R. Bless, M. Doll, K. Wehrle und M. Zitterbart

DiffServ-basierte Dienstgüte im Internet der nächsten Generation

Roland Bless studierte von 1990 bis 1996 Informatik an der Universität Karlsruhe (TH) und promovierte 2002 am dortigen Institut für Telematik über integrierte Managementarchitekturen für DiffServ. Forschungsschwerpunkte sind Dienstgüte und Dienstgütemanagement. Mitarbeit in den IETF Arbeitsgruppen DiffServ und Nsis.

Mark Doll studierte von 1992 bis 2000 Physik an der Technischen Universität Braunschweig. Als wissenschaftlicher Mitarbeiter am dortigen Institut für Betriebsysteme und Rechnerverbund wechselte er 2001 an das Institut für Telematik. Forschungstätigkeit im Bereich Dienstgüte, Dienstgütemanagement.

Klaus Wehrle studierte von 1993 bis 1999 an der Universität Karlsruhe (TH) Informatik und promoviert seitdem am Institut für Telematik über die flexible Realisierung von skalierbaren Dienstgütemechanismen. Ab September 2002 Postdoc-Stipendium am ISCI der UC Berkeley. Standardisierungsbemühungen in der IETF im Bereich DiffServ.

Prof. Dr. Martina Zitterbart promovierte 1991 und habilitierte 1994 in Informatik an der Universität Karlsruhe (TH). 1991 und 1992 Visiting Scientist am IBM T. J. Watson Research Laboratory, New York, USA, 1994 und 1995 Vertretungsprofessuren in Magdeburg und Mannheim, 1995 bis 2001 Professorin für Informatik an der TU Braunschweig und seit 2001 Leiterin des Instituts für Telematik. Arbeitsgebiete: Hochleistungskommunikation (Dienste, Protokolle und Implementierungen), Kollaborations-Anwendungen, Mobile Systeme.

KURZFASSUNG

Die Differentiated Services Architektur der Internet Engineering Task Force (IETF) bietet eine gute Grundlage, um flexibel Dienstgüte im Internet der nächsten Generation bereitzustellen. Dieser Beitrag führt zunächst in die grundlegenden Konzepte und Mechanismen der Architektur ein. Anschließend werden ausführlich die einzelnen Mechanismen und der aktuelle Stand ihrer Standardisierung in der IETF beschrieben. Den Abschluss bilden eigene Forschungsbeiträge zu diesem Thema.

1 EINLEITUNG

Erweitertes Traffic Engineering umfasst – neben der üblichen Verkehrssteuerung zur besseren Auslastung vorhandener Ressourcen – auch die unterschiedliche Behandlung verschiedener Verkehrsströme zur Erreichung differenzierter Dienstgüte. So bietet beispielsweise Multi-Protocol Label Switching (MPLS) zwar die Möglichkeit, den Verkehrsfluss gezielt zu steuern, jedoch können dadurch alleine noch keine Dienste mit einer besseren Dienstgüte erreicht werden. Die Paketweiterleitung erfolgt – ohne Nutzung entsprechender Garantien der darunter liegenden Netztechnik – weiterhin nach dem Prinzip der „Best-Effort"-Zustellung, d.h. das Netzwerk versucht, alle Pakete gleichberechtigt entsprechend den vorhandenen Ressourcen bestmöglich weiterzuleiten, ohne eine Garantie bezüglich der Paketankunft abzugeben.

Überdies kann ein Internet-Dienstbetreiber durch Einsatz entsprechender Dienstgüte-basierter Kommunikationsdienste beispielsweise Virtuelle Private Netze mit garantierten Bandbreiten realisieren, basierend auf dem existierenden Netz, d.h. ohne dass dedizierte neue physikalische Verbindungen eingerichtet werden müssen. Deshalb ist die Realisierung von Dienstgüte im Internet insbesondere im Umfeld des Traffic Engineering interessant und sinnvoll.

2 Die Differentiated-Services-Architektur

Die Einführung unterschiedlicher Dienste für die Paketweiterleitung im Internet ist schon seit geraumer Zeit das Ziel verschiedener Arbeitsgruppen innerhalb der Internet Engineering Task Force IETF. Nachdem ein erster Ansatz (Integrated Services) für die Realisierung integrierter Dienste sich für einen Internet-weiten Einsatz aufgrund seiner schlechten Skalierbarkeit als weniger gut geeignet herausstellte, wurde ein neuer Ansatz verfolgt [1].

"Differentiated Services" (DiffServ, DS) ist der aktuelle Ansatz der IETF, um Dienste mit unterschiedlicher Dienstgüte im Internet einführen zu können. Primäres Ziel der DiffServ-Arbeitsgruppe ist es, „skalierbare" Basismechanismen zu definieren, mit denen qualitätsbasierte Dienste

Internet-weit realisiert werden können. Der Skalierbarkeitsaspekt bedeutet, dass das Konzept auch für um Größenordnungen wachsende Parameter wie Anzahl der Dienstnutzer und Anzahl der Datenströme noch effizient funktioniert. Der zuvor definierte „IntServ“-Ansatz für integrierte Internet-Dienste hatte aufgrund seiner Zustandshaltung pro Datenstrom in jedem Router entlang des Weges ein Skalierbarkeitsproblem bezüglich der Anzahl der Datenströme, so dass ein Einsatz in Kernbereichen des Internets, in denen große Verkehrsmengen zusammenfließen, nicht mehr sinnvoll erscheint [2].

Neben der Skalierbarkeit wurden weitere Entwurfsziele definiert. Es sollten beispielsweise die existierenden Internetstrukturen, d.h. verschiedene autonome Verwaltungsbereiche, berücksichtigt werden. Ebenso sollten flexible Basismechanismen entworfen werden, die nicht von speziellen Diensten oder Anwendungen abhängen. Die Dienste selbst sind derzeit nicht Gegenstand der Standardisierung, da sie meistens relativ schnell anhand von Marktgegebenheiten durch Betreiber umgesetzt oder angepasst werden. Ein weiterer wichtiger Aspekt ist auch, dass Anwendungen nicht verändert werden müssen, um dienstgütebasierte Kommunikation nutzen zu können. Stattdessen müssen lediglich die Router entsprechend konfiguriert werden. Adaptive oder dienstgütefähige Anwendungen stellen allerdings weiterhin eine sinnvolle Ergänzung dar [16].

Die reinen Weiterleitungsmechanismen sind von den Funktionen zur Verkehrsbeeinflussung (Traffic Conditioning – beispielsweise Verkehrsformung oder Verwerfen von nichtkonformen Verkehr) und der Kontrollebene separiert worden. Die komplexeren Funktionen der Kontrollebene sollen erst später behandelt werden, denn zunächst können die meisten Mechanismen auch manuell und statisch konfiguriert werden. Es ist jedoch vorgesehen, dass später automatisch Dienste von Ende zu Ende auf Anforderung mit Hilfe einer entsprechenden Verwaltung bereitgestellt werden.

