tráfico no circule por colas, o si lo hace éstas sean sumamente pequeñas.

Las colas surgen cuando la velocidad de arribo del tráfico excede la velocidad de salida en algún nodo. De esta forma, un servicio que no utilice encolamiento para cierta clase de tráfico es equivalente a limitar la velocidad, es decir la velocidad de arribo máxima del flujo debe ser menor que la velocidad máxima de salida que puede manejar ese dispositivo.

La creación de un servicio EF tiene dos partes:

1. Configura los nodos tal que el BA tenga una velocidad de salida mínima muy bien definida.

2. Condiciona el BA (vía administración y/o formado) tal que su velocidad de arribo en algún nodo sea siempre menor, máximo igual, a la velocidad de salida mínima configurada en el nodo.

El establecimiento de una velocidad de envío muy bien definida significa que ésta debe ser independiente del estado dinámico del nodo, en particular independiente de la intensidad de otro flujo presente en el nodo. La segunda condición la proveen los acondicionadores de tráfico en las fronteras de la red.

4.5.2.1 Descripción del PHB EF

El PHB del tipo EF define un tratamiento de envío para un BA particular, en el cual la velocidad de envío de los paquetes conformantes del BA desde algún nodo debe ser a lo más igual al valor de una velocidad configurable, es decir la velocidad de salida de los paquetes conformantes del BA debe ser una velocidad tal que su valor en el peor de los casos iguale a la velocidad configurada.

Si el PHB EF es implementado por algún mecanismo que le otorgue a un tráfico ilimitados privilegios sobre otras clases de tráfico, debe analizarse el daño que el tráfico EF hará al resto. Una solución a este problema es establecer un limitador de velocidad superior, mediante el cual todo tráfico que exceda este límite sea descartado. Esta máxima velocidad EF debe ser configurada por el administrador

de la red. Si se desea dar un tratamiento sencillo, la máxima y mínima velocidad podrían ser las mismas.

4.5.2.2 Mecanismos para implementar PHB EF

Algunos tipos de mecanismos de encolamiento pueden ser empleados para entregar el servicio propuesto por este PHB. Una simple prioridad de encolamiento entregará el apropiado comportamiento siempre y cuando no exista tráfico de mayor prioridad.

Es posible también usar una sola cola en un grupo de colas servidas por un mecanismo WRR (*Weighted Round Robin*) en el cual, el ancho de banda de salida asignado a la cola EF sea igual a la velocidad preestablecida. Otra implementación posible es un organizador CBQ que asigna el tráfico EF a una clase de alta prioridad.

Los mecanismos anteriormente mencionados tienen las propiedades básicas requeridas por el PHB EF a fin de entregar el tipo de servicio ofrecido en su definición.

El valor de DSCP recomendado para este PHB es 101110. [37]

Como puede analizarse, este valor de DSCP guarda total concordancia con las definiciones dadas en el establecimiento de los bits de la cabecera de un paquete IP. Así los tres primeros bits (101) indican un valor de precedencia igual a Crítico / ECP, el bit D (1) indica que los paquetes circularán con una baja pérdida , el bit T (1) señala que los paquetes utilizarán para su envío un *thoughtput* alto y finalmente el bit R (0) igual que el servicio a entregarse, no goza de opciones de confiabilidad extremas.

La cabecera de un paquete IP que goce del servicio asignado a un PHB EF, es la siguiente:

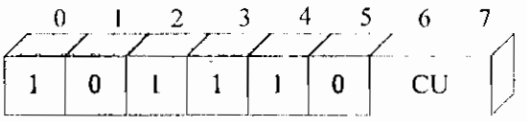

Figura 4.11 DSCP asignado al EF PHB. [37]

4.5.3 GRUPO DE PHBs DE ENVÍO ASEGURADO (AF, *Assured Forwarding*)

Actualmente existe una gran demanda para proveer un envío seguro de los paquetes IP sobre el Internet. En una aplicación típica, una compañía usa el Internet como medio para interconectar sus oficinas distribuidas geográficamente en varias ciudades o en varios sitios dentro de la misma ciudad y desea asegurar que sus paquetes IP en el interior de la intranet sean enviados con una alta prioridad siempre y cuando los paquetes de ese flujo se encuentren dentro de algún perfil de tráfico.