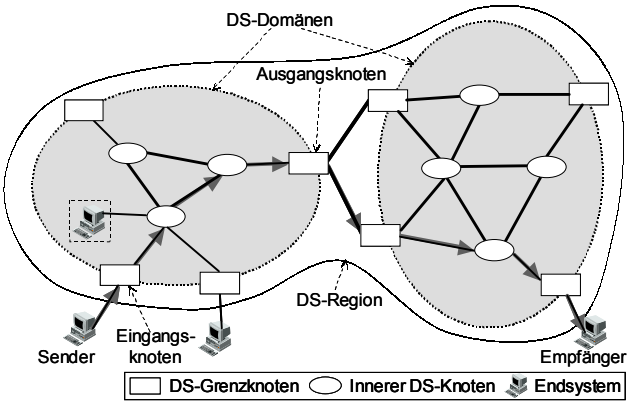

Abbildung 1: DiffServ-Domänen

Um das primäre Ziel der Skalierbarkeit zu erreichen, wird vor allem das Prinzip der Aggregation angewendet. Anstatt einzelne Datenströme zwischen zwei Anwendungen – so genannte „Microflows“ – in jedem Router zu unterscheiden, werden nur wenige Dienstklassen (bis 64) anhand einer kurzen Kennung im Paketkopf, dem DS Codepoint (DSCP), unterschieden [3]. Der DSCP wird im ursprünglichen ToS-Feld von IP (Version 4, kurz IPv4) bzw. Class-Feld von IPv6 übertragen, so dass keine Kopferweiterungen notwendig sind. Aufwändigere Klassifikationen von Paketen anhand der Quell- und Zieladressen werden nur noch an den Netzgrenzen durchgeführt, wo die Anzahl der zusammenfließenden Datenströmen noch handhabbar ist. Die Router verwenden den DSCP, um das konkrete Weiterleitungsverhalten des Pakets zu selektieren. Dieses so genannte „Per-Hop Behavior“ (PHB) wird durch bestimmte Warteschlangenmechanismen realisiert, um die gewünschte differenzierte Behandlung während der Paketweiterleitung zu erreichen.

Die DiffServ-Architektur [4] definiert eine DiffServ-Domäne (vgl. Abbildung 1) als einen administrativen Bereich, der DiffServ-konforme Knoten (Router oder auch Endsysteme) enthält. Während innere Knoten (Interior Router) sich meistens auf eine einfache DSCP-Klassifikation und die Implementierung mehrerer PHBs beschränken, enthalten die Grenzknoten (Boundary Router) am Rand einer Domäne üblicherweise mehr Funktionalität, um den durch die Domäne fließenden Verkehr überwachen zu können. Je nach Lage werden sie in Eingangsknoten (Ingress Router) und Ausgangsknoten (Egress Router) unterschieden. Die Rolle des so genannten First Hop Routers, welcher die Pakete initial mit einem DSCP markiert, übernimmt entweder der Sender selbst oder der erste Router auf dem Weg zum Empfänger, der damit automatisch zu einem Grenzrouter wird, auch wenn er sich im Inneren einer DS-Domäne befindet.

Das DiffServ-Modell sieht vor, dass ein Dienstnutzer sich zunächst an seinen Dienstbetreiber wendet und mit ihm eine Dienstleistungsvereinbarung (Service Level Agreement – SLA) abschließt. Letztere enthält auch technische Parameter, die den gewünschten Dienst genauer spezifizieren, beispielsweise welcher Durchsatz unter welchen Umständen garantiert wird. Insbesondere wird festgelegt, welcher Verkehr des Dienstnutzers als nicht mehr „konform“ angesehen wird, d.h. für diesen Verkehr können vom Dienstbetreiber üblicherweise bestimmte Parameter nicht mehr zugesichert werden. Was im Einzelnen mit solchen nicht-konformen Paketen passiert, also ob sie z.B. verworfen oder anders markiert werden, ist ebenfalls Gegenstand der Vereinbarung. Diese Parameter werden daher in Form eines „Verkehrsprofils“ in dem entsprechenden Grenzrouter am Kundenzugang abgelegt. Liegt das Ziel nicht innerhalb der gleichen DS-Domäne, so muss der Domänenbetreiber den nächsten benachbarten Domänenbetreiber in Richtung des Ziels kontaktieren, um ebenfalls eine Dienstleistungsvereinbarung zu schließen.

Der Kunde kann nach erfolgreicher Dienstaushandlung seine Pakete sodann zum Grenzknoten der Domäne schicken. Dieser markiert als First Hop Router zunächst mit Hilfe eines Mehrfeld-Klassifizierers, der z.B. Quell- und Ziel-IP-Adressen, Quell- und Ziel-Ports sowie das Protokoll berücksichtigt, die Pakete mit dem entsprechenden DSCP. Als Ingress-Router überprüft er den Verkehrsstrom auf Konformität zum (den Dienstvertrag entsprechend) konfigurierten Profil. Der Router überprüft die Konformität des Pakets. Weitere Funktionen wie Verkehrsformung zur Erzeugung möglichst regelmäßiger Verkehrsmuster können ebenfalls dort ausgeübt werden. Die Pakete erfahren anschließend ein ihrer DSCP-Markierung zugeordnetes Weiterleitungsverhalten (PHB). Der nächste innere Knoten enthält üblicherweise nur noch einen Codepoint-

Klassifizierer, der das zugehörige PHB selektiert, entsprechend dem das Paket dann weitergeleitet wird. Beim Verlassen der Domäne wird anschließend der Ausgangsgrenzrouter nochmals die Konformität des Verkehrs mit dem Vertrag der Nachbardomäne sicherstellen, beispielsweise durch Verkehrsformung. Der Eingangsgrenzrouter der Nachbardomäne wird seinerseits überprüfen, ob der dort ankommende Verkehr zur getroffenen Dienstleistungsvereinbarung passt. Alle Router, die nach dem ersten DiffServ-Router entlang des Weges passiert werden, unterscheiden also nur noch Aggregate mit unterschiedlichem Verhalten. Anders ausgedrückt gehören alle Pakete mit gleichem DSCP auf einer Teilstrecke in einer Richtung zu einem „Verhaltensaggregat“ (Behavior Aggregate), d.h. sie sollten im nächsten Knoten prinzipiell das gleiche Weiterleitungsverhalten erfahren.

Das funktionale Modell eines DiffServ-Routers ist in Abbildung 2 dargestellt. Welche Funktionen und Elemente real vorhanden sind, hängt hauptsächlich von der Position bzw. Rolle des Routers ab. Innere Knoten werden üblicherweise nur über Codepoint-Klassifizierer und entsprechende Warteschlangenmechanismen in den Ausgangsschnittstellen zur Realisierung der PHBs verfügen, während Grenzknoten zusätzliche Funktionen wie Mehrfeld-Klassifizierer, Messelemente und Markierer, Verwerfer oder Verkehrsformer an den Eingängen enthalten können. Die konkrete Implementierung eines PHBs ist nicht standardisiert, so dass ein Betreiber wählen kann, welche Mechanismen er zur Realisierung einsetzt. Während die Basismechanismen im Datenpfad weitgehend standardisiert wurden, sind die ebenfalls in Abbildung 2 angedeuteten Mechanismen in der Kontrollebene hingegen noch weitgehend Gegenstand der Forschung. Die Standardisierung einer Management Information Base zur Konfigurierung der DiffServ-Mechanismen ist nahezu abgeschlossen.

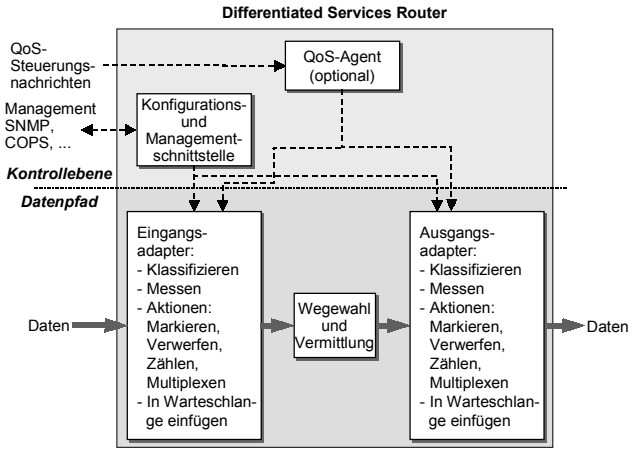

Abbildung 2: Funktionales Routermodell

Um einen weiteren Schritt in Richtung konkreter Dienste voranzukommen, wurde das Konzept des Per-Domänen-Verhaltens „Per-Domain Behavior“ (PDB) definiert [5], welches die erwartete Behandlung von Paketen von Randknoten zu Randknoten der Domäne angibt. Mit einem PDB werden ein bestimmtes PHB (bzw. eine Menge von PHBs) und die Anforderungen zur Verkehrsbeeinflussung verbunden. Daraus lässt sich eine Menge von messbaren, quantifizierbaren Attributen ableiten, die ebenfalls mit dem PDB assoziiert werden, beispielsweise ein Verlustattribut in folgender Form: „Höchstens 0.001% der Pakete werden verworfen, gemessen über einem Intervall, welches größer als T ist“. Im Unterschied zu einer PHB-Definition müssen Auswirkungen auf ein Verhaltensaggregat durch Aggregations- und Deaggregationsvorgänge explizit betrachtet und ggf. durch Regeln definiert werden. Durch wiederholte Anwendung der spezifizierten Regeln an den internen Aggregationspunkten einer Domäne soll es möglich werden, quantifizierbare Aussagen über ein Aggregat zu treffen. Die Definitionen und Parameter von sinnvollen PDBs werden derzeit vielerorts untersucht, sowohl durch Simulationen als auch durch reale Implementierungen, da eine realistische analytische Untersuchung sich aufgrund der Aggregations- und Deaggregationseffekte als recht aufwändig und schwierig erweist [5].

Eine PDB-Definition unterscheidet sich von einer Dienstdefinition, wie sie beispielsweise in einer Dienstleistungsvereinbarung getroffen wird, da technische Details und Attribute der PDB-Implementierung einem Dienstkunden üblicherweise nicht offengelegt werden. Es wird davon ausgegangen, dass ein Dienstanbieter selbst entscheidet, welche Attribute in Dienstverträgen dem Kunden gegenüber offengelegt werden. Diese Attribute können sehr verschieden von den in einer PDB-Definition festgelegten technischen Attributen sein. So können beispielsweise Angaben über die erwartete Verlustrate in der Dienstleistungsspezifikation abgeschwächt oder entfernt werden, um im Falle der Nichteinhaltung Strafzahlungen zu entgehen.

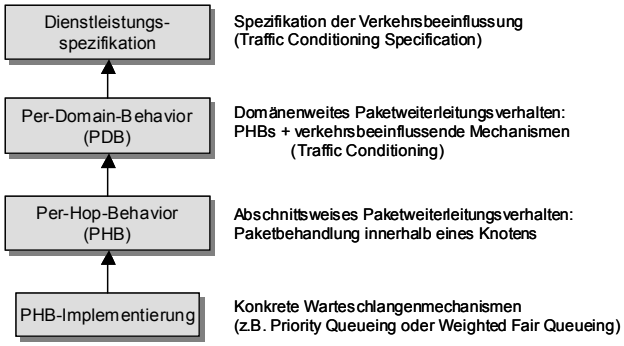

Abbildung 3: Dienstbausteine der DiffServ-Architektur

Eine PDB-Definition stellt somit einen weiteren technischen Baustein dar, den ein Dienstanbieter einsetzen kann, um Dienste zu entwickeln (vgl. Abbildung 3). Diesen Dienst kann er zunächst innerhalb seiner eigenen Domäne anbieten, bevor entsprechende Vereinbarungen mit anderen Dienstbetreibern getroffen werden. In einem weiteren Schritt können Ende-zu-Ende-Kommunikationsdienste durch eine Menge von PDBs realisiert werden.

3 Per-Hop und Per-Domain Behavior

Trotz einer Vielzahl von Entwürfen unterschiedlichster PHBs wurden von der IETF nur zwei Per-Hop Behavior standardisiert, die „Assured Forwarding“ (AF) PHB Group [6] sowie das „Expedited Forwarding“ (EF) PHB [7]. Die darauf aufbauenden Per-Domain Behavior „Assured Rate“ (AR) PDB [8] bzw. „Virtual Wire“ (VW) PDB [9] sind dagegen nicht soweit fortgeschritten und befinden sich, insbesondere das VW PDB, noch im Entwurfsstadium.

Daneben stellt die DiffServ-Architektur [3] in Form so genannter „Class Selector Codepoints“ die Kompatibilität zum bisherigen Gebrauch des ToS-Feldes gemäß RFC 791 sicher. Sie umfassen acht Werte, die den acht relativen Prioritäten (Precedence) „Network Control" bis „Routine" entsprechen. Die DSCP-Werte sind so gewählt, dass keines der alten ToS-Attribute Verzögerung, Durchsatz und Zuverlässigkeit gesetzt ist. Darüber hinaus muss jeder DiffServ-konforme Router Paketen mit DSCP Null weiterhin das Standardweiterleitungsverhalten (Default PHB), üblicherweise als „Best Effort“ bezeichnet, zuordnen. Pakete der beiden damals höchsten Prioritäten „Network Control" und „Internetwork Control" sind gegenüber Best Effort bevorzugt zu behandeln, um den Routing-Kontrollverkehr nicht DiffServ-fähiger Router vor dem übrigen Verkehr zu schützen.