El grupo de PHBs AF fue definido para que un proveedor de Dominio DS esté en capacidad de ofrecer diferentes niveles de envío asegurado para paquetes IP pertenecientes a algún cliente del dominio.

Se estableció que el Grupo PHB AF conste de cuatro clases AF muy bien definidas, donde cada una de ellas está presente en cada nodo, mediante la asignación de cierta cantidad de recursos de envío (espacio en *buffer* y ancho de banda). [38]

Dentro de cada clase los paquetes IP son marcados, ya sea por el cliente o por el proveedor del dominio DS, con uno de tres posibles valores de precedencia de descarte. Así, en caso de problemas de congestión, la precedencia de descarte de un paquete determina la importancia relativa de un paquete dentro de una clase AF.

Como es lógico un nodo DS congestionado trata de evitar que se pierdan los paquetes marcados con baja precedencia de descarte, para ello comienza descartando los paquetes que se hallen señalados con un valor mayor.

En un nodo DS, el nivel de seguridad de envío de un paquete IP depende de los tres factores siguientes:

1. Cuánto de los recursos disponibles para el envío han sido asignados a cada clase AF.

2. Cuál es la carga actual de la clase AF.

3. En caso de congestión dentro de la clase, cuál será la precedencia de descarte.

El Grupo de PHBs AF provee el envió de paquetes IP en *N* clases AF independientes. En el interior de cada clase AF, un paquete IP es asignado a *M* diferentes niveles de prioridad de descarte. Un paquete IP que pertenezca a una clase AF "i" y haya sido marcado con la precedencia de descarte "j" será etiquetado con un valor de DSCP AF_{ij}, donde $1 \leq i \leq N$ y donde $1 \leq j \leq M$.

Como se mencionó anteriormente, en la actualidad se han establecido cuatro clases de AF que incluyen cada una de ellas tres niveles de precedencia de descarte, por lo tanto hoy en día $N = 4$ y $M = 3$ [38]. Cabe mencionar que estos PHBs son de uso general y se encuentran estandarizados, siendo posible definir nuevas clases AF o introducir nuevos niveles de precedencia de descarte para uso local.

Un nodo que sea capaz de soportar *DiffServ* debe necesariamente implementar las cuatro clases de servicio AF. Los paquetes de una clase deben ser enviados independientemente de los paquetes de otra, es decir, un nodo DS no debe juntar dos o más clases. La asignación de recursos de envío a cada una de las clases debe ser un parámetro configurable, a fin de que toda clase sea servida de una manera tal que alcance mínimo la velocidad de servicio preestablecida, y en caso de existir disponibilidad de un exceso de recursos de la red esta velocidad sea superada teniéndose una mayor eficiencia para cada clase.

Los tres niveles de precedencia de descarte tienen que ser implementados también en todo nodo que soporte *DiffServ*, y éstos (los nodos) deben tener al

menos dos niveles de probabilidad de descarte. En algunas redes, particularmente en redes de negocios donde la congestión se produce muy raramente y es de muy corta duración, sería razonable que un nodo DS implemente únicamente dos niveles de probabilidad de pérdida por cada clase AF. En este caso, la clase AF_{x1} debe proveer la probabilidad más baja de pérdida y las clases AF_{x2} y AF_{x3} deben producir la probabilidad de descarte más alta.

En cambio en aquellas redes en las cuales la congestión sea común deben implementarse los tres niveles de precedencia de descarte.

El grupo PHB AF puede ser usado para implementar un servicio diferenciado de principio a fin o un servicio diferenciado entre frontera – frontera dentro de un dominio.