3.1 Assured Forwarding Per-Hop Behavior

Das Assured Forwarding RFC definiert N unabhängige AF-Klassen, innerhalb derer jedes Paket einer von M verschiedenen Verwurfswahrscheinlichkeiten zugeordnet wird, abgekürzt als AFij mit $1 \leq i \leq N$ und $1 \leq j \leq M$. Zur Zeit sind insgesamt 12 DSCPs, verteilt auf vier Klassen (N = 4) mit je drei Verwurfsprioritäten (M = 3) AFx1 – AFx3 zur allgemeinen Verwendung standardisiert. Pakete der niedrigsten Verwurfspriorität 1 dürfen im statistischen Mittel nicht häufiger verworfen werden als Pakete der Priorität 2 aus derselben Klasse. Wenn nur zwei unterschiedliche Verwurfswahrscheinlichkeiten implementiert sind, ist den Prioritäten 2 und 3 dieselbe Wahrscheinlichkeit zugeordnet, ansonsten gilt obige Relation auch zwischen den Prioritäten 2 und 3.

Zwischen unterschiedlichen Klassen bestehen dagegen keine Abhängigkeiten, im Gegenteil, die AF-Definition fordert, dass Verkehrsströme verschiedener Klassen nicht gemeinsam weitergeleitet werden, sie dürfen demnach nicht aggregiert behandelt werden. Jede AF-Klasse erhält einen konfigurierbaren Anteil der verfügbaren Ressourcen wie Bandbreite und Pufferplatz. AF schreibt keine konkrete Implementierung vor, stellt jedoch die Bedingung, dass sie in der Lage sein sollte, jeder Klasse mindestens die zugesicherten Ressourcen bereitzustellen, gemessen über kurze wie auch über lange Zeiträume. Ebenso implementierungsspezifisch bleibt die Strategie zur Aufteilung nicht in Anspruch genommener Ressourcen auf andere Aggregate. Damit stellt AF ein generisches Grundgerüst zur Differenzierung des Verkehrs in verschiedene Verhaltensaggregate bereit.

3.2 Assured Rate Per-Domain Behavior

Anwendung findet Assured Forwarding im Assured Rate Per-Domain Behavior. Es ist für Aggregate gedacht, die eine zugesicherte Bandbreite benötigen, auf exakte Delay- und Jittergrenzen jedoch verzichten können und eignet sich damit zum Aufbau virtueller privater Netze (Virtual Private LANs – VPN), sofern über sie keine Pakete interaktiver Realzeitanwendungen transportiert werden müssen. AR nutzt zur Paketweiterleitung eine einzelne AF-Klasse. Solange ein Aggregat die zugesicherte, als „Committed Information Rate“ (CIR) bezeichnete Rate nicht überschreitet, werden die Pakete mit der Prioritätsstufe 1, bei AR auch als „grün“ bezeichnet, markiert. Das Einhalten der Rate kann ein Ingress-Router durch Mittelung des Verkehrsaufkommens über ein gewisses Zeitintervall T_1 oder alternativ auch über einen Token Bucket mit der CIR als Tokenrate und der „Committed Burst Size“ (CBS) als Buckettiefe bestimmen.

Nicht-konforme Pakete werden nicht sofort verworfen, sondern als gelb oder rot markierte Pakete mit einer höheren Verwurfswahrscheinlichkeit (AF-Priorität 2 oder 3) weitergeleitet, um ungenutzte Ressourcen anderer Aggregate verwenden zu können. Es steht den jeweiligen Betreibern frei, nicht-konforme Pakete einheitlich entweder alle gelb oder alle rot zu markieren, oder eine weitere Differenzierung des nicht-konformen Verkehrs beispielsweise anhand eines zweiten Token Buckets mit den Parametern Peak Information Rate (PIR) und Peak Burst Size (PBS) bzw. anhand einer zweiten Mittelung über ein anderes Zeitintervall T_2 durchzuführen und so im zweiten Schritt „konforme“ Pakete gelb, den Rest rot zu markieren.

Um die CIR tatsächlich zusichern zu können, wird über die Definition in AF hinausgehend gefordert, dass im Normalfall in Priorität 1 keinerlei Paketverluste auftreten. AR spricht trotzdem nur von „low drop probability“, da Paketverluste nie ganz ausgeschlossen werden können. Da AF explizit aktives Warteschlangen-Management in Form von RED oder RIO (Random Early Detect bzw. Random Early Discard with In and Out) vorschlägt, müssen bei AR die Parameter so gewählt werden, dass es für grün markierte Pakete zu keinem Paketverlust durch aktives Queue-Management kommt, es für grün also faktisch deaktiviert wird. Gelbe und rote Pakete werden weiterhin vorzeitig verworfen.

Wie im Rahmen dieses Färbungsschemas tatsächlich gemessen und markiert werden soll, überlässt AR der Traffic Conditioning Specification (TCS), also dem Teil der Service Level Specification (SLS), der das vereinbarte Verkehrsprofil zusammen mit den Verfahren zur Konformitätsprüfung sowie den Umgang mit nicht konformen Verkehr beschreibt (Traffic Profile, Metering, Policing, Shaping). Die SLS wiederum umfasst alle technischen Aspekte eines Dienstvertrags (SLA) und bestimmt damit neben anderen Einflüssen wie der Topologie der Diffserv-Domäne die Parametrisierung eines PDB. Bei AR bestimmt z.B. die im SLS zu vereinbarende Rate den Bandbreitenanteil, den die verwendete AF-Klasse erhalten muss, bei Einsatz von Weighted Fair Queueing zur Bedienung der verschiedenen AF-Klassen also beispielsweise die Gewichtung der einzelnen Warteschlangen. Zusammen mit den wirtschaftlichen und betreiberpolitischen Aspekten eines Dienstvertrags bildet die SLS schließlich den aus der Einleitung bekannten Dienstvertrag oder Service Level Agreement (SLA).

3.3 Expedited Forwarding Per-Hop Behavior

Das EF PHB basiert auf der Idee des Premium Service [10] und beschreibt ein Weiterleitungsverhalten mit möglichst kleinem Paketverlust und vor allem kleiner Verzögerung (Low Loss, Low Delay). Wie im folgenden noch erläutert wird, verschlechtern sich seine Eigenschaften mit zunehmender Nutzung, was den Premium Service zu einem kostbaren Dienst macht. Es empfiehlt sich daher, ihn ausschließlich für jene Anwendungen, namentlich für die Klasse der interaktiven Realzeitanwendungen wie Voice over IP (VoIP) einzusetzen, die seine Eigenschaften zwingend erfordern.

Zur Erzielung dieser Eigenschaften muss eine geeignete Verkehrskontrolle (Policing, Shaping) an den Domänengrenzen dafür sorgen, dass in jedem Router in der Domäne die Summe der Datenraten aller der Eingänge kleiner als die konfigurierte EF-Rate des Ausgangs ist, über den deren Pakete den Router wieder verlassen. Dies soll verhindern, dass sich eine Warteschlange aufbaut und nachfolgende Pakete nur verzögert weitergeleitet werden können. Pakete, die von Ingress-Routern als nicht konform erkannt werden, müssen verworfen werden, da eine Degradierung durch Ummarkieren beispielsweise zu einem Best-Effort-Paket die kleine Verzögerung zerstören und es damit ohnehin nutzlos machen würde. Bei richtiger Auslegung der Filter an den Domänengrenzen sollten innere Router nie Pakete verwerfen müssen. Das konkrete Zuteilungsverfahren wird wie bei AF offen gelassen. Nur für den Fall, dass EF-Verkehr übrigen Verkehr beliebig lange blockieren kann, wenn also EF in eine Prioritätswarteschlange mit höchster Priorität eingeordnet wird, fordert der Standard, dass der anderen Aggregaten zugefügte Schaden begrenzt ist.

Die allgemein als kanonisch erachtete EF-Implementierung mittels einer Prioritätswarteschlange höchster Priorität soll nun etwas genauer betrachtet werden. Durch die höchste Priorisierung stellt sie die kleinstmögliche Weiterleitungsverzögerung sicher, andernfalls entspräche das gewonnene Verhalten eher dem von AF. Der Schaden für andere Aggregate wird dadurch begrenzt, dass Verkehrskontrolle an den Domänengrenzen sicherstellt, dass auf allen inneren Netzabschnitten die mittlere EF-Rate einen bestimmten Anteil an der jeweils zur Verfügung stehenden Gesamtbandbreite nicht überschreitet. Unter der „konfigurierten" EF-Rate ist dann jener Anteil zu verstehen. Schließlich verbietet der Wunsch nach geringer Verzögerung allzu häufiges Shaping, z.B. an inneren Knoten, da Shaping prinzipiell die Verzögerung erhöht, was unerwünscht ist. Es widerspräche auch dem Diffserv-Paradigma, das versucht, Skalierbarkeit durch ein möglichst einfaches Kernnetz sicherzustellen.

Bei einer solchen zustandslosen Auslegung des Kernnetzes kommt es durch Aggregationseffekte zwangsläufig zu Bursts. Bursts bedeuten, dass kurzzeitig die Bedingung „Summe der Eingangsraten kleiner Ausgangsrate" verletzt wird. Es bildet sich kurzzeitig eine Warteschlange, die wiederum zur Verzögerung folgender Pakete und schließlich zu Jitter führt, da manche Pakete eine Warteschlange sehen, spätere Pakete des gleichen Datenstroms jedoch wieder eine leere Warteschlange vorfinden. Und dieser Effekt verstärkt sich: Da es sich um eine Prioritätswarteschlange mit höchster Priorität handelt, verlassen alle gepufferten Pakete den Router in einem noch längeren Burst, der im nächsten Router in Richtung Empfänger zu noch größeren Warteschlangen und wiederum längeren Bursts führt.

Es lässt sich auch eine Topologie konstruieren, wenn auch eine sehr artifizielle, bei der die Worst-Case-Abschätzung der Verzögerung ins unendliche wächst [11], obwohl auf jedem Netzabschnitt die maximale Auslastung begrenzt ist und kein Paket mehr als eine bestimmte kleine Anzahl von Knoten passiert. Bei realen Systemen mit endlichen Puffern ist die Verzögerung eines Paketes durch die Summe der Puffergrößen aller auf seinem Weg liegenden Router begrenzt. In einem solchen Netz würden stattdessen Paketverluste einsetzen. Man kann sich zur Entstehung vorstellen, dass Pakete beim Passieren des Netzes einen anderen Paketstrom kreuzen und diesen so verzögern, dass sich ein kleiner Burst von zwei Paketen bildet. Dieser verzögert bei seinem Weg durch das Netz zusammen mit anderen ebenso entstehenden Zweier-Bursts wiederum einen kreuzenden Strom zu einem noch längeren Drei-Pakete-Burst usw.

Eigene Simulationen wie auch jene zum VW PDB [10] zeigen jedoch, dass dieser Worst Case derart unwahrscheinlich ist, dass er in realen Systemen mit ihrem statistischen Multiplexing nicht von Bedeutung sein sollte. Es sei nochmals klargestellt, dass Verkehrsformung im Inneren der Domäne keinesfalls den maximalen Jitter verringern kann, sondern lediglich verhindern hilft, dass die Bursts soweit anwachsen, dass es zu Pufferüberläufen bzw. Paketverlusten kommt. Zur Veranschaulichung führe man sich vor Augen, dass Pakete am Anfang eines Bursts leere Warteschlangen vorfinden und auf Token Buckets mit ausreichend Token treffen, also keine Zeit in Puffern verbringen. Die so erfahrene minimale Verzögerung ist praktisch ausschließlich durch die physikalische Paketlaufzeit bestimmt. Pakete im hinteren Teil eines Bursts dagegen wurden bereits durch eine volle Warteschlange verzögert und finden nun auch noch einen leeren Token Bucket vor, wodurch sie abermals verzögert werden. Verkehrsformung in inneren Routern erhöht also die maximale Verzögerung ohne die minimale anzuheben. Der maximale Jitter als Differenz der beiden Werte steigt.

Damit Routerhersteller die Konformität zum Standard wie auch die Güte ihrer Produkte beschreiben können, ohne Details über die von Ihnen gewählte Implementierung preisgeben zu müssen, definiert der Standard auf eine Fehlergröße E_a. Sie gibt die maximale Differenz zu einem nach dem kontinuierlichen Referenzmodell („Fluid Model") iterativ berechneten idealen Absendezeitpunkt an. Abweichungen vom Ideal entstehen durch verschiedene Zuteilungsstrategien und routerinterne Verzögerungen bei der Paketweiterleitung. E_a wird typisch als konstante obere Schranke oder auch in Abhängigkeit der EF-Rate angegeben.