Una implementación AF, requiere que siempre se intente minimizar la congestión sobre períodos de tiempo largos, mientras se permite congestión sobre cortos períodos producto de ráfagas de tráfico. El mencionado requerimiento necesita un algoritmo de administración activa de colas, como los estudiados en el capítulo 3, siendo una muy buena alternativa la utilización del algoritmo de Descarte Aleatorio Temprano (RED). Así, una implementación AF debe detectar y responder a una congestión en largos períodos dentro de cada clase, descartando paquetes, mientras manipula la congestión sobre períodos cortos (ráfagas de paquetes) mediante el encolamiento.

Los valores DSCP recomendados para el uso general de las cuatro clases AF se muestran a continuación:

PROBABILIDAD DE DESCARTE		CLASE AF1j	CLASE AF2j	CLASE AF3j	CLASE AF4j
j = 1	Baja	001010	010010	011010	100010
j = 2	Media	001100	010100	011100	100100
j = 3	Alta	001110	010110	011110	100110

Tabla 4.3 Valores DSCP para PHBs AF. [38]

El nivel de precedencia de descarte de un paquete puede ser asignado, por ejemplo, mediante el uso de un administrador de tráfico que utilice como

parámetros de operación una velocidad y un tamaño, los cuales son la suma de dos valores límites: un valor límite comprometido y un valor límite exceso. Un paquete es asignado a una precedencia de descarte baja si el número de señales en el *bucket* es mayor que el tamaño límite exceso, y asignado a una probabilidad de descarte alta si el *bucket* está vacío.

Es también útil y necesario colocar un límite superior a la cantidad de tráfico de algún cliente del dominio DS que posea una prioridad de descarte baja, a fin de evitar que una gran cantidad de tráfico de este tipo sature los recursos de red y perjudique al tráfico de alta prioridad de descarte de otras fuentes o dominios.

4.6 AGREGACIÓN DE COMPORTAMIENTOS (BA, *Behavior Aggregates*)

Un BA es un conjunto de paquetes que han sido marcados con el mismo valor de DSCP, que atraviesan un enlace en una dirección en particular [39]. Al estar un grupo de paquetes marcados con el mismo DSCP, conseguirán el mismo tratamiento por salto (PHB) en el interior de un dominio DS.

Por lo tanto un flujo de tráfico dentro de un escenario *DiffServ*, será "agregado" a otros flujos que posean o guarden alguna característica de tratamiento común, es decir serán juntados varios tráficos en un solo BA, a fin de establecer unas pocas clases de tráfico y lograr la sencillez que la Arquitectura *DiffServ* espera alcanzar.

Dentro de un dominio DS los distintos BAs se forman mediante las reglas de clasificación y acondicionamiento de tráfico aplicadas a los paquetes que arriban a la frontera.

4.6.1 CARACTERÍSTICAS

Los BAs deben tener propiedades tales que permitan a los paquetes individuales ser caracterizados por medidas especificas. Asociados con cada BA están características medibles y cuantificables que pueden ser usadas para describir lo que sucede con los paquetes de un BA, cuando éste cruza un dominio. Éstas se derivan de las reglas que se implementan a la entrada de los paquetes al dominio

(creación de un BA) y el tratamiento de envío aplicado (PHB) que consigue el tráfico.

Las características de un BA indican su comportamiento bajo condiciones ideales si fue configurado de una manera adecuada. Las características de un BA podrían ser: descarte, velocidad, *throughput*, límites de retardo medidos sobre algún período de tiempo, etc.

Las características mencionadas podrían ser límites tanto absolutos como estadísticos (por ejemplo "95% de todos los paquetes medidos sobre intervalos de al menos 5 minutos cruzarán el dominio DS en un tiempo menor a 5 milisegundos"). Una amplia variedad de características podrían ser usadas, pero éstas deben ser explícitamente cuantificables.