Die Referenz bei der Berechnung bildet der ideale Sendezeitpunkt f_i, an dem das letzte Bit des EF-Paketes der Länge L den Router verlassen hätte, wenn man annimmt, dass Pakete Bit für Bit mit genau der konfigurierten Rate R gesendet würden

$f_0 = 0,\ d_0 = 0,$

$f_i = \max(a_i, \min(d_{i-1}, f_{i-1})) + L/R, \quad i > 0.$

Hierin bezeichnet a_i den Ankunftszeitpunkt des letzten Bits des Paketes – bevor ein Paket empfangen wurde, kann es den Router auch nicht verlassen – und d_{i-1} den Zeitpunkt, an dem das letzte Bit des vorhergehenden EF-Paketes gesendet wurde. Router in paketvermittelten Netzen senden nur ganze Pakete mit voller Bandbreite (Line Rate) C in der kürzeren Zeit L/C. Der Fehlerterm E_a erlaubt einem anzugeben, wann im Vergleich zum Referenzzeitpunkt ein EF-Paket spätestens den Router verlassen haben wird

$d_i \leq f_i + E_a, \quad i > 0.$

Bei einem idealen Router ohne interne Verarbeitungszeit und Warteschlangen ausschließlich an den Ausgängen ergibt sich je nach Bedienstrategie z.B. für eine Prioritätswarteschlange höchster Priorität

$d_i = L/C + MTU/C.$

Der zur Paketzeit L/C hinzuzuaddierende Term MTU/C entsteht dadurch, dass kein preemptives Scheduling eingesetzt wird, weil es aus verschiedenen Gründen wie der gerade unter hoher Last immer schlechter werdenden effektiven Durchsatzes nicht wünschenswert ist. So muss ein EF-Paket trotz höchster Priorität u.U. warten, bis der bereits begonnene Sendevorgang eines maximal großen Paketes eines anderen Verhaltensaggregats abgeschlossen ist. Ohne Verkehrsformung existiert keine konfigurierte Rate R, man setzt hier die mittlere Rate des EF-Aggregats auf dem Ausgang an. Drückt man diese als Auslastung a = R/C aus, erhält man schließlich als Fehlerterm

$E_a = L/C(1 – 1/a) + MTU/C.$

oder als Konstante für den Worst Case von 100% Auslastung E_a = MTU/C (Maximum Transfer Unit – MTU). Es sei hier angemerkt, dass, auch wenn der Fehlerterm positiv ist, unabhängig davon im Mittel $d_i \leq f_i$ gelten muss, das „mittlere E_a" demnach null oder negativ sein muss. Andernfalls würde der Router die konfigurierte Rate gar nicht erreichen.

Bei realen Routern führen interne Vermittlungszeiten z.B. der Switch Fabric zu weiteren Verzögerungen. Die interne Arbeitsweise kann dazu führen, dass ein Paket, obwohl später eingetroffen als ein zweites auf einem der anderen der Eingänge empfangenes Paket, den Router noch vor dem ersten Paket verlässt (beide über denselben Ausgang). Deshalb definiert der EF-Standard einen alternativen Fehlerterm E_p, der sich von E_a insofern unterscheidet, als dass die Paketnummerierung beim Eintreffen und nicht beim Verlassen des Routers erfolgt, wobei obige Gleichungen nach Ersetzen von E_a durch E_p unverändert gelten. Für ideale Router sind E_p und E_a identisch.

Der maximale Jitter, der beim Passieren eines Routers entsteht, lässt sich mit der maximalen Verzögerung abschätzen, da die minimale Verzögerung größer null ist. Entscheidend ist hier die Burstiness des eingehenden Verkehrs. Die maximale Verzögerung D_{max} ist durch

$D_{max} = B/R + E_p$

nach oben hin beschränkt, sofern sich das EF-Aggregat, also die Summe aller EF-Eingangsströme für einen bestimmten Ausgang, durch einen Token Bucket mit der Tiefe B und einer Rate $r \leq R$ beschreiben lässt. Bei der vorgestellten Implementierung durch eine Prioritätswarteschlange höchster Priorität ist sie mit

$D_{max} = (B + MTU)/C$

minimal.

3.4 Virtual Wire Per-Domain Behavior

Das Virtual Wire PDB nutzt EF mit dem Ziel, ein Ende-zu-Ende-Verhalten bereitzustellen, das dem einer virtuellen Standleitung (Virtual Leased Line) entspricht, dass also ein Empfänger einen perfekten CBR-Datenstrom mit der Rate der virtuellen Standleitung sieht. Dazu muss das Netz am letzten Hop vor dem Empfänger den bursthaften EF-Verkehr mittels eines Verkehrsformers in einen CBR-Datenstrom der vereinbarten Rate formen. Für die Wahl des sogenannten Jitterfensters, um welches das erste Paket verzögert werden muss, damit nachfolgende Pakete nie zu spät eintreffen können, muss der maximale Jitter bekannt sein. Wie bereits erwähnt, zeigen Simulationen, dass zwar die maximale Verzögerung und damit der maximale Jitter sehr groß werden kann, aber dies nur für einen vernachlässigbaren Teil aller EF-Pakete. Akzeptiert man einen gewissen (sehr kleinen) Paketverlust, kann das Jitterfenster hinreichend klein gewählt werden. Schließlich sollte das Kernziel eines Dienstes wie des Premium Service bzw. der virtuellen Standleitung die möglichst geringe Ende-zu-Ende-Verzögerung sein, wesentliche Voraussetzung gerade im Hinblick auf den Internet-weiten Einsatz von interaktiven Echtzeitanwendungen. So scheitert derzeit die IP-Telefonie in Weitverkehrsnetzen an den großen Verzögerungsschwankungen, die zusammen mit der nicht vermeidbaren hohen Grundverzögerung zu nicht mehr akzeptablen Ende-zu-Ende-Verzögerungen führt.

4 Neue Dienste im Internet der nächsten Generation

Mit den eben vorgestellten Mechanismen können beispielsweise klassische Dienste für dienstgütebasierte Netzwerke realisiert werden:

- Audio-/Video-Übertragung, VoIP etc. über das Virtual Wire PDB, und
- VPN-Verbindung, elastische Anwendungen etc. über das Assured Rate PDB

Jedoch hat sich das Anwendungsspektrum im Internet in den letzten Jahren verändert. Im Zuge der Globalisierung nutzen Firmen das Internet als Basis zur Vernetzung ihrer Datenbasen, zur Kommunikation mit Kunden und Partnern und zur Abwicklung ihrer Bank- bzw. Börsengeschäfte. Im Freizeitbereich wächst die Zahl verteilter interaktiver Spiele, und Entertainment-Tauschbörsen wie Napster oder illegale Videoserver nehmen innerhalb nur weniger Monaten enorme Anteile an der genutzten Bandbreite ein.