Así un administrador de red, debería usar estas características como una guía para la creación de una especificación de servicio, en lugar de usarlas directamente. Por ejemplo, un "BA libre de pérdidas" probablemente no dice mucho, pero es mas comprensible si se expresa como un servicio cuya probabilidad de pérdida de paquetes es muy pequeña.

4.6.2 PROPIEDADES

Hay dos clases de propiedades que pueden ser especificadas en la definición de una agregación de comportamiento (BAs).

- La primera de ellas es asociada con "largos" períodos de tiempo y es conocida como comportamientos promedio, éstas podrían ser las velocidades o el *throughput* medidas sobre algún período de tiempo específico.

- La segunda clase de propiedades es asociada con "cortos" períodos de tiempo, se refiere específicamente al tamaño de los *buffers* que permiten soportar ráfagas de datos y provocar cambios instantáneos en las características de los BAs.

4.6.3 EJEMPLOS DE AGREGACIONES DE COMPORTAMIENTOS

Esta sección da una idea de cómo se alcanza, se caracteriza y se ejecutan los BAs mediante el análisis de algunos de ellos.

4.6.3.1 Agregación de Comportamientos *Best Effort*

Este tipo de BA se usa para el envío normal de tráfico en Internet a través de una red de servicios diferenciados. La definición y la utilización de éste preserva la actual forma de trabajo de Internet, la entrega sin diferenciación para ciertos paquetes dentro de una red *DiffServ*, que no requieran algún tratamiento particular.

Por lo tanto la velocidad y las ráfagas de paquetes se hallan limitadas únicamente por el enlace de entrada, mas no por alguna regla específica.

La caracterización de este BA es: "tanto y tan pronto como sea posible"; por lo que debe ser configurado para cursar el tráfico cuando los recursos de red se encuentren libres y no estén sirviendo a un BA de mayor jerarquía.

Los tipos de tráfico que podrían considerarse para el BA *best effort*, serían los siguientes:

- Para el tráfico normal de un usuario en su hogar, navegación, chat, *e-mail*.
- Para el tráfico de Internet normal de una organización.

4.6.3.2 Agregación de Comportamientos de Manipulación Voluminosa

Como su nombre lo indica, este BA es adecuado para cursar grandes volúmenes de tráfico que no es crítico. Por lo tanto es de esperarse que los paquetes pertenecientes a este BA sufran prolongados retardos o sean descartados, cuando otro tráfico este presente.

Al igual que en el caso del BA *Best Effort*, ninguna regla gobierna la velocidad de transmisión ni limita el tamaño de la ráfaga de datos. En todo nodo del interior de la red, los paquetes marcados con este BA podrían ser servidos cuando otro tráfico está presente, siempre y cuando existan recursos disponibles.

Los tipos de tráfico que podrían considerarse para conformar este BA, serían los siguientes:

- Para el tráfico de noticias sobre Internet
- Para tráfico no servido por otros BAs.

4.6.3.3 Agregación de Comportamientos Tolerante a las Pérdidas

Este BA es útil para paquetes que deban tener bajo retardo, pero a la vez sean tolerantes a las pérdidas. Se considera tráfico conformante de este BA al tráfico que no exceda una velocidad de transmisión pico y que no presente ráfagas de datos mayores a dos MTUs a dicha velocidad.

Este BA puede ser implementado mediante el PHB AF_{13} cuyo DSCP es 001110 y caracterizado por los siguientes parámetros:

- Asignación de un mínimo ancho de banda.
- Administración activa de colas con un umbral bajo.
- Máximo tamaño de cola de valor pequeño.

4.6.3.4 Agregación de Comportamientos Preferencial

Este BA es de carácter experimental, es decir aún no está estandarizado. Se lo propone para soportar tráfico que requiere de la red un tratamiento preferencial.

El tráfico que aspire a este servicio debe cumplir dos condiciones a su ingreso al dominio DS: velocidad de transmisión promedio menor a una preestablecida, y máximo tamaño de ráfaga menor a uno preestablecido.