Im Hinblick auf die Dienstgüteunterstützung und das Traffic Engineering im Internet der nächsten Generation ergeben sich im Zuge dieser Entwicklungen zwei Gebiete, in denen Handlungsbedarf besteht: Zum einen eine Dienstgüteunterstützung für transaktionsbasierte Anwendungen (verteilte Datenbankanwendungen, Bank- und Börsentransaktionen, Online-Spiele etc.) und zum anderen einen Schutz des „normalen" Internetverkehrs (Best-Effort) vor zu aggressiven und volumenintensiven Anwendungen, wie etwa Remote-Backups, Musik- bzw. Filmtauschbörsen etc.

In den folgenden beiden Abschnitten werden zwei am Institut entwickelte Ansätze vorgestellt, die im Rahmen der Differentiated-Services-Architektur zur Lösung der oben genannten Anforderungen realisiert/genutzt werden können.

4.1 Dienstgüteunterstützung für transaktionsbasierte Anwendungen

Die Verkehrscharakteristik von transaktionsbasierten Anwendungen unterscheidet sich von den bisher im Rahmen von DiffServ betrachteten Anwendungen in vielerlei Hinsicht. Als Szenario wird im Folgenden eine typische Client-Server-Kommunikation angenommen (vgl.

Abbildung 4).

- Transaktionsanfragen bestehen meistens aus einem oder wenigen Paketen.
- Antworten bestehen aus einer größeren Datenmenge, die als Burst über mehrere Pakete mit einer sehr hohen Rate vom Server ausgesandt wird.
- Nach einer Transaktion verarbeitet der Client die Daten und stellt nach einer gewissen Zeit eventuell wieder eine Anfrage.

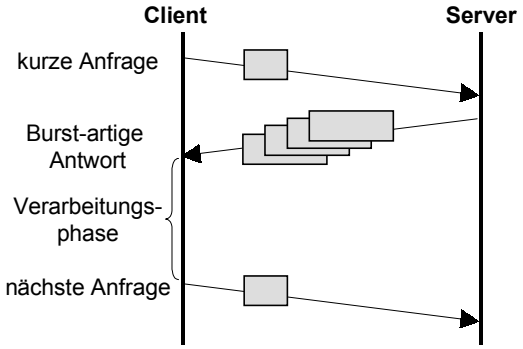

Abbildung 4: Typischer zeitlicher Ablauf einer Transaktion

Eine solche Verkehrscharakteristik wird von den in Abschnitt 2 vorgestellten PDBs Virtual Wire und Assured Rate nicht effizient unterstützt. Diese erwarten eher eine gleichmäßigere und längerfristige Kommunikationsbeziehung als in dem eben geschilderten Szenario. Des Weiteren stellt eine transaktionsbasierte Anwendung andere Anforderungen an qualitätsunterstützende Mechanismen im Netz, um die Zeit zwischen dem Aussenden einer Anfrage und dem Erhalten der Antwort möglichst gering zu halten:

- schnellstmögliche Weiterleitung im Netz, um die Ende-zu-Ende-Verzögerung sehr gering zu halten,
- sehr niedrige Paketverlustrate, um zeitaufwändige Übertragungswiederholungen auszuschließen,
- Burst-freundliche Weiterleitungsmechanismen, die ein Auseinanderziehen der Bursts verhindern.

Diese Forderungen werden von den PHBs EF und AF leider nicht erfüllt. EF besitzt zwar eine sehr geringe Verzögerung und sehr geringe Verlustrate, aber durch Verkehrsformung in den Grenzknoten von Domänen werden Pakete künstlich verzögert (vgl. Abbildung 5). Das AF PHB ist ebenfalls ungeeignet, weil es eine noch höhere Verzögerung als das EF PHB besitzt und Bursts bevorzugt verwirft.

Aus diesem Grund wurde am Institut für Telematik das Quick Forwarding (QF) PHB entworfen, mit der Maßgabe, transaktionsbasierte Anwendungen so gut wie möglich zu unterstützen. QF erhält eine separate Warteschlange unterhalb des EF, die mit der zweithöchsten Priorität bedient wird. Somit wird eine Beeinflussung des EF PHBs vermieden und QF kann ungenutzte Ressourcen des EF ausnutzen. Eintreffende QF Bursts werden nur durch EF-Pakete unterbrochen mit voller Geschwindigkeit des jeweiligen Ausgangs bedient und damit ohne größere Verzögerung und ohne Auseinanderziehen der Paketfolge durch eine Verkehrsformung weitergeleitet. Um eine Benachteiligung der niederprioreren PHBs zu vermeiden, wird eine Zugangs- und Nutzungskontrolle für das QF PHB eingeführt. Somit kann durch eine geschickte Ressourcenplanung die Anhäufung von Bursts vermieden werden. Dies resultiert in einer niedrigen Verlustrate und in einer geringeren Belastung der unterliegenden Dienstklassen.

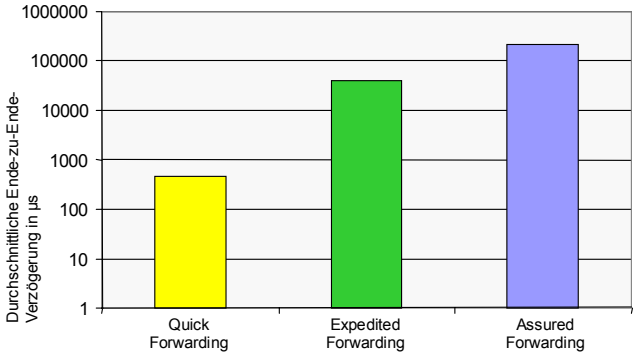

Abbildung 5: Verzögerung der PHBs EF, QF, AF

Abbildung 5 zeigt das Ergebnis von simulativen Untersuchungen des QF PHBs in größeren Szenarien [12]. Es ist deutlich erkennbar, dass beim Verzicht auf eine aktive Verkehrsformung, wie sie beim EF PHB zur Glättung eines Datenstroms eingesetzt wird, eine äußerst geringe Ende-zu-Ende-Verzögerung erreicht werden kann. In [12] wurden dabei keine Datenverluste bei QF-Paketen festgestellt.

Das QF PHB erfüllt somit die gestellten Anforderungen an eine qualitätsbasierte Unterstützung transaktionsbasierter Anwendungen. Weitere Anwendungen, beispielsweise die Weiterleitung von Routingkontrollpaketen für Intra-Domänen-Routingprotokolle, profitieren ebenfalls von QF. Weitere Untersuchungen zu detaillierteren Beschreibungen der Verkehrscharakteristik und flexibleren Algorithmen zur Nutzungskontrolle transaktionsbasierter Anwendungen sind vorgesehen.