Este BA utiliza el DSCP 000100, y tiene las siguientes características:

- Define límites probabilísticos basados en la suma de todas las velocidades de transmisión y tamaños de ráfagas asignados.

- El *Throughput* medido a intervalos de 5 minutos debe ser por lo menos igual a la velocidad de transmisión promedio acordada.

Ejemplos de utilización de este BA pueden ser:

- Servicio de voz, que garantice una tasa de pérdida menor a 0.5% (para el tráfico conformante) y una latencia límite de 20 ms.

- Reemplazo de enlaces dedicados en los cuales se garantiza un *Throughput* determinado y un retardo de 20 ms a través de una nube DS.

El BA Preferencial es un caso ideal que requiere grandes recursos de la red, debido a lo cual está en duda una implementación a futuro. [39]

CLASIFICACIÓN EN LAS DIFERENTES CAPAS

La clasificación del tráfico de red puede realizarse a nivel de capa 3 y capa 2, cada una de las cuales presenta diferentes características, analizadas a continuación.

4.7.1 CLASIFICACIÓN EN CAPA 3

En una empresa, el énfasis está en la priorización de ciertos tipos de aplicaciones identificadas por un número de puerto en lugar de las direcciones origen y destino. Los proveedores de servicio tienen limitado conocimiento de qué aplicaciones (es decir, los números de puertos) un cliente considera importantes, pero ellos desean ofrecer a sus clientes una cierta certeza de que sus paquetes gocen de cierta prioridad a través de su red.

En consecuencia, la información del origen y destino presente en capa 3 es útil para administrar el tráfico de redes vecinas. En una red de ISP, por ejemplo, el puerto TCP o UDP podría entregar información a los dispositivos acerca de la sensibilidad del tráfico al retardo pero no acerca de lo crítico que representa para el cliente ese tráfico.

Por otro lado, la dirección origen de un paquete indicará a los dispositivos del ISP si el usuario tiene o no derecho a obtener un servicio diferenciado. El uso de las direcciones origen y destino son útiles para entregar una clasificación basada en el usuario, siendo este tipo de clasificación posible si se la implementa como parte de un sistema de administración dinámica.

En desarrollos anteriores, las políticas basadas en usuarios son muy complejas para ser utilizadas, pues la tendencia ha sido utilizar políticas basadas en el dispositivo (identificado por direcciones de red). Por ejemplo, una máquina de videoconferencia en el cuarto de proyecciones de una compañía podría tener una dirección IP estática y su tráfico podría ser reconocido examinando únicamente la dirección IP de origen. Los *routers* frontera podrían tener su dirección estáticamente configurada en su lista de acceso o primitivas *DiffServ* que les permitan asignar los recursos adecuados a fin de dar servicio a este tráfico de alta prioridad.

4.7.2 CLASIFICACIÓN EN CAPA 2

En capa 2 se encuentra presente la dirección MAC, la cual es un buen indicador de los usuarios si una organización mantiene una base de datos correcta, en este aspecto las direcciones MAC podrían servir como un mecanismo para asociar usuarios a una VLAN[1] (*Virtual Local Area Network*) determinada.

Debido a que estas direcciones identifican un dispositivo, es poco probable que un sistema basado en las direcciones MAC sea utilizado para clasificar el tráfico, sin embargo una forma de implementación sería mediante la colocación en un *switch* de una tabla de prioridades asociadas con las direcciones MAC. Es importante tener en cuenta que el *switch* no es capaz de diferenciar una aplicación (número de puerto), únicamente puede tomar decisiones a nivel de la capa del medio.

[1] VLAN.- Red LAN Virtual o lógica que define la red de área local sobre la base de alguna característica común que no sea la ubicación geográfica, por ejemplo basándose en el departamento de trabajo, el tipo de usuario o una aplicación primaria.