4.2 Faire Nutzung der Best-Effort-Ressourcen

Das heutige Internet unterstützt noch keine Mechanismen zur Dienstgüteunterstützung. Dennoch spricht man von einer „erwarteten Qualität“, welche von den Nutzern – vor allem des World Wide Webs – gefordert wird. Unter der erwarteten Qualität wird verstanden, dass ein Dienst genutzt werden kann, auch wenn die Antwortzeiten oder Übertragungsraten schwanken. Schließlich bezahlen auch die heutigen Kunden für einen Dienst – die Konnektivität – und wenn diese nicht gewährleistet ist, sei es durch Leitungsunterbrechungen oder extensive Stausituationen, werden Kunden die Wahl ihres ISPs überdenken.

Durch die zunehmende Nutzung des Internets zur Übertragung von Massendaten (Backups, Offline-Verteilung bzw. Tauschbörse multimedialer Daten) können die traditionel-

len Nutzer des Best-Effort-Dienstes enorm beeinträchtigt werden. Ebenso kann bei der Nutzung von Multicast in DiffServ-Netzwerken die Situation entstehen, dass der Best-Effort-Verkehr durch degradierte Datenströme höherer Dienste unfair verdrängt wird [13].

In [14] wird deshalb das „Limited Effort“ PHB zur Trennung des o.g. „Bulk“-Verkehrs vom traditionellen Best-Effort-Verkehr vorgeschlagen. Er soll alle Datenströme weiterleiten, bei denen die faire Ausnutzung durch Adaption an die zur Verfügung stehende Bandbreite nicht gewährleistet ist (Non-Responsive Sources). Dem Limited-Effort-PHB steht eine garantierte Mindestbandbreite zur Verfügung, wobei ungenutzte Ressourcen höherpriorer PHBs genutzt werden können. Auf diese Weise werden die Best-Effort-Datenströme vor unfairen Kontrahenten geschützt und können die Best-Effort-Bandbreite fair unter sich aufteilen.

5 Ausblick

Die bisherigen Anstrengungen zur Dienstgüteunterstützung im Internet der nächsten Generation haben ein skalierbares Rahmenwerk hervorgebracht, in dem Internet Service Provider individuelle Weiterleitungsmechanismen und darauf aufbauende Dienste anbieten können. Des Weiteren wurden für die traditionellen Dienste zur statistischen und deterministischen Garantie einer bestimmten Bandbreite für elastische und nicht-elastische Anwendungen die Per-Domain Behavior Assured Rate und Virtual Wire definiert. Im Zuge der Realisierung der DiffServ-Architektur gilt es nun, Erfahrungswerte mit diesen Mechanismen zu sammeln, z.B. über Aggregationseffekte, die aufgrund der Einfachheit des DiffServ-Ansatzes auftreten können und durch mathematische Methoden nur schwer erfassbar und vorhersagbar sind.

Weiterhin müssen neue Mechanismen untersucht werden, um dem sich ständig erweiternden Anwendungsspektrum des Internets gerecht zu werden. Die PHBs Quick Forwarding und Limited Effort bilden hier einen ersten Ansatz.

Abschließend darf als wichtiger Punkt die Verwaltbarkeit eines dienstgüteunterstützenden Internets der nächsten Generation nicht außer Acht gelassen werden. Nachdem mit der Differentiated-Services-Architektur eine skalierbare Dienstgüteunterstützung in der Weiterleitungsebene (Datenpfad) erarbeitet wurde, muss dies nun auch auf der Kontrollebene und im Signalisierungspfad erfolgen. Ansonsten droht auch dieser Dienstgüteansatz zu scheitern, denn die Mechanismen der DiffServ-Architektur lassen sich zwar statisch konfigurieren, aber für den letztendlichen Erfolg wird eine Möglichkeit zur dynamischen Reservierung von DiffServ-Ressourcen benötigt. Eine dafür notwendige skalierbare Management-Architektur befindet sich derzeit in Entwicklung [15].

LITERATUR

[1] K. Kilkki: Differentiated Services for the Internet, Macmillan Technical Publishing, 1999

[2] F. Baker, B. Braden, S. Bradner, M. O'Dell, A. Mankin, A. Romanow, A. Weinrib und L. Zhang: Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement – Some Guidelines on Deployment, RFC 2208, IETF, September 1997.

[3] F. Baker, D. Black, S. Blake und K. Nichols: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, RFC 2474, IETF, Dezember 1998.

[4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang und W. Weiss: An Architecture for Differentiated Services, RFC 2475, IETF, Dezember 1998.

[5] K. Nichols und B. Carpenter: Definition of Differentiated Services Per Domain Behavior Aggregates and Rules for their Specification, RFC 3086, IETF, April 2001.

[6] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski: Assured Forwarding PHB Group, RFC 2597, IETF, Juni 1999.

[7] B. Davie, A. Charny, J.C.R. Bennett, K. Benson, J.Y. Le Boudec, W. Courtney, S. Davari, V. Firoiu, D. Stiliadis: An Expedited Forwarding PHB (Per-Hop Behavior), RFC 3246, IETF, März 2002.

[8] N. Seddigh, B. Nandy, J. Heinanen: An Assured Rate Per-Domain Behaviour for Differentiated Services, draft-ietf-diffserv-pdb-ar-01.txt, Arbeitsdokument der IETF, Juli 2001.

[9] V. Jacobson, K. Nichols, K. Poduri: The `Virtual Wire' Per-Domain Behavior, draft-ietf-diffserv-pdb-vw00.txt, Arbeitsdokument der IETF, Juli 2000.

[10] K. Nichols, V. Jacobson, L. Zhang: A Two-bit Differentiated Services Architecture for the Internet, RFC 2638, IETF, Juli 1999.

[11] A. Charny, J.-Y. Le Boudec: Delay Bounds in a Network With Aggregate Scheduling, Tagungsband der QoFIS 2000, Berlin, September 2000.

[12] R. Bless, K. Wehrle: Towards Better Support of Transaction Oriented Communications in DiffServ Networks, Tagungsband der QoFIS 2001, Coimbra, September 2001.

[13] R. Bless, K. Wehrle: IP Multicast in Differentiated Services Networks, IETF-Internet-Draft, draft-bless-diffserv-multicast-01.txt, November 2000.

[14] R. Bless, K. Wehrle, K. Nichols, B. Carpenter: A Bulk Handling Per-Domain Behavior for Differentiated Services, Arbeitsdokument der IETF, draft-ietf-diffserv-pdb-bh-03.txt, April 2002.

[15] R. Bless: Integrierte Managementarchitektur für Differentiated-Services-Netze, In: Klausurtagung des Instituts für Telematik, Interner Bericht 2000-6 der Fakultät für Informatik, Universität Karlsruhe, S.11–16, März 2000, http://www.ubka.uni-karlsruhe.de/vvv/ira/2000/6/6.pdf

[16] G. Huston: Next Steps for the IP QoS Architecture, RFC 2990, November 2000.